CRANFIELD UNIVERSITY

VINAYAK ASHOK PRABHU

A FRAMEWORK FOR DIGITISATION OF MANUAL MANUFACTURING TASK KNOWLEDGE USING GAMING INTERFACE TECHNOLOGY

SCHOOL OF AEROSPACE, TRANSPORT AND MANUFACTURING

PhD
Academic Year: 2014 - 2015

Supervisor: Professor Ashutosh Tiwari
October 2015

CRANFIELD UNIVERSITY

SCHOOL OF AEROSPACE, TRANSPORT AND MANUFACTURING

PhD

Academic Year 2014 - 2015

VINAYAK ASHOK PRABHU

A Framework for Digitisation of Manual Manufacturing Task Knowledge using Gaming Interface Technology

Supervisor: Professor Ashutosh Tiwari
October 2015

This thesis is submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy

ABSTRACT

Intense market competition and the global skill supply crunch are hurting the manufacturing industry, which is heavily dependent on skilled labour. Companies must look for innovative ways to acquire manufacturing skills from their experts and transfer them to novices and eventually to machines to remain competitive. There is a lack of systematic processes in the manufacturing industry and research for cost-effective capture and transfer of human skills. Therefore, the aim of this research is to develop a framework for digitisation of manual manufacturing task knowledge, a major constituent of which is human skill.

The proposed digitisation framework is based on the theory of human-workpiece interactions that is developed in this research. The unique aspect of the framework is the use of consumer-grade gaming interface technology to capture and record manual manufacturing tasks in digital form to enable the extraction, decoding and transfer of manufacturing knowledge constituents that are associated with the task. The framework is implemented, tested and refined using 5 case studies, including 1 toy assembly task, 2 real-life-like assembly tasks, 1 simulated assembly task and 1 real-life composite layup task. It is successfully validated based on the outcomes of the case studies and a benchmarking exercise that was conducted to evaluate its performance.

This research contributes to knowledge in five main areas, namely, (1) the theory of human-workpiece interactions to decipher human behaviour in manual manufacturing tasks, (2) a cohesive and holistic framework to digitise manual manufacturing task knowledge, especially tacit knowledge such as human action and reaction skills, (3) the use of low-cost gaming interface technology to capture human actions and the effect of those actions on workpieces during a manufacturing task, (4) a new way to use hidden Markov modelling to produce digital skill models to represent human ability to perform complex tasks and (5) extraction and decoding of manufacturing knowledge constituents from the digital skill models.

ACKNOWLEDGEMENTS

At the outset, I would like to express my gratitude to my supervisor, Professor Ashutosh Tiwari, for his brilliant supervision that helped keep this research in focus from day one while providing ample space for imagination and creativity. I thank him for sharing his knowledge and experience in manufacturing informatics and best practices in academic research, and for opening up opportunities for industrial collaborations.

I would like to express my sincere thanks and appreciation to my sponsor, Nanyang Polytechnic, Singapore for believing in me and giving me the precious opportunity to study for a PhD. I would like to dedicate my thesis to this wonderful organisation.

No amount of appreciation can ever be enough for my family; my parents for their inspiration and blessings; my wife, Nivedita for her enduring support, patience and understanding; my children, Deepika and Ridhi for their love and enthusiasm; and my sister Purnima for her cheerful encouragement. Without my family, this journey would not have even started.

My gratitude to Mr Windo Hutabarat for making me feel comfortable during the initial difficult days of this research and creating a vibrant research environment within the research group. Special thanks to Dr Christopher Turner, who as associate supervisor, has supported me whenever I needed his help especially in writing research publications. Special thanks also to Dr John Oyekan who is a valued friend and with whom I could bounce off research ideas and from whose PhD experience I took inspiration.

A big thank you to fellow researchers in Building 50 for their friendship and for making the research area a lively place to work in. Without them, this research journey would not have been as enjoyable as it has been.

Last but not least, I would like to acknowledge and appreciate the research support that I have received from the University of Bristol, the EPSRC Centre for Innovative Manufacturing in Intelligent Automation, Airbus Group Innovations, Jaguar Land Rover Limited and Holovis international.

LIST OF PUBLICATIONS

Journal Papers

1. Prabhu, V. A., Song, B., Thrower, J., Tiwari, A., & Webb, P. (2015). Digitisation of a Moving Assembly Operation using Multiple Depth Imaging Sensors. *International Journal of Advanced Manufacturing Technology.* (*Published*)
2. Prabhu, V. A., Tiwari, A., Hutabarat, W., & Turner, C. (2015). Capture, Digitisation and Segmentation of Human-Workpiece Interactions in a Manual Assembly Operation using Kinect. *International Journal of Design Engineering. (In press)*
3. Prabhu, V. A., Muhandri, N., Song, B., & Tiwari, A. (2015). Decision Support System Enabled by Depth Imaging Sensor Data for Intelligent Automation of Moving Assemblies. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture. (Revision under review)*
4. Prabhu, V. A., Elkington, M., Crowley, D., Tiwari, A. & Ward, C. (2015). A Framework for Digitisation of Manual Manufacturing Task Knowledge using Gaming Interface Technology*. IEEE Transactions on Industrial Informatics. (Under review)*
5. Prabhu V. A., Tiwari, A. (2015). Survey of Methods for Digitisation of *Human* Skills. IEEE Robotics and Automation Magazine. *(Under review)*

Conference Papers

1. Prabhu, V. A., Tiwari, A., Hutabarat, W., & Turner, C. (2014). Monitoring and Digitising Human-Workpiece Interactions during a Manual Manufacturing Assembly Operation Using Kinect. *Proceedings of the 5th International Conference on Advanced Design and Manufacture (ADM 2013)*, 609-612, 25 - 28 September 2013, Valencia, Spain. *(Published)*
2. Prabhu, V. A., Tiwari, A., Hutabarat, W., Thrower, J., & Turner, C. (2014). Dynamic Alignment Control Using Depth Imagery for Automated Wheel

Assembly. *Proceedings of the 8th International Conference on Digital Enterprise Technology (DET 2014)*, 161-168, 25 - 28 March 2014, Stuttgart, Germany. *(Published)*

3. Prabhu, V. A., Oyekan, J., Tiwari, A., Advikolanu, Y., Burgess, M., & McNally, R. (2015). Remote Real-Time Collaboration Platform Enabled by Capture, Digitisation and Transfer of Human-Workpiece Interactions. *Proceedings of the Sustainable Design and Manufacturing Conference 2015*, 12 – 14 April 2015, Seville, Spain. *(Published)*

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

LIST OF ABBREVIATIONS

2D	2-Dimensional
3D	3-Dimensional
API	Application Programming Interface
AR	Augmented Reality
BP	Base Point
CAD	Computer Aided Design
CCD	Charge Coupled Device
CRF	Conditional Random Fields
CSV	Comma Separated Values
CTA	Control Task Analysis
CWA	Cognitive Work Analysis
DMM	Depth Motion Maps
E2LSH	Exact Euclidean Locality Sensitive Hashing
ECG	Electro-Cardio Gram
EMG	Electromyography
EM	Expectation Maximisation
FPS	Frames Per Second
GMM	Gaussian Mixture Model
HAR	Human Activity Recognition
HMM	Hidden Markov Modelling
HOG	Histogram of Oriented Gradients
HTA	Hierarchical Task Analysis
ICA	Independent Component Analysis
ICT	Information and Communication Technology
IR	Infra-Red
JLR	Jaguar Land Rover
KBS	Knowledge Based System
LP	Leg Point
MLE	Maximisation of Likelihood Estimation

MOCAP	Motion Capture
MRF	Markov Random Field
ORB	Oriented Brief
OS	Operating System
PCA	Principal Component Analysis
PCD	Pitch Centre Diameter
PCOG	Pyramid Correlogram of Oriented Gradients
PHMM	Parametric Hidden Markov Modelling
RGB	Red Green Blue
RGB-D	Red Green Blue - Depth
SA	Strategy Analysis
SDK	Software Development Kit
SGPLVM	Scaled Gaussian Process Latent Variable Model
SOCA	Social Organisation and Cooperation Analysis
SPORE	Self-Organising and Dimensionality Reducing
SRK	Skills Rules Knowledge
SSVM	Structural Support Vector Machine
SVM	Support Vector Machine
TSTG	Tempo-Spatial Topological Graph
VIBE	Visual Background Extractor
VR	Virtual Reality
WCA	Worker Competency Analysis
WDA	Work Domain Analysis

CHAPTER 1

1 INTRODUCTION

This chapter introduces the research topic by stating the need for this research, describing the motivation behind conducting it, providing the basic terms of reference, explaining the research concept and outlining the structure of this thesis document. This chapter also introduces the gaming interface technology used in this research.

1.1 Research need and motivation

There is intense global competition in the manufacturing industry with jobs being outsourced from high-wage to low-wage economies. The emergence of the global skill supply crunch (Figure 1) makes matters worse for manufacturing companies especially in high-wage economies resulting in the ever increasing cost of hiring skilled manpower, a constant decline in productivity due to unmatched skillsets of existing manpower and the prospects of reduced growth due to constrained manpower capacity.

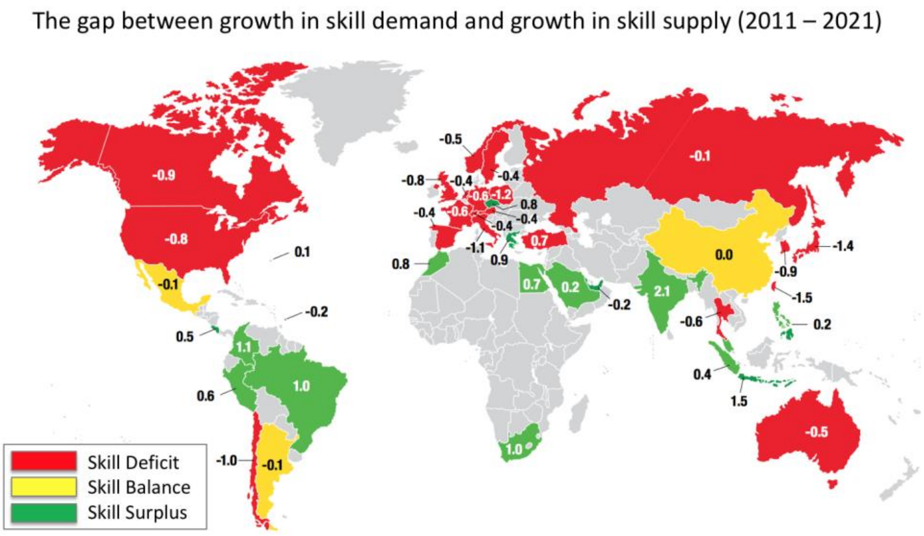


Figure 1: The global skill heat map (Oxford Economics, 2012)

According to the UK Commission for Employment and Skills (UKCES, 2012), the UK is facing the effects of this global skill crunch with 1 in 5 vacancies proving difficult to fill. Out of the total skill shortage vacancies, about 30% belong to the manufacturing sector alone.

Despite the challenges, manufacturers must ensure that their products remain competitive and that they speed up the time-to-market and at the same time minimise manufacturing cost (Padrón et al., 2009). In such a situation, it is the need of the hour for the industry to capture and digitise the manufacturing task knowledge from skilled experts that perform these tasks today so that this knowledge could be used to up-skill the next generation of workforce and build intelligent solutions to automate some of these tasks in the future. This requirement was confirmed by a recent workshop on autonomous manufacturing conducted by the Engineering and Physical Sciences Research Council (EPSRC), with representatives from manufacturing companies such as Airbus, Jaguar Land Rover and Siemens, in which capture of human skills and modelling of complex manufacturing tasks were considered essential for tomorrow's factories to be more productive and adaptive (EPSRC, 2014).

A cohesive and holistic framework to capture and digitise manufacturing task knowledge is currently lacking both in the industry as well as in academic research and this research aspires to develop such a framework.

1.2 Terms of Reference

1.2.1 Manufacturing task

Before presenting an overview of manufacturing knowledge and the framework to digitise it, it is necessary to briefly describe the general representation of a manufacturing task in the perspective of this research and explain the use of the common terms used in this chapter.

Manufacturing tasks can be classified into three primary areas; machining, assembly, and inspection as represented using simple illustrations in Figure 2.

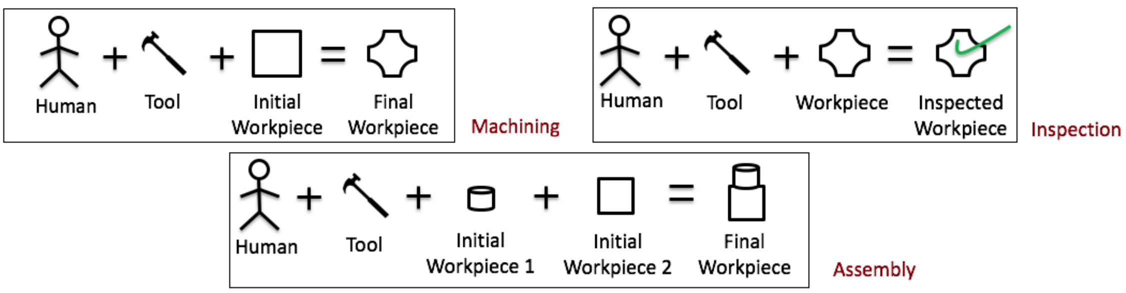


Figure 2: Simplified representation of a manufacturing task

Terms:

- Human = a skilled expert in a manufacturing enterprise who can be an engineer, an operator or a technician.
- Workpiece = an object or a set of objects that are processed within a manufacturing task
- Tool = an object used to process the workpiece. It could be a machining tool, an inspection tool or a machining centre.

Figure 2 represents typical machining, inspection and assembly tasks. In case of machining, the human worker uses a machining tool to process the workpiece from its initial state to its final state or in case of inspection, uses an inspection tool to inspect a workpiece (Inaba et al., 1999). In an assembly task, a human worker assembles a minimum of two workpieces with or without a tool to form the assembled workpiece (Nof et al.,1997).

1.2.2 Manufacturing knowledge

Hicks et al. (2002) consider data, information and knowledge as important commodities for modern, globalised companies and effective use of these commodities is increasingly proving to be a solution to gain and sustain a competitive advantage. Before exploring the concept of knowledge digitisation, the differences between data, information and knowledge must be explained. Data is a collection of words, symbols or numbers without a context or meaning. Information is a collection of data that when processed provides particular meaning within a context. Knowledge is a collection of information with details on how the information should be used within a specific context and an understanding of data relationships. A significant body of research work (Wiig, 1994, Nonaka and Takeuchi, 1995, Davenport and Prusak, 1997, Tuomi, 1999, Guerra-Zubiaga and Young, 2008) on knowledge management has addressed the subject of what is knowledge and how it is related to data and information.

In the manufacturing domain, knowledge is multi-faceted and primarily contains data and information about the products such as product designs, models and materials and about the manufacturing tasks such as people, machines, tools, processes, standard operation procedures and the manufacturing environment constraints. This research focuses on digitisation of manual manufacturing task knowledge. A general classification of manufacturing knowledge is shown in Figure 3 in which there are two main types of knowledge associated with a manufacturing environment, namely, explicit knowledge and implicit/tacit knowledge (Polanyi, 1966).

Explicit knowledge is the technical or academic data that is described in formal language, like manuals, mathematical expressions, copyright and patents. According to Smith (2001) this 'know-what' or systematic knowledge is readily communicated and shared through print, electronic and other formal means.

Tacit knowledge or hidden and implicit knowledge is a term related to human skills. Polanyi (1966) was the first to associate tacit knowledge with physical action that cannot be described verbally, such as the knowledge of riding a bicycle. Tacit knowledge as defined by Nonaka (1991) is the knowledge based on experience and insight that cannot be expressed verbally and therefore cannot be documented and transferred easily.

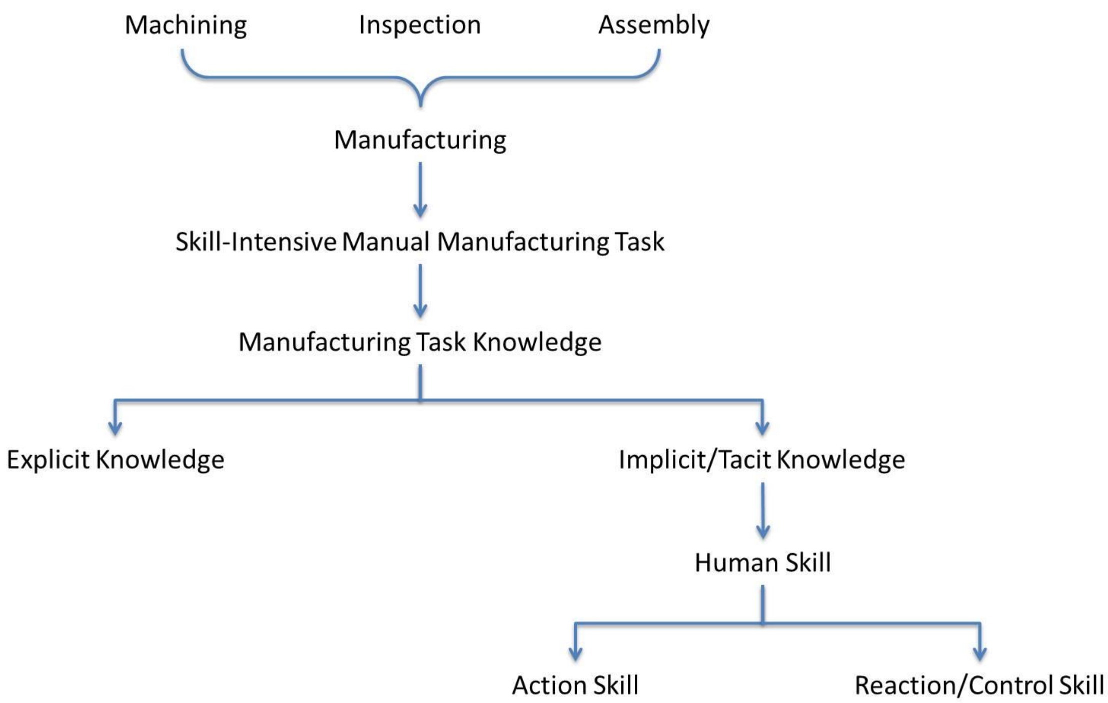


Figure 3: Human skill as tacit knowledge in manufacturing

Smith (2001) defines technical tacit knowledge as a specific body of knowledge or human skills learnt by people such as those gradually developed by master craftsmen. Nickols (2000) is of the opinion that tacit knowledge can be communicated or transferred, and it can still be acquired by other means besides verbal descriptions. According to Nonaka and Takeuchi (1995), the acquisition of tacit knowledge takes place through observation, imitation and practice, and the most efficient way to transfer or store tacit knowledge is through the use of sketches, video clips, storytelling and patterns. This finding is also confirmed by the research reported by Guerra-Zubiaga and Young (2008).

This research agrees with the opinion of Nickols (2000) that tacit knowledge can be acquired and communicated and therefore aims to develop a framework for digitisation of manual manufacturing task knowledge. In this framework, human-workpiece interactions, within which human skills are embedded, are modelled and the manufacturing knowledge extracted from the models is represented in a digital reusable form.

1.2.3 Human skill

Human skill forms a significant part of tacit knowledge and in turn is classified into action (or motion) skill and reaction (or control) skill according to Xu and Yang (1995). Duan et al. (2008) consider reaction skill as a human decision making ability to select the most appropriate actions to execute according to the task situation and action skill as the human's superior movement to execute the task when he/she knows what to do. The reaction skill is responsible for the choice of actions made depending on the situation of the task which the human observes and analyses whereas action skill is responsible for the precise gestures, motion mechanics, workpiece manipulation techniques, etc. that the human performs during the task. In this classification, skill is hierarchical: the reaction skill is a higher level skill than action skill i.e. the output of the reaction skill is the input for the action skill.

1.3 Need for skill extraction and transfer

Increasing global competition and the global skill supply crunch has created a need for the manufacturing industry to look for quick and effective up-skilling solutions. Yoshida et al. (2011) are of the opinion that for a sustainable developed society in which manufacturing industries thrive; expert engineers must transfer their knowledge and skills to young learners. Since human skill is in tacit form, it is not easily shared even if the work procedures of operation in manufacturing industries are rigidly defined. However, if this tacit knowledge were to be extracted and converted to explicit knowledge, the young engineers can easily learn the skill and reach a level of competency required to maintain the skill levels and efficiencies of companies.

Automation is another solution that will make the industry less dependent on human skill supply. An effective and efficient integration of automation tools and technologies for industrial production has the potential to improve the competitiveness of manufacturing companies (Sattar et al., 2014). However, the paradox of manual labour in specific operations of otherwise highly automated manufacturing systems is consistent in today's industry (Georgilas and Tourassis, 2008). Polishing, spray painting, layup of composite fibre plies, complex assembly and inspection tasks are still performed manually by skilled workers. Automation of such jobs is not easy because according to Duan et al. (2008), the main problem is that the robots do not have the abilities to react to uncertainties in an unpredictable environment because they have been rigidly programmed to do a particular task unlike the humans who handle complex unpredictable tasks well.

Humans manage skill-intensive jobs with relative ease but it is difficult for them to make formal definition to describe their own skills; therefore, effectively analysing and extracting human skills to make skill models and then using these skill models to realise the skill transfer process is a worthwhile idea (Duan et al., 2007). These skill models can be used to transfer human skill to train less-skilled human operators or robots. Since the action and reaction skills are interlinked, a skill capture and transfer process for both skills is desired.

1.4 Need for the proposed digitisation framework

A manual manufacturing task is a complex activity comprising humans, tools, workpieces and environments, and results in the building of products. The human is the main actor in this activity, who possesses the intellect and skills to significantly influence the outcome of the task, guiding it to successful completion by solving problems along the way. Though intellect is an inherent virtue, skills can be learnt via training, repetitive practice and/or on-the-job experiences. However, the extent of teaching or learning is subjective and depends on the individuals involved in the skill-transfer process (Yoshida et al., 2011). Thus the competitiveness and longevity of a manufacturing firm that is dependent on skilled labour is a function of how quickly and effectively it enables the transfer of skills from one generation of its workforce to the next.

There are several reasons for these manufacturing tasks to have remained manual in nature despite the widespread use of automation in industry. One of them is the dearth of complete understanding of these tasks, especially the tacit knowledge such as human skills that are embedded within the tasks (Georgilas and Tourassis, 2008). This lack of knowledge deters automation of these tasks thereby resulting in a constant demand of skilled manpower.

Therefore, there is a need to develop a framework for the capture of tacit knowledge from skill-intensive manual manufacturing tasks. The framework must be cohesive, comprehensive, scalable and flexible to be adopted widely by the industry covering a broad spectrum of manual manufacturing tasks.

1.5 Research concept

In this research, human skill is considered as an enabler that allows the human to successfully perform a manual manufacturing task and as a major component of knowledge embedded within the task. The research further considers a manual manufacturing task as a series of human-workpiece interactions in which every action by the human on the workpiece is followed by feedback from the workpiece on its state of progress to which the human reacts by performing the next suitable action. This action-feedback-reaction loop continues till the task is successfully completed. Thus, skill manifests itself through these interactions and this research postulates that by capturing and analysing these interactions, human skill could be extracted.

The concept of this research is to capture and record human-workpiece interactions that take place during a manual manufacturing task. These captured interactions are segregated into human action and workpiece states that are then studied to establish a cause effect relationship between them. Once these relationships are established, human action sequences responsible for critical workpiece changes during the task are extracted and the key ingredients that make up those human actions are decoded. These ingredients are precisely what make up manufacturing knowledge, including human skill that is embedded within the task (Figure 4). A framework is thus developed to implement this concept in a structured manner and it is tested on both simulated and real-world case studies by using it to digitise the knowledge that is embedded within those tasks.

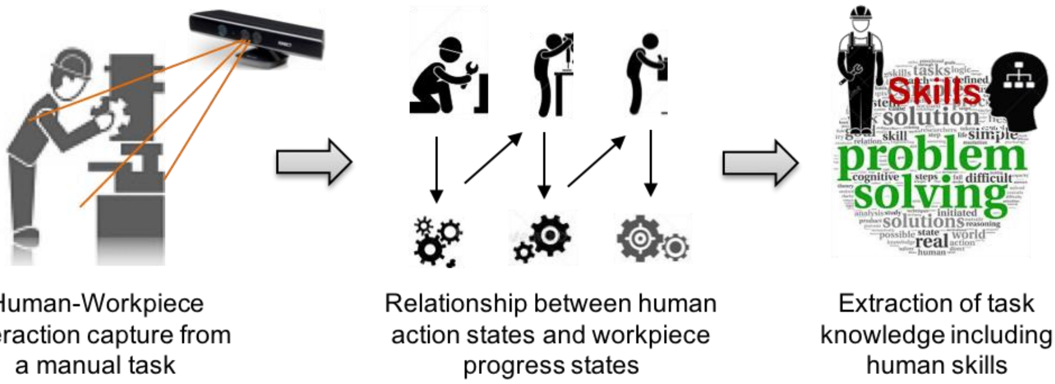


Figure 4: Research Concept

The tasks used in the first 4 case studies are manual assembly tasks. The task in the 5^{th} case study can also be broadly classified as an assembly task. Assembly is a sub-system of a manufacturing system and involves bringing and joining parts and/or sub-assemblies together (Marian, 2003). The primary reason for this choice is that assembly tasks can be simulated under simplified and structured laboratory conditions with model artefacts and this simplification reduces the complexity of human action and object tracking methods needed in the framework. The choice of assembly tasks is also valid from the industry perspective because assembly consumes up to 50% of total production time and accounts for more than 20% of total manufacturing cost (Pan, 2005).

Gaming interface technology is proposed as a task capture tool to capture the human-workpiece interactions. This technology brings human motion capture and workpiece tracking functionality to the framework via the use of commodity-priced off-the-shelf gaming device without the need for any pre-calibration. One such device is the Microsoft Kinect™ (Microsoft, 2014). Other motion capture technologies studied were multiple camera systems from Vicon (Aachen, 2012) and wearable inertial accelerometer sensor systems from Xsens (Xsens, 2015). Both these systems were not selected because they of their relatively higher costs (two orders of magnitude greater than the price of Kinect), both are marker-based hence obtrusive and both are not suited for object recognition and tracking.

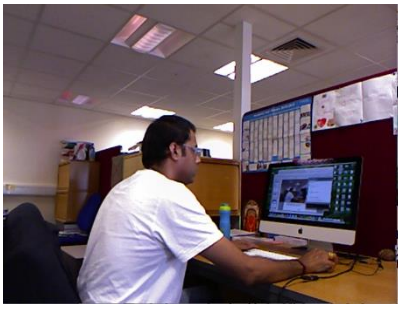
1.5.1 Gaming interface technology

Modern game design is human-centric and aims to involve a gamer in the game not only mentally but also physically. This is achieved by capturing the gamer's body movements using portable motion capture technology as he/she navigates through the game, for instance while playing a game of virtual tennis. This motion is analysed to extract specific human actions and reactions to game situations and the game scenarios are adjusted accordingly. Portable motion capture technology is made available using 3D sensing devices that are packaged with the latest high-tech gaming consoles. Kinect™ is one such device that is used in conjunction with the Xbox™ gaming console made available by Microsoft Corporation.

About the Kinect

The Kinect is a motion sensing device which enables gamers to control and interact with their virtual games through a natural user interface without the need for a game controller. This interface uses human gestures and spoken commands rather than joystick, keyboard or mouse inputs. Though the device was launched in November 2010 as a gaming tool, it was released for the development of gesture and speech controlled applications only in June 2011. In July 2014, Kinect V2, the second generation of the device with better specifications and features was launched. The price point of under £150 per unit and its portability and robustness proved to be major advantages for application and game developers alike. The first Kinect generation (Kinect V1) holds the Guinness World Record for being the 'fastest selling consumer electronics device' (BBC, 2011).

The Kinect V1, which works at a resolution of 640 x 480 pixels, is the primary task capture tool used in this research. It comprises an RGB camera that produces colour images at the rate of 30 frames per second (fps) and an IR camera that produces depth images also at the rate of 30fps (Figure 5). Therefore, for every pixel in the scene recorded by the Kinect, six elements of useful information can be acquired, namely, its red, green and blue colour values, its x and y positions relative to the screen coordinate system, and its distance (in mm) from the Kinect in absolute value (Borenstein, 2012).

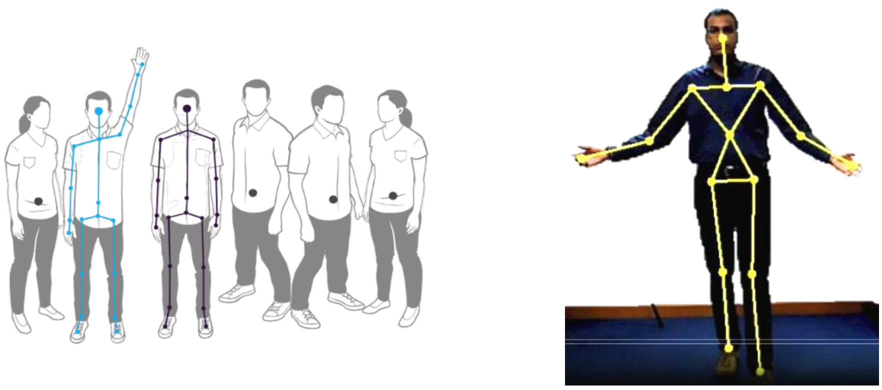


(a)

(b)

Figure 5: (a) RGB image (b) Depth image produced by the Kinect

The Kinect comes with a software development kit (Kinect MSDN, 2013) that comprises standard functions to identify up to 6 humans from within the scene and to track the positions of up to 20 skeletal joints for up to 2 humans (Figure 6). This motion tracking functionality is provided without the need for any additional software coding. This research uses the standard human skeletal motion tracking functionality of the Kinect with a standard high-pass filtering and threshold-averaging algorithm to filter and smooth the motion capture data.



(a) (Microsoft, 2014) (b)

Figure 6: (a) Standard human identification and skeletal tracking (b) Human skeletal tracking in this research

Object recognition and tracking was a difficult task earlier using colour image processing techniques. But with the availability of depth information, object recognition and tracking has become relatively easier. In this research, the Kinect is used to recognise and track objects at the same time as it tracks and captures human motion. Therefore, with just one Kinect device, the human-object interactions occurring during a manual manufacturing task can be captured and recorded. The Kinect is therefore used as the primary task capture tool in the proposed digitisation framework.

There are several software development kits, which implement markerless motion capturing algorithms on Kinect depth images. PrimeSense NiTE middleware library from OpenNI offers a platform-independent solution with low computational costs and is the platform used in this research. The other development kit is from Microsoft called the ‘Kinect for Windows SDK’, which works only on the Windows platform. These software development kits provide functions that extract absolute human joint coordinates in 3D in real-time.

The advantages of using the Kinect are its low cost, high portability and that it does not require any pre-calibration. However, human motion can only be captured from the front view where the human faces the device. If the human turns his back to the sensor, it may fail to track the motion correctly. The use of multiple Kinect devices may be used to capture different perspectives to solve this problem but the projected Infra-Red (IR) patterns from the devices would interfere with each other resulting in bad depth image quality and therefore unreliable motion capture. The Kinect also suffers from unreliable motion capture due to occlusions, which is when the complete view of the human is obstructed by a large object. More details about the Kinect including its limitations and the introduction of the next generation of the sensor, Kinect V2 is presented in CHAPTER 7.

1.6 Thesis structure

This thesis is structured into 8 chapters (Figure 7):

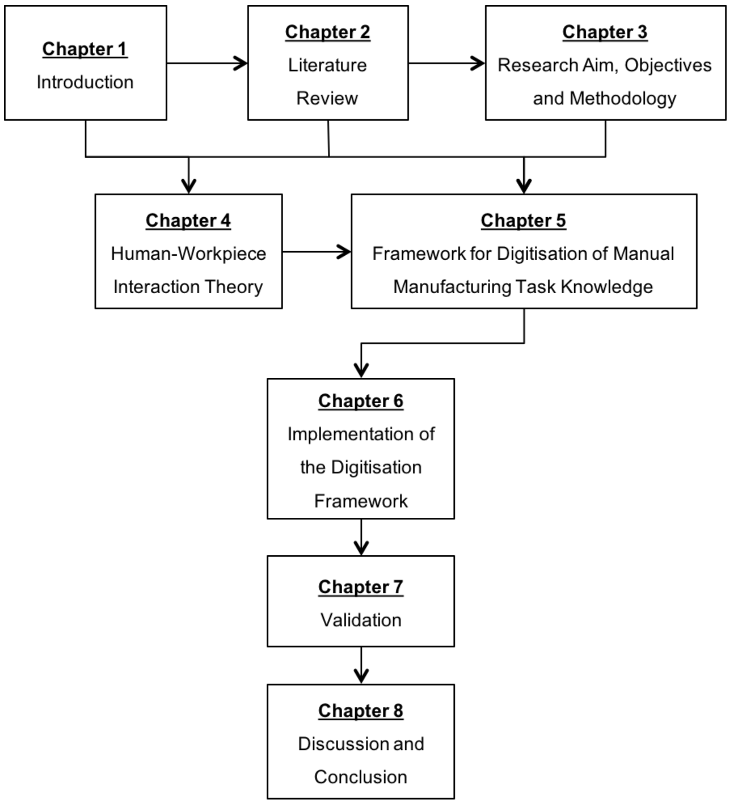


Figure 7: Thesis structure diagram

Chapter 1 gives an introduction to the research. It also presents the research problem and motivation.

Chapter 2 reviews literature in the human skill capture, extraction, modelling and transfer domains, collectively referred to as digitisation, for manufacturing applications. In this chapter, the literature survey is performed to identify the research trends and find research gaps in the area.

Chapter 3 outlines the research aim, objectives and scope. This chapter also presents the research methodology to explain how this research is conducted.

Chapter 4 gives the description of the underlying concept of human-workpiece interactions and the use of seminal theories from literature for the advancement of human-workpiece interaction theory.

Chapter 5 proposes a framework for digitisation of manual manufacturing task knowledge. This chapter introduces Hidden Markov Models and proposes their use as a stochastic tool to model human-workpiece interactions.

Chapter 6 presents the implementation of the framework by using an example task of Lego blocks assembly. It also presents the process of digitising the manufacturing environment using an automotive wheel-loading task.

Chapter 7 presents the validation case studies to test whether the digitisation framework can extract and decode manual manufacturing task knowledge from 3 different tasks, including one from the composites manufacturing industry. The case studies presented are i) digitisation of a pen assembly task, ii) digitisation of an Ikea table assembly task and iii) digitisation of a manual composite layup task. Collectively in these studies, different methods and tools proposed within the framework are implemented, tested, refined and validated.

Chapter 8 discusses the contribution of this research to knowledge and the advantages and limitations of this research. Finally, this chapter discusses future research direction and presents conclusions.

1.7 Chapter summary

This chapter introduces the terms frequently used in this thesis; manufacturing, manufacturing knowledge and human skills in manual manufacturing. It highlights the need to digitise manufacturing knowledge embedded within manual manufacturing tasks and presents a research concept to investigate the development of a framework to digitise this knowledge.

CHAPTER 2

2 LITERATURE REVIEW

This chapter reviews the existing work on digitisation of manual manufacturing task knowledge as well as existing theories in human behaviour and human-object interaction that could be investigated to develop this research. In the context of this research, digitisation means the capture, modelling and extraction of the constituents that make up manufacturing knowledge, especially those that are tacit in nature, such as human skill, and representing the extracted knowledge in digital form.

The aim of this chapter is two-fold: (i) to give an overview of the proposed processes of manufacturing knowledge digitisation from literature including the reported approaches, methods, tools and techniques, and (ii) to learn from established theories that underpin the development of the proposed human-workpiece interaction theory in this research. This chapter also discusses the research gaps in manufacturing knowledge digitisation and the research trends in the areas of human motion capture, object recognition and tracking, human-object interaction and gaming interface technology.

This chapter attempts to achieve the following objectives:

- ❖ Provide an overview of manufacturing knowledge and reported research in the digitisation of manual manufacturing task knowledge.
- ❖ Provide an overview of established theories in human behaviour analysis and human-object interaction analysis pertaining to the manufacturing industry.
- ❖ Identify the steps used in the manufacturing knowledge digitisation process.
- ❖ Analyse the research approach and different Information and Communications Technology (ICT) methods, tools and techniques used.
- ❖ Identify the research gaps.
- ❖ Discuss the research trends.

2.1 Background

Though automation is pervasive in modern manufacturing, manual labour is still used in a large variety of complex tasks that use human skill, which is a complex mix of dexterity, precision, accuracy and sophisticated cognition. Human skill is a major part of the manufacturing knowledge associated with skill-intensive manual manufacturing tasks. Companies relying heavily on skilled manpower must effectively capture and archive skills of their experts so that they can be transferred to the next generation of the workforce and eventually to machines. Therefore digitisation of manufacturing knowledge has become a key requirement for the industry (Young, 2003) and the associated processes for capturing that knowledge have received keen interest from many researchers. Most of the human action and object recognition work in literature belongs to the social robotics, human-robot interaction and human-machine interface domains. While, some of this work is analysed and presented later in this chapter, relevant work in the core area of digitisation of human skills applicable in the manufacturing domain has been consolidated and presented in this chapter. The structure of the literature review is presented in Figure 8.

This review is made up of two distinct focus areas. The first focus area targets those articles in literature that report research in digitisation of manufacturing knowledge, especially human skills from manual manufacturing tasks. The literature is surveyed to uncover any existing frameworks that are reported to digitise human skill and also to understand the different methods and tools used in the digitisation process, including the adoption of gaming interface technologies. By analysing the results of this survey, research gaps are identified and research trends are predicted. This focus area forms the basis for the development of the proposed digitisation framework in this research. The second focus area targets those articles in literature that study manual manufacturing tasks to obtain further insights into human behaviour, human problem-solving processes, human task analysis and human-object interaction in manufacturing industry settings. A survey of the literature is conducted to study if the concept of human-workpiece interactions has been proposed before to digitise manual manufacturing task knowledge.

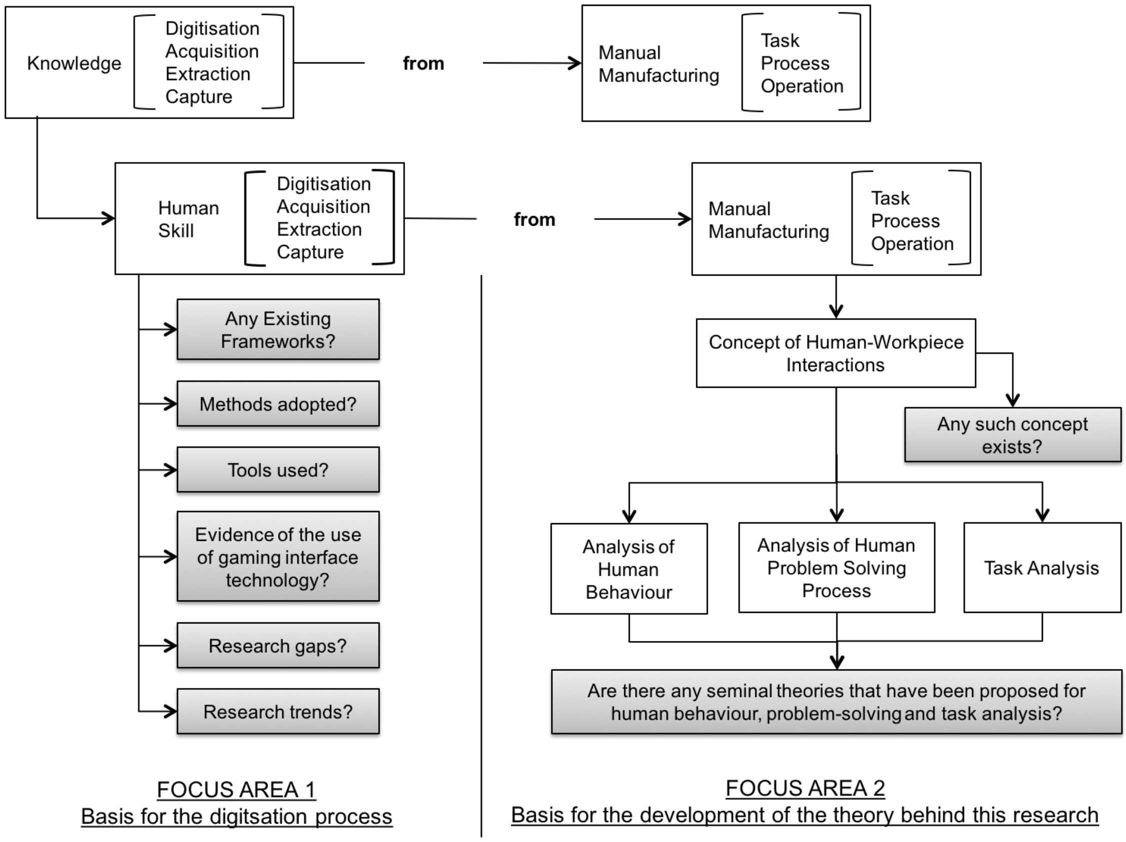


Figure 8: Literature Review Structure

Seminal articles are obtained to find useful insights into the interplay of human, workpieces and the manufacturing environment during manufacturing tasks in order to reinforce the proposed human-workpiece interaction concept. Recent articles in the areas of human action recognition, object recognition and tracking and human-object interface studies are also studied to examine the current state-of-the-art in this area.

2.2 Focus Area I: Digitisation of human skills – skill capture, modelling, extraction and transfer

Skill capture, extraction, modelling and transfer, referred to collectively as ‘digitisation’ has been centred on the following main entities belonging to a manual manufacturing task, namely, a) human, b) workpiece, c) tool, d) a combination of human and tool, and e) a combination of human and workpiece. The process level classification of this skill digitisation process comprising five main sequential steps is illustrated in Figure 9.

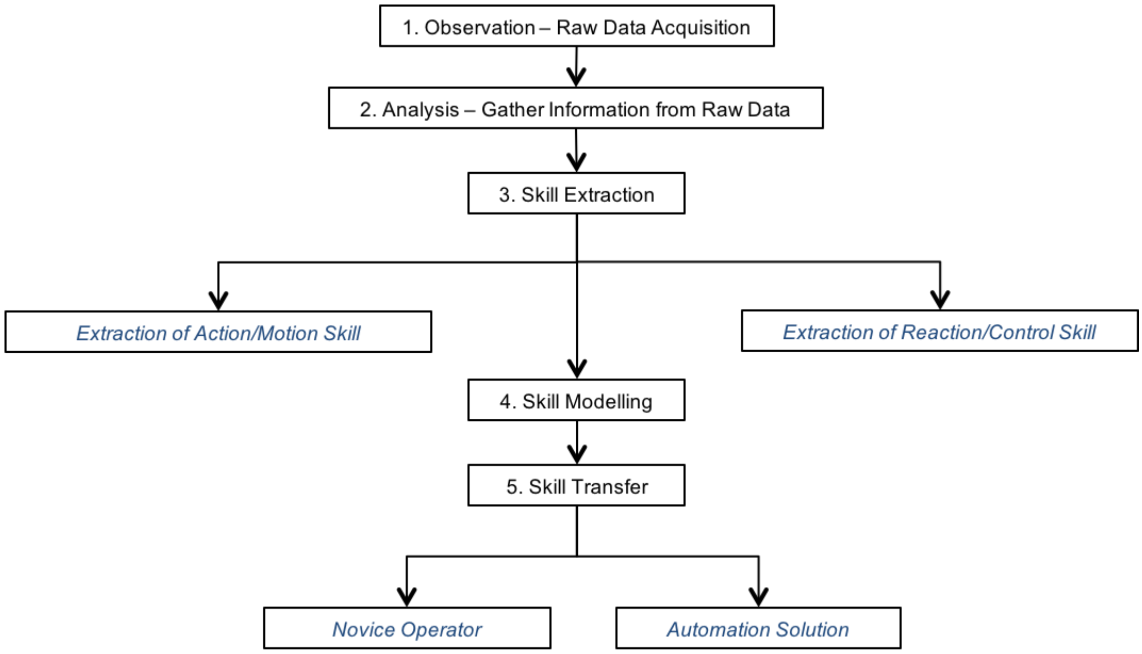


Figure 9: Process-level classification of human skill digitisation

The digitisation process begins by acquiring raw data upon observation of the human perform the manual manufacturing task. A stereovision system is one type of tool used for raw data acquisition (Kruger et al., 2010). The raw data is then analysed to obtain useful information about the operation such as positions and states of the human (Marfia et al., 2012), workpiece (Matsuki, 2010) and tool (Kawashimo et al., 2009) during the operation from start to finish. From this preliminary information, human action and reaction skills are extracted using algorithms that look for specific features in the data such as human motion trajectories (Tommaso et al., 2012), human action primitives, such as picking up and releasing objects, (Faria et al., 2012), workpiece contact states (Skubic and Volz, 2000), and decision making and control strategies (Duan et al., 2008). This extracted skill is then modelled and represented using standard methods such as hidden Markov models (Calinon and Billard, 2005) so that the skills can be documented as explicit knowledge and transferred to novice operators or to enable automation (Duan et al., 2008).

Several methods, tools and techniques have been reported in literature to implement each of the above 5 steps and are illustrated as technical-level classification in Figure 10 and Figure 11.

2.2.1 Raw data acquisition

Raw data is acquired by observing the skilled human perform the manufacturing operation. 8 methods used for observation and data acquisition have been reviewed.

(i) The stereo vision system consists of one or more cameras that capture colour stereo images, at a specified resolution and frame rate, the motion traversed by the human operator and his/her manipulation of the tool and the workpiece while performing the operation (Yeasin and Chaudhuri, 2000). Two types of vision systems are used: one which requires the human and the objects to wear markers for visible distinction and tracking (Kikuchi et al., 2013) whereas the other which does not require any markers (Marfia et al., 2012) in which object recognition and filtering algorithms are utilised for motion tracking. (ii) The multi-sensor system is a combination of two or more devices, such as vision cameras, eye movement trackers, force sensors, tactile sensors and inertial sensors (Faria et al., 2012), where not only the human motion coordinates but also the kinematics of motion, human body orientation, grasp transitions on objects, and tactile signatures of the human hand and eye movements are acquired. (iii) In order to capture human control strategy data, a skilled human operator programs a robot (Okuda, 2007) or tele-operates a robot (Grudic and Lawrence, 1996) remotely or using a haptic interface device (Skubic and Volz, 2000) to manipulate tools and workpieces. The resulting robot motion is tracked to obtain its position and orientation during the entire tele-operation. (iv) The fourth data acquisition method is to use simulations as a tool to acquire operation data. Expert skill such as tool manipulation paths and operation conditions are built into kinematic simulation software (Tsai et al., 2012) to predict how the workpiece will move based on the set of forces and constraints acting on it. Dynamic simulators are used by Duan et al. (2008) to record control data as a result of human manipulation of objects in simulated environments. (v) Visual observations of a skilled human while performing an operation and interviewing the human after the operation is a method of acquiring data about the actions and decisions made during the operation (Hashimoto et al., 2011).

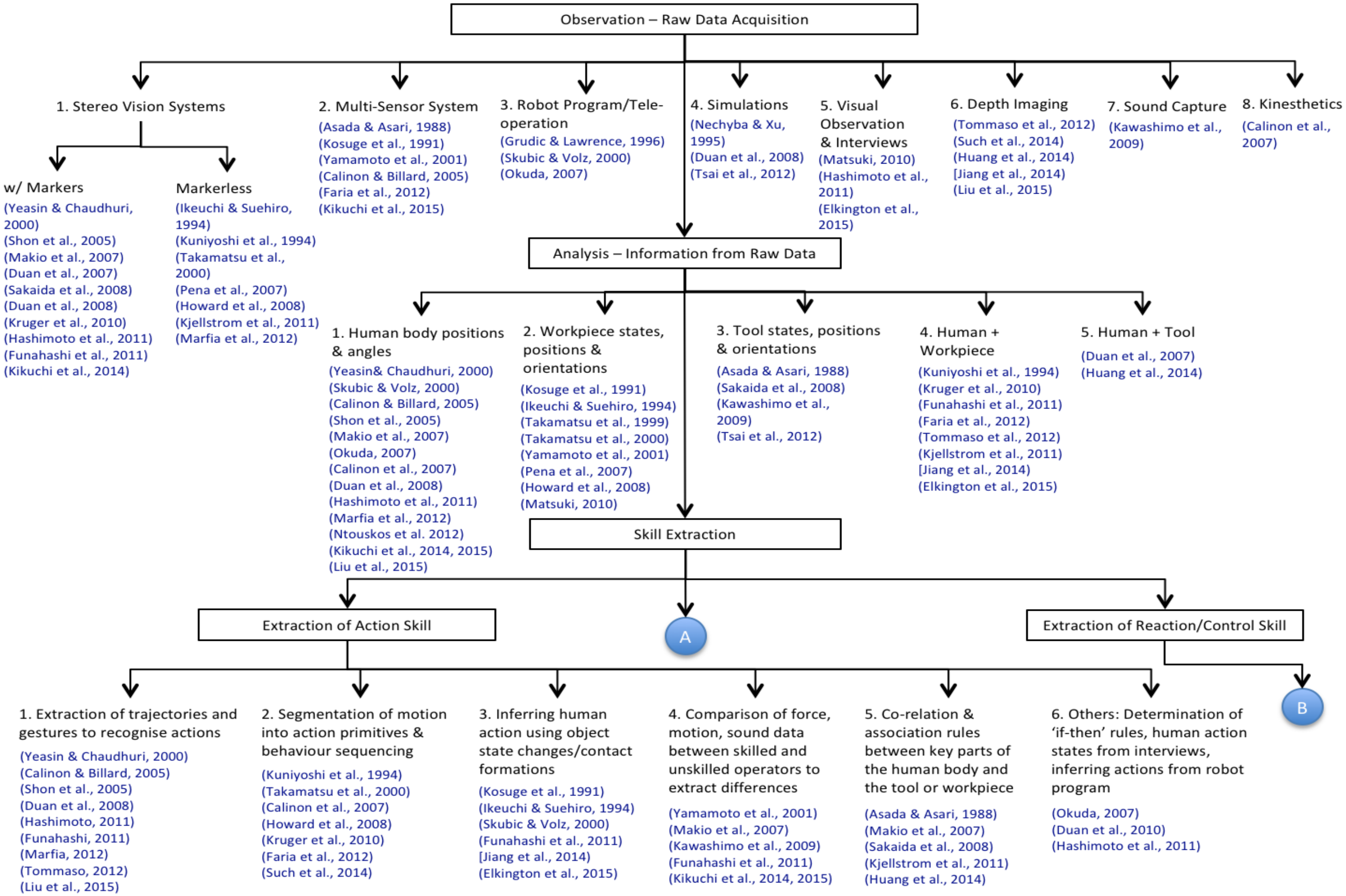


Figure 10: Technical-level classification of human skill digitisation process

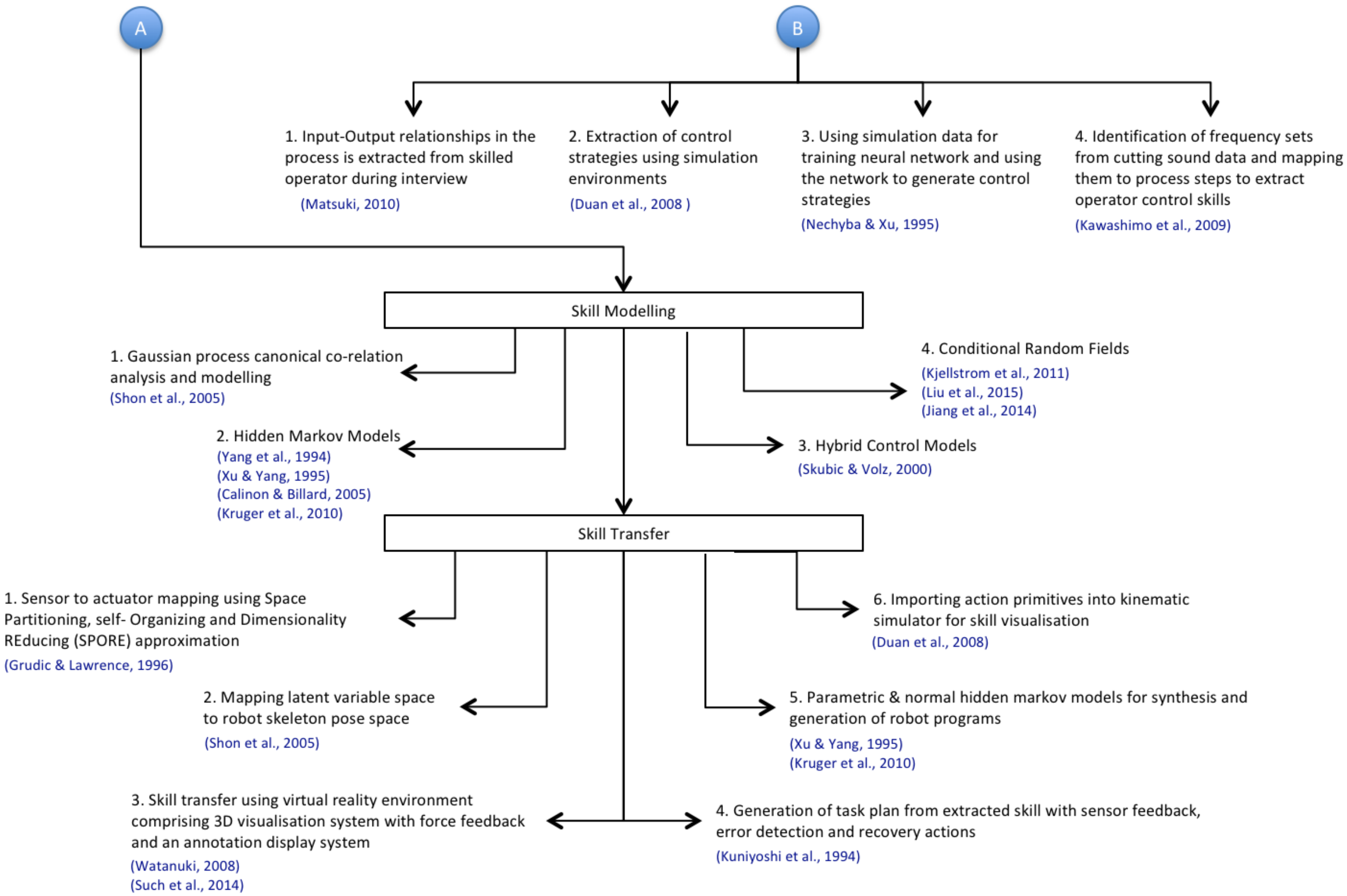


Figure 11: (Continued) Technical-level classification of the human skill digitisation process

Of all data acquisition methods, this is the most cumbersome and time-intensive. (vi) Low-cost gaming interface sensors, such as the Microsoft Kinect, that provide both Red-Green-Blue (RGB) and depth information of the captured scene and provide easy human motion capture are used in recent works to digitise human actions and recognise objects. Tomasso et al., (2012) and Huang et al., (2014) have used the Kinect to obtain real time information of human joint positions and angles during an operation. (vii) The analysis of audible data associated with manufacturing like the cutting sound in machining is vital to the decision-making skill of the human operator. Kawashimo et al. (2009) have used a microphone recorder to record cutting sound in a machining operation for subsequent frequency analysis. (viii) Finally, human motion data can also be acquired using Kinesthetics (Calinon, 2007), in which the human demonstrates the action by moving the passive arms of a humanoid robot and the motor encoders within the robot record this movement and joint angles.

2.3 Raw data analysis

The next step in the human skill digitisation process is the analysis of raw data. In this step, raw data associated with an entity or a combination of entities during the observation of manufacturing operation is specifically targeted and analysed. Five groups of methods have been reviewed. In the first group, human data is analysed to obtain a series of 3-dimensional (3D) coordinates of positions of the human body like the arms, torso and legs (Kikuchi et al., 2013, 2014) as well as the kinematics of human motion such as joint angle data (Duan et al., 2008). Ntouskos et al. (2012) report on various articulated 3D human motion analysis comprising compression, motion synthesis, indexing and classification. They also report their own work on statistical analysis of a diverse set of human action categories using publicly available motion capture (MOCAP) databases. In the second group, data associated with the workpiece are analysed to obtain its 3-dimensional motion, contact states with other workpieces, orientations in space and change in configurations as the manufacturing operation progresses from start to finish (Takamatsu et al., 2000, Yamamoto et al., 2001). The third group is specific to the analysis of data associated with the tool used in the manufacturing operation. From these methods, the tool's 3-dimensional motion coordinates, angles between its constituent parts and orientation in space are obtained (Tsai et al., 2012, Sakaida et al., 2008). In the fourth and fifth group, researchers have used a combination of two entities such as human-workpiece or human-tool to specifically obtain human body motion as well as object movements, states and orientations obtained from the analysis. An advantage of using a combination of entities for analysis is that the effect of the human actions on the transformation of the object (workpiece or tool) status from the starting conditions to the task goal is obtained (Jiang et al., 2014, Huang et al., 2014).

2.3.1 Skill extraction (Action skill)

The next step in the human skill digitisation process is skill extraction from the previously analysed data. More work has been reported to extract action skills as compared to reaction skills.

To extract action skills, researchers have identified specific trajectories traversed and gestures produced by the human body to recognise actions that have some effect on the tool or workpiece (Liu et al., 2015, Tommaso et al., 2012). Some researchers have segmented the motion data into action primitives, classifying these action primitives to specify specific sub-tasks and identifying and labelling the sequence and dependency of motion behaviours (Such et al., 2014, Kruger et al., 2010). Another method used is to infer human actions on the workpieces based on the changes in position, orientation and contact state formation when two or more workpieces come in contact with each other (Elkington et al., 2015, Funahashi et al., 2011). This method is usually used when only the workpiece is tracked during the operation or when human action is observed visually without any aid to record the motion. When the skill involved in an operation is not clear, comparison of analysis of motion and/or force data is made between the data extracted from a skilled operator and a non-skilled operator. The distinct difference in data is then inferred to be the change required to go from non-skilled to skilled for that operation (Kikuchi et al., 2013, 2015, Funahashi et al., 2011). Skill is also interpreted as the presence of specific association or co-relation rules that exist between different parts of the human body or tool or the workpiece during an operation (Kjellstrom et al., 2011, Huang et al., 2014, Sakaida et al., 2008). There are also some other methods used by researchers like producing 'if-then' rules based on the effect of human action on the tool or workpiece (Duan et al., 2010), extracting control strategies via interviews (Hashimoto et al., 2011) or inferring human action skills via a robot program written by a skilled human (Okuda, 2007).

2.3.2 Skill extraction (Reaction skill)

Reaction skill is considered high-level skill possessed by humans to make decisions and solve problems. It is acquired by way of experience and is difficult to extract using the methods stated in the previous paragraph since it does not involve any explicit motion. However, some researchers have attempted to extract reaction skill like Matsuki (2010), who used interviews to extract input to output relationships of an operation from a skilled operator. Duan et al. (2008) have used a dynamic simulator that simulates a real environment through which the human performs virtual manoeuvres and his/her corresponding control strategies are recorded. Nechyba and Xu (1995) have used simulation that a human uses to produce data, which is then used to feed and train the neural network. The neural network then produces control strategies depending on the network function set. Kawashimo et al. (2009) use analysis of frequencies of cutting sound recorded from a machining operation and map critical frequencies to process steps taken by the human operator in order to extract his/her decision-making skills in choosing process parameters.

2.3.3 Skill modelling

According to Xu and Yang (1995), skill modelling has the potential to enable automatic transfer of human skills to automation solutions like robots and create a skill library that can effectively be used by the robots in real-time operations. They have considered skill modelling and skill transfer in a stochastic framework and used Hidden Markov Models (HMM) to model and transfer both action and reaction skills. Human action is considered observable data as the output symbols of an HMM resulting from human mental states, which are not observable and considered as hidden states. The most likely set of human mental states responsible for a given set of human actions can be predicted from the HMM thus extracting human skill. Calinon and Billard (2005) have used principal component analysis (PCA) and independent component analysis (ICA) to pre-process human motion data to remove noise and encode the gestures using HMM so that the skill could be recognised, generalised and reproduced. Kruger et al. (2010) have enhanced the representation of human action by using parametric HMMs (PHMM) to map movement trajectories to their desired effect on the workpiece by taking the effect of the movement as a parameter. Shon et al. (2005) have used Scaled Gaussian Process Latent Variable Model (SGPLVM) to perform regression from high dimensional human motion data to low dimensional latent variable space to represent human action This way, the researchers have produced a learning model that uses imitation of humans. Skubic and Volz (2000) have used a hybrid control model, which provides a mechanism for combining velocity changes in motion and mapping of force control to human-workpiece contact formation events during an assembly task. Field et al. (2011) in their review of skill modelling for robotics have briefly described stochastic skill models like Gaussian Mixture Models (GMM) and HMM, transformational models like PCA, non-linear dimension reduction models and connectionist models.

Conditional Random Fields (CRF) is a commonly used modelling technique to recognise and model human actions from continuous motion sequences. CRF is a statistical modelling method often applied in pattern recognition and machine learning, where they are used for structured prediction. It is used to determine relationships between observations and construct consistent interpretations from sequential data and is an alternative to the related hidden Markov models (HMMs). However unlike HMM, CRF is limited because it cannot capture hidden-state variables and it assumes the human action sequences to be fully observable, which is not the case where skill heavily influences human actions. Researchers have proposed enhanced versions of the CRF such as Gaussian Process Latent CRF (Jiang et al., 2014) to build a probabilistic model that not only models human actions but also captures the relationships between human and other entities in the environment and Coupled Hidden CRF (Liu et al., 2015) in which the hidden low dimensional state variables are used for improved human action segmentation and classification.

2.3.4 Skill transfer

Skill transfer is the final step in which the skill extracted and sometimes modelled is passed on to novice operators for skill upgrading or to the automation solution that imitates human skill in the task. Skill is transferred using simple methods like generation of a task plan from extracted skill with sensor feedback, error detection and recovery actions (Kuniyoshi et al., 1994), and generation of robot programs from skill models (Xu and Yang, 1995, Kruger et al., 2010).

Duan et al. (2008) have entered their extracted human motor skill information into a kinematic simulator that synthesises motion and generates 3D coordinates and joint angle data for robotic arms to reproduce. However, due to the differences in degrees of freedom of the human joints and robot arm joints, the extracted motor skill information cannot be directly programmed into the robot arm. Depending on the lengths and proportion of the different parts of the robot arm, kinematic mapping is required to convert the human arm motion coordinates and angles to those of the robot arm. Also, certain movements that are not possible for the human, for example, backward bending of the arms, are possible for the robot arm. Therefore, a combination of human motor skill transfer and the use of special robot capabilities may be a practical solution for task automation.

Shon et al. (2005) have used SGPLVM to perform regression on low dimensional latent variable space in which human action skill was represented to high dimensional motion space representing degrees of freedom of the robot's motorised arms. Grudic and Lawrence (1996) have used Space Partitioning, Self-Organising and dimensionality Reducing (SPORE) approximation framework to generalise human action skill and map sensor to actuator outputs in order to transfer skill from human to robot. Watanuki (2008) has reported work on acquisition of manufacturing knowledge in casting by using Virtual Reality (VR) technology comprising a 3D visualisation and motion capture system using force feedback and an annotation display device. By adding annotation to the space in which a model is displayed in VR environment, technical information is shared between engineer and skilled operatives and embodied knowledge in the casting process is acquired. Such et al. (2014) use the Kinect to capture human hand motion during a composite layup task and depending on the observed motion and the pre-defined task strategy, annotate the next area of the workpiece that the human should be working on thereby transferring layup skill in real-time.

2.3.5 Recent advances in basic building blocks of this research

The basic building blocks that underpin this research are human action recognition, object recognition and human-object interaction tracking. Recent advances in this area have been identified to understand the current research landscape and the possibilities of future adoption of some of these advances.

Recent human action recognition research

Several articles have reported human action recognition using Red, Green and Blue (RGB) vision-based methods. Among the recent papers, Cheng et al., (2015) have used a combination of 'Bag-of-Words', in which the words are distinctive trajectory groups obtained using K-Means clustering, and the spatio-temporal relationships between the words to recognise human actions from RGB videos. Yoon and Kuijper (2013) have proposed a method to detect human actions by identifying and extracting skeletal features from the RGB images and using multiple kernel-based support vector machines for recognising actions. Rahman et al. (2013) have proposed a region-based method to recognise human actions by analysing the surrounding negative space regions of the human silhouette and Shao et al. (2012) have used temporal human action segmentation using methods based on colour intensity and motion gradients and action recognition using Pyramid Correlogram of Oriented Gradients (PCOG) shape descriptor. For research prior to 2011, the reader is referred to the survey of vision-based methods for action representation, segmentation and recognition by Weinland et al. (2011) and a review of human activity analysis, where an activity is a set of human actions, by Aggarwal and Ryoo (2011).

Recently, human action recognition using a combination of RGB vision and depth imaging (RGB-D) have gained in popularity due to the emergence of inexpensive gaming interface technologies that use RGB-D imaging, such as the Kinect. The Kinect and its associated Software Development Kit (SDK), provides up to 20 skeleton joints of a human in the 3D scene as well as the depth information of each pixel of that scene (Shotton et al., 2011). Chen et al. (2015) have used a fusion of RG-D imaging and wireless inertial sensors strapped to the body for recognition of human actions; Chaaraoui et al. (2014) have used an evolutionary algorithm to select the optimal set of joints identified by the RGB-D device to recognise human actions; Chen et al. (2013) have used spatio-temporal local feature representations to characterise and recognise human action from RGB-D images instead of relying on the skeletal joint data. For research prior to 2013, the reader is referred to a survey of human motion analysis using depth imagery by Chen et al. (2013).

Recent object recognition and tracking research

Object recognition has been a widely researched subject with applications in surveillance, medical imaging, social robotics and automation to name a few. Among the recent articles, Wohlhart and Lepetit (2015) have used the Euclidean distance between object descriptors, computed using Convolutional Neural Network (CNN), and the Nearest Neighbour search method to detect poorly textured objects and their 3D poses from RGB and RGB-D images, Zhang et al. (2014) have proposed a multiple kernel approach based on the Exact Euclidean Locality Sensitive Hashing (E2LSH) method of object detection; Yoon et al. (2013) have proposed a fuzzy particle filter algorithm to detect and track objects from a sequence of RGB images, Dou and Li (2013) have reported a moving object detection method based on improved Visual Background Extractor (VIBE) and graph cut optimisation from monocular video sequences. There are a few articles that report the use of RGB-D imaging methods for object recognition and tracking. Ali et al. (2013) have used both

RGB-D images to recognise object features using a combination of global appearance and shape-based feature vectors. Koo et al. (2013) have proposed multiple objects tracking from an RGB-D point set using GMM with a Tempo-Spatial Topological Graph (TSTG). Liu et al. (2013) have reported object segmentation from RGB-D data using a probabilistic boundary detector to detect object boundary and refine the boundary using Graph Cuts. Asif et al. (2013) have presented an approach to detect and track 6D pose of rigid objects from RGB-D image sequences using Oriented Brief (ORB) feature key points for object segmentation followed by feature extraction. For a detailed review of research prior to 2013 in this area, the reader is referred to a survey of approaches and methods adopted in literature for object recognition by Andreopoulos and Tsotsos (2013).

Recent human activity tracking and object affordance research

Social robotics applications, in which robots have to recognise objects in their environments and interact with them just as humans would, have opened up new approaches to identify and model mutual contexts in which humans interact with objects. In such research, human activities are recognised from a sequence of RGB or RGB-D images and by tracking changes to the objects that the human manipulates, affordances are assigned to those objects. Object affordance categorises an object based on its function and context of use, such as ‘throwability’ of a ball or ‘sittability’ of a chair. Hu et al., (2015) have used exemplar-based human-object interaction descriptors from RGB video data without the need to accurately obtain human pose estimation or object tracking data. Koppula et al. (2013) have proposed a method to extract human activities and object positions from RGB-D image sequence from which spatio-temporal co-relations between human and objects and between different objects can be extracted and object affordances can be assigned. The human activities and affordances are then modelled using Markov Random Field (MRF), the parameters of which are learnt using a Structural Support Vector Machine (S-SVM) formulation. Liu et al., (2013) have proposed a framework for human activities that manipulate objects. The framework does not rely on standard skeletal tracking library but uses local spatial statistics based algorithm to identify the human’s arms and torso. Based on the location of the arm, multi-class Support Vector Machine (SVM) object recogniser identifies the object held by the human and temporal super-segmentation is used to identify human activities responsible for manipulating the object. Ren and Sun (2013) have proposed a Human-Object-Object interaction affordance learning approach to model the inter-object affordance based on human hand motion trajectories and reaction of manipulated objects, and then used the inter-object affordance relationship to improve object recognition. Kjellstrom et al. (2011) have proposed a method to track human activity with objects in the environment to identify the affordances of these objects. Object feature and human action tracking is done by using SVM classification using object feature and human hand pose feature classifiers and human actions are co-related with objects based on both temporal and feature-level dependencies. A short review of methods used to recognise human-object interactions to establish semantics of human actions is provided by Ziaeefard and Bergevin (2015).

2.4 Focus Area II: Human-workpiece interactions

In order to analyse human-workpiece interactions, it is necessary to capture and digitise human motion during a manual manufacturing operation, generate human action states from this continuous motion data and identify the effect of those action states on the engineering workpiece/s in real-time. There is no reported work in literature that proposes the above approach to capture and analyse human-workpiece interactions for extracting and digitising human skills in manufacturing. There are several articles however that report human action recognition and object state detection, addressed as separate disconnected tasks. However, a few articles do report detection of human activities and the associated object states as connected tasks in order to extract/infer object affordances in a social environment. Most of these articles belong to the human-machine interface and social robotics domain, focusing on the image processing aspect of human and object detection, areas that are not within the remit of this research review; this research review focuses on capturing and analysing human-workpiece interactions for human skill extraction and modelling from a manufacturing task with a manufacturing perspective.

The study of human-workpiece interactions is conducted in this research as part of a body of work in literature known as ‘Cognitive Work/Task Analysis’ within the overall purview of ‘Human Activity/Task Analysis’ but applied to the field of manufacturing. There are 3 landmark theories reported in literature that investigate human behaviour, human problem solving and the human perception of task objects. These seminal theories are, namely, Rasmussen’s Skill-Rules-Knowledge (S-R-K) framework (Rasmussen, 1983), Rasmussen’s decision ladder concept (Rasmussen, 1980) and Gibson’s theory of object affordances (Gibson, 1979). Over the years and till date, these theories have been adopted and advanced for developing human-machine and human-computer interfaces, and assessment of human errors and industrial accidents in manufacturing systems. The theories are described below.

2.4.1 Rasmussen’s Skill-Rules-Knowledge (S-R-K) framework

Rasmussen developed a simplified human information-processing model called as the S-R-K framework (Rasmussen, 1983) to classify the performance of skilled human operators into skill-based, rule-based and knowledge-based behaviour (reproduced in Figure 12). This division is made based on the cognitive contribution of the human during the performance of the task.

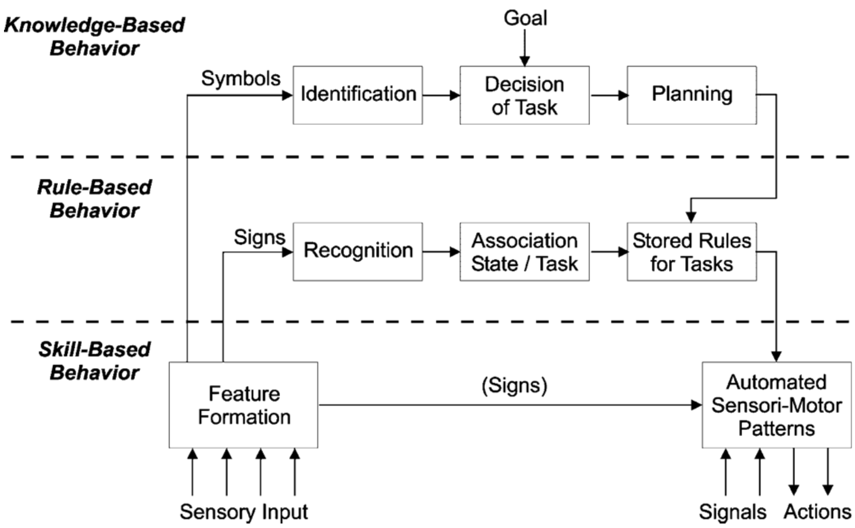


Figure 12: Rasmussen's S-R-K framework (Rasmussen, 1983)

At the skill-based behaviour, the human performs the task at the subconscious level implementing a sequence of sensorimotor processes without explicitly referring to any specific procedures or steps. The human actions in this task are smooth, automated, well-coordinated and display highly integrated patterns of behaviour based on a feed-forward control. Cycling and swimming are basic examples of skill-based behavioural tasks in which sensor feedback does not play a major role, since the human senses are too slow for direct feedback correction of rapid movements.

At the second level of rule-based behaviour, human performance is goal oriented and is guided by a set of specific rules and procedures. These rules may be obtained through training, instructions, derived from past experience, or developed by conscious problem solving or planning before the task. Rules are typically defined as 'if-then-else' clauses, and human operators would pick a specific rule and implement the associated procedure at any instance based on his/her perception of the system at that instance. Rasmussen makes a distinction between skill and rule-based behaviour by the level of training and attention a human requires for performing a certain task. Whereas skill-based performance does not necessitate a person's conscious attention and therefore cannot be easily documented, rule-based performance is generally based on explicit documented know-how.

The third and the highest level of behaviour is knowledge-based behaviour and is most commonly called for when new and unique problems arise within the system for which there is no experiential basis of best answers, procedures, or rules. This type of behaviour is observed when a human tries to manage unfamiliar situations and therefore has to rely on formal formulation, analysis and interpretation of the situation to evolve a plan to solve the problem. Therefore, the human must have a detailed and thorough knowledge of a system and/or process in a knowledge-based behaviour. These problem solving sessions when remembered and documented can be converted to rule-based behaviours when the human is faced with such situations again in the future.

The information observed by the human from the system in the different behavioural levels can be divided into 3 categories, namely, signals, signs, and symbols. At the skill-based level, sensed information is perceived as time–space signals that are subconsciously noted but otherwise have no explicit meaning or impact on the human's behaviour during the task. These signals are processed by the human as continuous variables. At the rule-based level, the human perceives the indicators of the state (or situation) of the system as signs. Signs are used to select or modify the stored rules, which in turn control the sequencing of skilled subroutines to perform the rule-based task. At the knowledge-based level, intelligent reasoning and the generation of new rules are based on system information perceived by the human as symbols. Symbols are defined by the internal conceptual representation of the system, which is the basis for intelligent reasoning and task planning. They represent variables, relations, and properties and can be formally processed and unlike signals, can be communicated to other humans.

2.4.2 Rasmussen's decision ladder

In the proposed human-workpiece interaction model, there are three main aspects of human involvement, namely, (a) observation of workpiece feedback, (b) making decisions about the actions to execute on the workpiece and (c) executing the chosen actions. Out of these 3 aspects, decision-making is probably the most critical especially for unstructured complex tasks that are manually performed. Such decision-making processes are naturalistic and intuitive rather than classical or analytical due to the highly dynamic and complex nature of the manual task not devoid of uncertainties. This is because the human is adept at making naturalistic decisions due to characteristics such as sophisticated cognition, memory of past experiences and mental simulation of the implementation of the decision to evaluate the outcome even before the action is executed. For structured and repetitive tasks, decision-making is simplified and therefore most such tasks are already automated.

To provide a frame of reference for the naturalistic and cognitive decision-making aspect of the proposed human-workpiece interaction model, Rasmussen’s Decision Ladder template (Rasmussen, 1980) is chosen. Though Rasmussen developed the decision ladder template (reproduced in Figure 13) to study the decision-making processes of experienced workers in operational settings such as thermal power stations, it can also be applied to manual manufacturing tasks. This is because, (a) both the environments have dynamically changing states, (b) human workers in both settings make naturalistic decisions based on observing and recognising the dynamically changing system states and (c) human workers in both settings rely on past experiences to solve new and unforeseen problems rather than using detailed analytical methods.

Rasmussen’s Decision Ladder’ template has several levels of information processing, flowing sequentially from the time the need for action is established to the execution of the chosen action. The main information processing activities are:

i. The detection of events that establish a need for action.
ii. The observation of the set of variables that allow the operator to identify the state of the system given the previous system state and the goals of the operation.
iii. The evaluation of the effects of those variables on the performance parameters of the system.
iv. The identification of a general plan, which will tend to correct the system.
v. A set of procedures to carry out the actions.

Every information processing activity guides the human decision making step that gives rise to a new state of knowledge of the system. This new state of knowledge acts as information to the next information processing activity and this sequence continues till the action is eventually performed to modify the system and solve the problem.

2.4.3 Gibson’s theory of object affordances

Gibson (1979) in his seminal work defined object affordances as action possibilities available to an individual for an object in an environment depending on his/her action capabilities. Gibson pointed out that, depending on the current behavioural goal, the same object could ‘afford’ different actions – a chair may be used for sitting, but it could also be used to stand up on and reach the top of a tall shelf if that was what was needed.

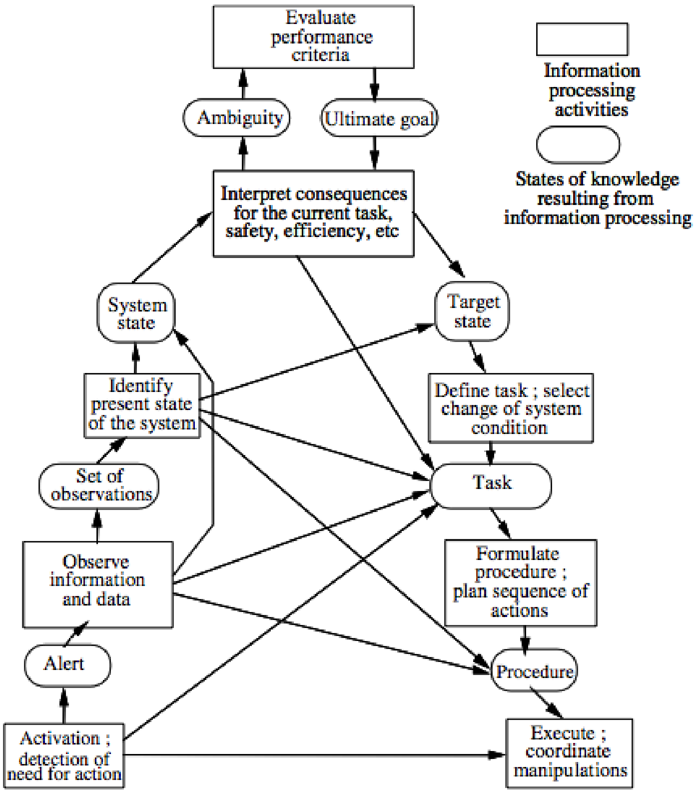


Figure 13: Rasmussen's decision ladder (Rasmussen, 1980)

Gibson claimed that objects were perceived in terms of these affordances for action. Therefore, Gibson used the theory of affordance to establish a relationship between a human and his/her environment and to state that the relationship is shaped by the sensory perception and the human action capabilities. According to Montesano et al. (2008), this relationship is the basis for humans being able to perform complex tasks by choosing appropriate actions from a vast repertoire to obtain the desired task results.

The concept of object affordances has been widely used in robotics though very little is known on how humans learn affordances. In robotics, affordances are used to capture the properties of an environment and the objects in that environment in terms of the actions the robot is able to perform. These actions are limited by the abilities of the robot. By observing humans perform their tasks in an environment, the affordances presented by the objects in that environment can be captured. For example, a ball affords the actions of catching and throwing whereas a cup affords the actions of grasping, drinking and releasing. These affordances can be used in robotics to predict the effects of an action performed by the robot in a human environment, to plan a series of actions to achieve a specific task goal, or to select objects that produce certain effects if acted upon in certain ways (Montesano, 2008).

It is evident from the above description that object affordances are used by the human to select the most appropriate action to perform on objects depending on the situation of the task and his/her own abilities/skills. These skills enable the human to perceive the characteristics of the object dynamically as they change during the task and to perform the necessary actions on the object to achieve the final task goal.

2.4.4 Human task analysis

Rasmussen's S-R-K framework and decision ladder have been applied to classical human task analysis and modelling techniques. Of all the techniques reported in literature, Hierarchical Task Analysis (Annett, 2003) and Cognitive Work Analysis (Vicente, 1999) are most widely referenced.

Hierarchical Task Analysis (HTA)

HTA defines a task as a set of goals, sub-ordinate goals, operations and plans; it focuses on what an operator is required to do, in terms of actions and/or cognitive processes to achieve a system goal (Kirwan and Ainsworth, 1992) along with a set of constraints present in the task environment. The 'plan' component of HTA is especially important since it specifies the sequence, and under what conditions, different sub-goals have to be achieved in order to satisfy the requirements of a main goal. In an HTA, data about the task is collected using techniques such as observation of skilled experts doing the tasks and interviewing them and then using this data to decompose and describe the goals and sub-goals involved (Salmon et al., 2010). The HTA procedure as presented by Stanton (2006) is shown in Figure 14.

However, according to Ezzedine et al. (2005), HTA may not be the most practical analysis technique for tasks involving complex human cognitive processes. Manual manufacturing tasks involving human skills are considered highly cognitive because the human relies on his/her intellect, knowledge and past experiences to take cognition of the various factors that influence the task such as the dynamically changing workpiece states, and chooses his/her actions based on this cognition.

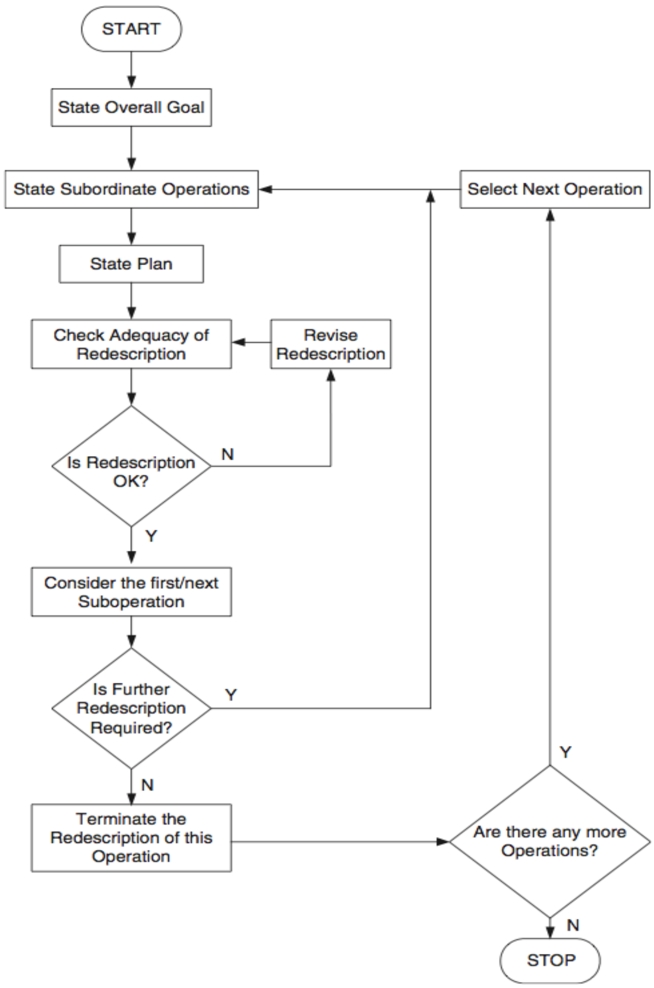


Figure 14: Hierarchical task analysis procedure (Stanton, 2006)

In HTA, a task description is limited to sequential performance of actions, which may not be the case in manual manufacturing tasks especially during problem-solving. Also, it is assumed that the information required to achieve the task goals and sub-goals are time and process invariant. This assumption is not valid in tasks with uncertainties where situations can change dynamically based on the effects of human actions on the task. Therefore in order to analyse and model complex dynamic tasks, Cognitive Work Analysis (CWA) is preferred.

Cognitive Work Analysis (CWA)

CWA was originally developed at the Risø National Laboratory in Denmark (Jenkins, 2008) for use within nuclear power process control applications with the need to design for new or unexpected situations such as industrial accidents and incidents. Though CWA is most suited for industrial work settings, such as process control, it has been used in many other domains as well (Hassall and Sanderson, 2012) and this research proposes to use some of its components to analyse and model human-workpiece interactions during a manual manufacturing task for digitisation of manufacturing task knowledge.

CWA is a multifaceted framework for performing work/task analysis, especially for analysing human information behaviour in complex systems open to disturbances. The approach works with constraints rather than goals, which is based on the notion that making constraints explicit in an interface can potentially enhance human performance during a task. CWA consists of different phases of analysis (Vicente, 1999) that focus on different classes of constraints within a complex task system. The phases are in the order in which the constraints on effective action logically flow. A brief description of the CWA phases is reproduced from Vicente (1999) in Figure 15 and described below.

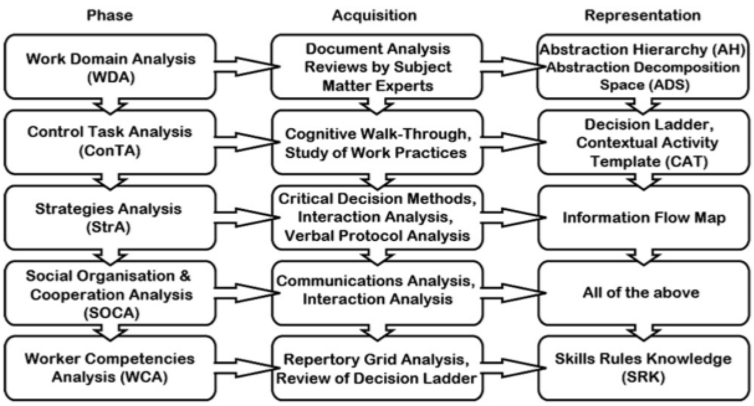


Figure 15: Cognitive work analysis phases (Jenkins, 2008)

Work Domain Analysis (WDA): WDA describes the schematic and structure of the task including purposes, priorities, functions, physical processes and physical objects at different levels of decomposition (Vicente, 1999).

Control Task Analysis (CTA): CTA describes requirements associated with human or machine interventions in the task that will help to achieve the purpose of the task such as human actions performed in different work situations (different physical or temporal work contexts) and human cognition of task information and decisions are taken based on the information (Vicente, 1999).

Strategies Analysis (SA): SA identifies the physical action and decision control strategies that humans use to perform tasks in the work domain. According to Rasmussen, SA should (1) identify the different factors that may influence the strategies, (2) describe strategies as 'generic' categories of cognitive processes, (3) identify the criteria used to select one category of cognitive processes over other possibilities and (4) identify the cues that prompt the selection or change in strategy (Vicente, 1999, Hassall and Sanderson, 2012).

Social Organisation and Cooperation Analysis (SOCA): SOCA determines which agent (human, machine or both) is best placed to perform each intervention. It is used to identify how the actions and the associated strategies required can be distributed amongst human operators and technological artefacts within the system in question and also how these agents could communicate and cooperate (Vicente, 1999).

Worker Competencies Analysis (WCA): WCA identifies the knowledge, rules and skills that workers need to successfully perform the work allocated to them. Its usual outcome is an analysis of how humans might perform control tasks in knowledge, rule or skill based manner, via annotations on a decision ladder.

According to Salmon et al. (2010), due to its flexibility and the varying perspectives on complex systems, the CWA framework has been applied in various complex domains for a number of different purposes, including system design and modelling, machine interface design and evaluation and the development of human performance measures. These applications have taken place in a variety of complex safety critical domains, including air traffic control, health care, nuclear power, manufacturing, military command and control, petrochemical process control and transport systems.

A review of CWA research is conducted and reported in literature by Hassall and Sanderson (2012). From this survey it is evident that most reports use Work Domain Analysis, with fewer reports using Control Task Analysis, and even fewer reports using Strategies Analysis, Social Organisation and Cooperation Analysis or Worker Competencies Analysis. One possible reason for this is that there is a lack of satisfactory methods that integrate two or more of the CWA phases successfully. The biggest advantage of combining the phases into an integrated analysis is that a single task information acquisition system can be used instead of multiple ones, one for each CWA phase.

The different CWA phases allow the analysis of constraints related to the domain within which the activity is conducted (WDA), what activity is conducted (CTA), how the activity is conducted (SA and WCA) and whom the activity is conducted with (SOCA). In practice, the purpose and context of an investigation of cognitive tasks will determine whether all five phases are used and the order in which the analyses are done.

2.5 Chapter Summary

Skill intensive manual manufacturing tasks form a significant part of the high value manufacturing industry. It is getting increasingly difficult for companies to sustain such operations because of the scarcity and high cost of skilled labour as well as the market needs for higher production speeds and improved quality amidst tough global competition. Therefore, there is an urgent need for companies to look for ways to automate skill-intensive manufacturing operations to reap the benefits of automation or transfer existing manpower skills to new recruits to maintain in-house skill competency. However, human skill being tacit in nature is difficult to document and hence difficult to transfer from one human to another or to an automation solution. Therefore, important research is being conducted in the area of human skill digitisation and therefore all the major methods, tools and techniques used in the manufacturing task knowledge digitisation process are stated, classified and discussed in this chapter.

In the process-level classification (Figure 9), human skill digitisation process can be segregated into 5 distinct steps. It is clear from Figure 10 that most research attention is given to the first three steps of data acquisition by observation, data analysis and skill extraction with the last two steps being reported by only a handful of articles. This phenomenon is not surprising given that data acquisition and analysis are critical steps with multiple technological solutions available, each having its own pros and cons depending on the nature of the manufacturing operation, which in itself is vastly varied.

Skill extraction again takes various forms based on the nature of operation. For example, an operation where human dexterity and motion is vital, extraction of trajectories and gestures is required, whereas for complex assemblies, workpiece contact formations and states along with human action primitives is required to be extracted and finally when decision making skills are involved, human control strategies derived from cause and effect rules or human action corresponding to simulated conditions are extracted.

Skill modelling is required when human skill used for one operation is required to be generalised so that it could also be utilised in automating other operations. Modelling is also required to reproduce a learnt skill performed in a structured environment to an operation to be automated in an unstructured environment.

The skill transfer step is commonly a by-product of the skill extraction step where the motion trajectories, action primitives, and control strategies are extracted. From this data, information such as a list of motion coordinates sub-tasks and hierarchies, movement association rules, and ‘if-then’ rules that can be directly applied to automation solutions as inputs are extracted. Therefore, the need for a separate skill transfer step is not compelling in most cases. However, in situations where specific mapping is required between sensors that capture the environment and actuators that run the automation, or robot programs are generated directly from skill models, or skill is visualised using a simulator after being extracted, skill transfer is a necessary step.

In the technical-level classification (Figure 10), the methods, tools and techniques used by researchers in each of the five steps of the manufacturing task digitisation process is presented. Data acquisition is commonly done using vision systems or a combination of systems using vision, force and motion sensing devices. These systems are most adept at observing human actions and interactions with the tool and the workpiece due to their relatively broad catchment space, real-time data acquisition and an abundance of technologies and products available to choose from. Stereovision systems, both with and without markers are a popular choice with more researchers opting for marker-based systems. The advantage of marker-based systems is that the parts of the human body and objects to be tracked can be distinguished easily from the rest of the scene without the need for complex image processing algorithms. The disadvantage however is that this system is obtrusive in nature and the markers can hinder the human operator trying to demonstrate the operation in a real factory environment. Multi device systems also suffer from the same disadvantage with the human wearing force, motion and tactile sensors. Markerless systems are unobtrusive in nature but require complex computing for image processing the data to recognise targeted motion. With faster computer speeds, markerless single camera systems could gain significant traction and see widespread use for skill extraction in the future.

Using interviews for data acquisition, especially for skill information is not as effective as other techniques because of the very tacit nature of this information and its dependency on the communication skills of both the interviewer and the interviewee. Simulations are a good way of gathering data due to the ease with which various parameters and conditions of a manufacturing operation can be set and tweaked without additional costs and efforts.

Depth capture by using infra-red cameras such as the Kinect shows significant promise because of its popularity in human motion and gesture tracking for gaming applications. Due to its low cost, robust technology, markerless nature, anonymity in motion capture, and availability of mature software libraries for motion data acquisition and analysis, depth imagery could easily be extended to observe manual manufacturing operations in actual factory environments. A detailed review of different motion capture technologies is presented by Field et al. (2011) and Chen et al. (2013) provide a recent survey of human motion capture and analysis using depth imagery.

In data analysis, human motion is analysed the most, followed by workpiece and tool motion and states. A manufacturing task as mentioned in section 2.1 is an interplay between the human, the tool and the workpiece and this interaction over the period of the manufacturing operation results in a successful product. Therefore, by analysing a combination of human, tool and workpiece data, not only key characteristics of individual motion but also the interdependencies could be identified and extracted. Some attempts have been made to combine human-workpiece and human tool analysis but no work was found that analysed all the 3 entities together.

In the skill extraction step, action skill has received a lot more attention than reaction skill. This could be because a lot of skill-intensive manufacturing operations involve complex human gestures to machine, inspect or assemble a product. Also, reaction skill, though being abstract in nature, can be extracted using simpler methods of interviews and simulations, and can be documented as human control strategies for simpler operations. For complex operations where dynamic problem solving is involved, it is difficult to extract reaction skills currently. In action skills extraction for simple operations, identification of human motion trajectories and gestures is sufficient whereas for complex operations, segmentation of motion into action and behavioural primitives and sequencing of behaviours into hierarchies is required. Identification and extraction of object contact formations and states is used for assembly operations and is currently limited to simple objects with regular geometrical edges. The method of comparing motion data of a skilled operator with an unskilled operator and extracting the difference as identification of skill is useful for training novice operators and may not be useful in enabling automation of the operation.

In skill modelling, the stochastic technique of HMM was one of the first few models used to represent human motion data and is still relevant because of its ability to model different aspects of human skill even if the observed and captured data is not perfect. The parametric or adaptive forms of HMM that recognise effects of human movements and generate new trajectories having the same effect on objects is gaining interest. This feature could enhance the generalisation of skills and their adaptation to unstructured environments.

In skill transfer, generation of robot programs from skill models, task plans and control strategies from neural networks and mapped sensor to actuator outputs form direct inputs to the automation solution whereas kinematic simulation of extracted skills and documented differences between skilled and unskilled operators can be used to enhance skill competency of new operators. In the most basic form of skill training, animations can be used to graphically render how a particular manual manufacturing task should be done by overlaying the manufacturing knowledge extracted on top of the animation. Virtual and augmented reality can also be used to transfer skills from experts to novices by using the medium of demonstration in an immersive 3D visualisation space.

Finally, CWA within the ambit of HTA is a preferred method of analysing complex manual manufacturing tasks. Within CWA, it is particularly important to have integrated analysis of all five phases to understand the interdependencies of strategy, control, worker competency, cooperation and task structure to extract the manufacturing knowledge embedded within the task.

2.5.1 Research trends

Human skill transfer is an important process in today's manufacturing industry. Important research is evident across all aspects of the process though some facets will see more growth than others. Human motion capture to record manual manufacturing operations for example may see an exponential increase in research with newer technologies like depth imaging devices, muscle-control-sensing devices, wireless accelerometer-based motion tracking systems and eye tracking devices being introduced with increased portability and reduced cost. Over the last 14 years, a steady increase can be seen in the number of articles published in human motion capture and action recognition research (Figure 16). This trend is likely to continue.

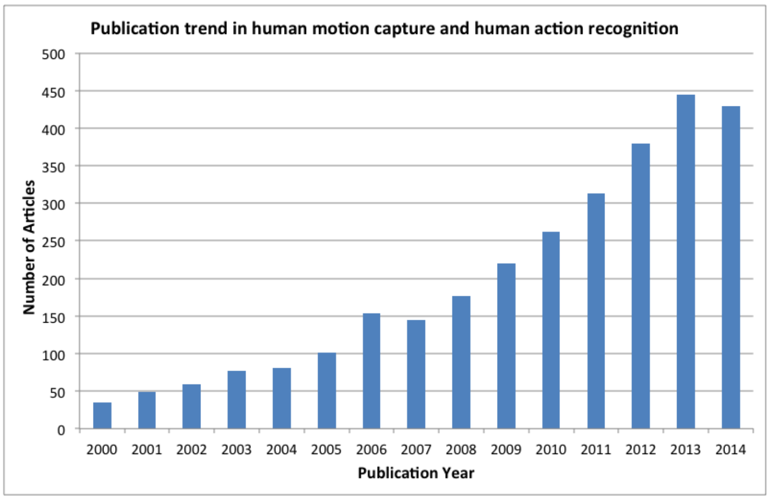


Figure 16: Human motion capture and action recognition research (2000-2014)

Object recognition and tracking has been a popular topic of research over the past several decades with applications in machine learning, industrial and social robotics, surveillance, defence, etc. With the advent of portable depth imaging sensors such as the Kinect™, object recognition has become easier due to the availability of the depth information as against the complex computation required earlier with only RGB information available. Research in object recognition and tracking is at a considerably greater scale than human motion capture and is steadily growing year on year except for a dip in the year 2010 (Figure 17). Again, this upward trend is likely to continue.

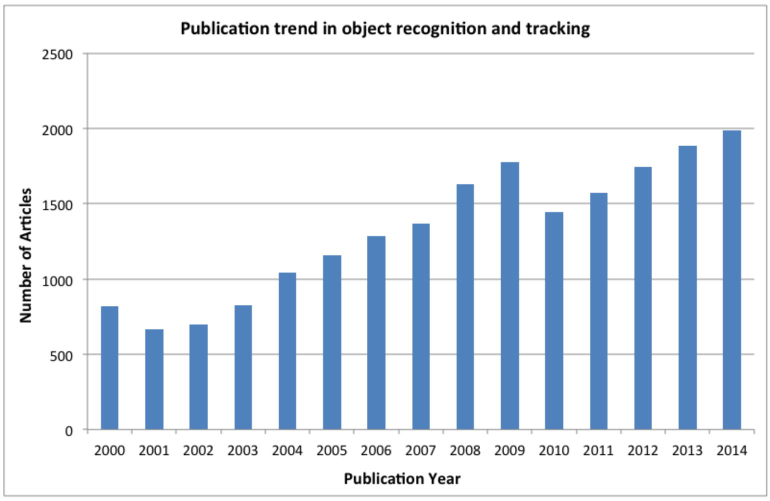


Figure 17: Object recognition and tracking research (2000-2014)

Human action recognition and object recognition if done simultaneously can provide an insight into the interactions between the human and the objects during a task. Such interaction studies are conducted predominantly by researchers in the social robotics area where it is important to understand human behaviour with objects so that the same behaviour could be imparted to robots for human robot co-existence. Research is also on going to identify object affordances so that robots can be given the intelligence to choose the correct actions that could be taken on objects as they interact with the human environment. In human-object interaction research, there has been a gradual increase in the number of articles published over the past 14 years (Figure 18) and this trend is likely to continue for diverse applications because the technical difficulty in simultaneous human and object tracking is decreasing with the emergence of low-cost depth imaging technologies.

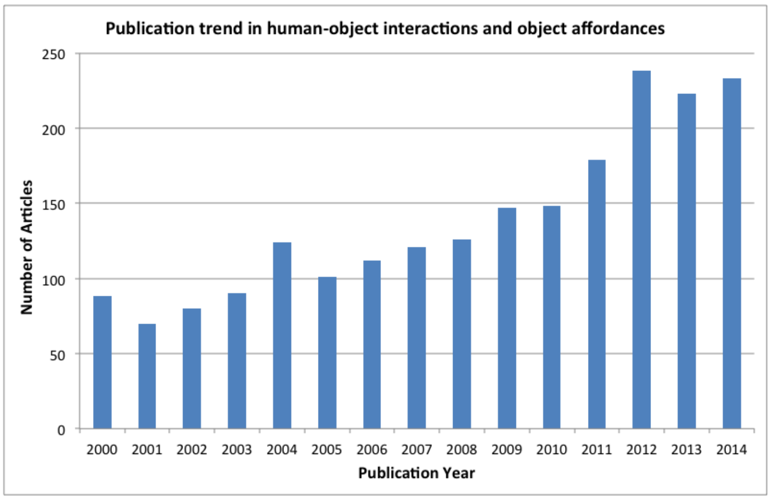


Figure 18: Human-object interaction research (2000-2014)

With the advent of the Microsoft Kinect™ in 2010, the first commodity priced gaming interface device, human-centric research across multiple domains such as engineering, computer science and health science has got access to inexpensive, portable, robust and markerless motion capture technology. Research on or using the depth imaging technology offered by devices such as the Kinect™ has exponentially grown in the past 5 years (Figure 19) and this trend will continue as the next generation of these devices that are better than the first generation are made available to satiate the ever increasing appetite of the gaming industry for improved human interface technologies.

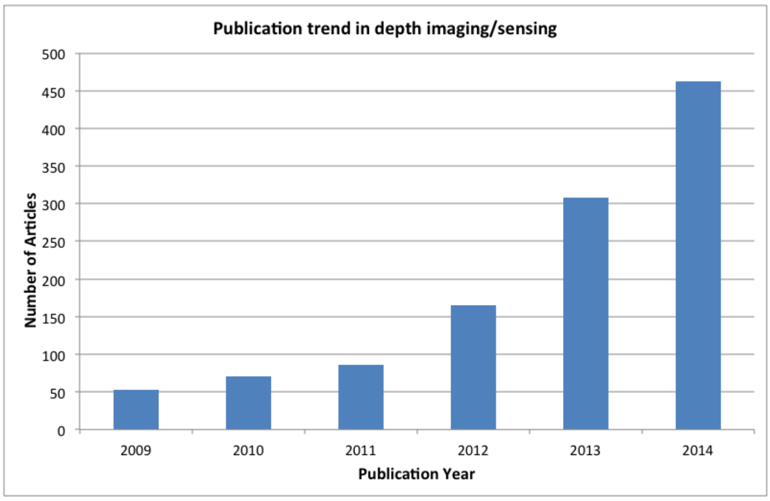


Figure 19: Depth imaging/sensing research (2009-2014)

Though not much work can be seen in using depth imaging technologies to extract human skills, these technologies may overtake the others such as stereo-vision systems due to their ability to simultaneously track human, tool and workpiece without any dependence on ambient light conditions and to capture human and workpiece motion without the need for markers. However, a bottleneck in these depth-imaging devices is the complexity of image processing code required to compensate for the lack of adequate tracking accuracy. This challenge would be overcome with advances in depth imaging and motion tracking technology, brought about by the gaming industry.

Increasingly, dynamic problem solving, which skilled humans are adept at, is also required to be a feature of automation. Therefore, more research may be seen in the area of extraction of action and reaction skills in a complete skill digitisation process. For this purpose, simultaneous, multi-modal data acquisition combining vision, sound, inertia and force may see increased use. Skill modelling using HMM and its adaptive versions like parametric HMMs will continue to be used for their versatility and stability in representing and modelling human skills.

A major application of skill digitisation is to develop a skill library and use it for training and skill upgrading programmes of companies to maintain their skill competency levels. However, with increased labour costs and developments in designing bespoke automation solutions incorporated with extracted human skill, a shift towards intelligent automation of skill-intensive manufacturing operations may be witnessed. A unified framework for intelligent automation with embedded human skills is another area that has not been explored fully.

2.5.2 Research gaps

As a result of this review, five major research gaps (RGs) have been identified that this research proposes to bridge, which could lead to the development of a framework for digitising manual manufacturing task knowledge.

[RG1] There is a lack of representation of manual manufacturing operations as interactions between human and workpiece. A lot of research has been previously focusing on each of these 2 entities as isolated entities but their interdependence on one another over the duration of the manufacturing task has not been investigated effectively. The effect of human action and reaction on the workpiece during the task can be significant because of the skill-intensive nature of the manual task. Therefore, by observing and digitising these interactions, human action and reaction states could be identified and mapped to workpiece states along the entire duration of the operation.

[RG2] There is no established framework reported in literature or used in the industry that cohesively captures, extracts, decodes and stores manufacturing knowledge, especially human skill, from manual manufacturing tasks. Knowledge acquisition and modelling for operational level and management level decision-making is possible using Knowledge Based Systems (KBS) but at the task or process level, there is no framework reported.

[RG3] There is a lack of simultaneous, multi-modal data acquisition methods like RGB imaging, depth imaging and sound/voice capture for observing and analysing a manual manufacturing task in a single digitisation process to enable extraction of both human action and reaction skills. While image processing can only provide human action or motion data, extracting human reaction skills would require a combination of multiple modes of task capture. Another possibility of using multiple depth and RGB imaging devices to capture different aspects of the same manufacturing operation to get multi-perspective data is also not investigated enough in literature.

[RG4] Stochastic modelling tools, such as Hidden Markov Modelling (HMM), have been used to model human motion, extract gestures from motion and map human gestures to the thought process or mental states. HMM still remains a popular method for identifying patterns in time-sequence data such as for human activity detection or speech recognition applications. However, stochastic modelling has not been used to represent human-workpiece interactions in order to map human action states to corresponding workpiece states. This interplay of human action, human decision-making and resulting workpiece states modelled using HMM is not reported in literature.

[RG5] Finally, the study of human-workpiece interactions is conducted in this research as part of a body of work in literature known as 'Cognitive Work/Task Analysis' within the overall purview of 'Human Task Analysis' but applied to the field of manufacturing. Human task analysis research reported in literature and studied in this review has been applied at the operational level of a manufacturing plant rather than at an individual task level. There are no references in literature that have proposed the advancement and application of these seminal theories in the digitisation of manual manufacturing task knowledge.

CHAPTER 3

3 RESEARCH AIM, OBJECTIVES AND METHODOLOGY

This chapter presents the research aim and objectives based on the hypothesis of this research and informed by the research gaps identified from the literature review. Following that, the scope is clarified and methodology for conducting this research is explained. This chapter aims to achieve the following goals:

- ❖ State the research hypothesis.
- ❖ State the research aim.
- ❖ Outline the research objectives.
- ❖ Clarify the research scope.
- ❖ Explain the research methodology.

3.1 Research Hypothesis

The hypothesis of this research is as follows:

'*By simultaneously capturing human actions and the effects of those actions on the workpiece in real-time for a manual manufacturing task, it should be possible to extract, decode and digitise the manufacturing knowledge associated with the task. This knowledge mainly comprises expert human gestures to execute the task and deft human responses to unexpected problems in the task.*'

Once digitised, this knowledge can be used to (a) impart skill training to other humans, (b) study manufacturing tasks to analyse ergonomic correctness, human errors and industrial accidents, and (c) develop digital skill models to power the next generation of automation solutions to completely replace skilled manual tasks.

Based on the above hypothesis, the research aim and objectives are as follows:

3.2 Research Aim

The aim of this research is to develop a framework for digitisation of manual manufacturing task knowledge. This research uses gaming interface technology as a reliable and cost-effective means to generate human-workpiece interactions data from which manual manufacturing task knowledge is extracted and decoded using stochastic machine learning.

3.3 Research objectives

In order to achieve the research aim, the specific research activities are distributed into six main objectives:

i. To develop a human-workpiece interaction theory to comprehend manual manufacturing tasks and to provide the theoretical underpinning for development of the proposed digitisation framework.
ii. To design and develop a framework for digitisation of manual manufacturing task knowledge and stochastic machine learning.

iii. To develop a method to capture and record human-workpiece interactions using gaming interface technology
iv. To develop a method to extract and decode manufacturing task knowledge using stochastic machine learning.
v. To develop a method to reproduce the decoded manufacturing task knowledge using animation.
vi. To implement, test and validate the framework with simplified, lab-scale and real-life manual manufacturing tasks.

3.4 Research scope

This research investigates whether a framework for digitisation of manual manufacturing task knowledge can be conceptualised and developed. The review of literature revealed that though a large body of work exists that investigates human activity recognition and object recognition for applications such as social robotics, surveillance, human-robot interactions, etc., there is no reported work that establishes a complete framework that uses human-workpiece interaction tracking and modelling to extract and decode manual manufacturing task knowledge. There are a few articles, especially from Japan that have reported their research in human skill acquisition and transfer with application in manufacturing but none have proposed a systematic structure for skill acquisition and transfer. Therefore, this research focuses on the development of a cohesive digitisation framework for digitisation of manual manufacturing tasks. The research also aims to use consumer-grade, low-cost gaming interface technology as the primary digitisation tool. The scopes of individual aspects of this research are briefly outlined below:

Manufacturing knowledge: In this research, 3 main constituents of manufacturing knowledge are extracted and decoded from within a manual manufacturing task:

1. The nature of human actions and spatial characteristics of human motion during those actions. This constituent contributes to understanding and digitising the human motor skills used during the task.

2. Spatio-temporal relationships between human actions and the workpiece/s during the task. This constituent illustrates the effects of different human actions on the workpiece/s at any given task instance.
3. The choice of actions and the sequence in which to implement those actions during the task. This constituent contributes to digitising the human reaction skills, especially useful in knowing which actions to execute when (planned and implemented task strategy) during the task.

The manufacturing knowledge constituents that are not within the scope of this research relate to the physical parameters of a manual manufacturing task such as force, torque and human strength as well as to the tactile and audible feedback provided by the workpiece in response to human actions during the task. Also, extraction of sub-conscious decision making ability of the human in response to continuous changes he/she observes in the workpiece/s during the task are outside the scope of this research.

Human-Workpiece interaction model: This research focuses on the development of a human-workpiece interaction theory with inputs from Rasmussen's S-R-K framework, Rasmussen's decision ladder concept and Gibson's theory of object affordances. The proposed theory is expected to be generic enough to represent all the 3 major categories of manufacturing tasks, namely, machining, assembly and inspection.

Proposed digitisation framework: This research proposes a framework for digitisation of manual manufacturing task knowledge enabled by ICT methods and tools obtained off-the-shelf and developing bespoke solutions whenever required. The proposed framework is expected to be generic enough to be used to digitise manual tasks belonging to all the 3 major manufacturing categories, namely, machining, assembly and inspection.

ICT methods and tools: ICT methods and tools, including gaming interface technologies are used to implement the framework. Of the several gaming interface technologies available, human motion capture sensors, such as the Microsoft Kinect™, are proposed. Apart from using the standard motion capture feature, this research extends the functionality of these sensors by using their colour and depth imaging capability to recognise and track moving and changing objects within a manual task. The rest of the ICT methods and tools such as motion data filtering and smoothing algorithms, data segmentation algorithms, and stochastic machine learning algorithms are used in their fundamental form from literature. For example, hidden Markov modelling is used as the stochastic machine learning tool and is adopted in its fundamental form in the framework.

Validation case studies: The case studies are manufacturing task examples for implementing, testing and validating the proposed digitisation framework. These tasks are a mix of simplified and actual task examples that contain a variety of workpieces and human actions in varying degrees of complexity. The case studies begin with simplified tasks, such as Lego block assemblies, before progressing to complex ones, such as composite fibre layups. The gradual increase in task complexity is designed to deal with the uncertainties of using gaming interface sensors that are proposed for digitising manufacturing tasks for the first time with no existing benchmarks in literature.

3.5 Research strategy

Two main types of research strategies exist, namely, fixed design strategy and flexible design strategy (Robson, 2002). Fixed design strategy enables structured experimentation with a pre-defined input and output parameter space under controlled conditions and a well-defined boundary of exploration to reach a conclusion. Research governed by this strategy is quantitative and takes a traditional scientific approach, which is pervasive and objective, and strives for reliability as well as reproducibility of results. However, quantitative research can sometimes suffer from being constricted to a limited exploration space when the solutions desired could be outside of this space. Much of engineering research tends to adopt the quantitative approach (Burns, 2000). Flexible design strategy enables an investigative approach in which the parameter space and the exploration boundary keep evolving with changing research contexts to reach a conclusion. Research governed by this strategy is qualitative and usually involves interviews, surveys and observations for data collection. The main advantage of the qualitative approach is that the research framework and direction can be quickly revised as new information emerges enabling the researcher to cover a wider exploration area for in-depth inquiry. However, qualitative research can sometimes suffer from heavy dependence on the individual skills of the researcher and influence of the researcher's personal biases and idiosyncrasies resulting in in-valid generalities and subjective conclusions. Much of social and business management research tends to adopt the qualitative approach (Gummesson, 1991).

A fixed design strategy is used in this research to develop, implement and test the proposed digitisation framework. The strategy is theory-driven with a well-defined hypothesis and is implemented using the quantitative approach with structured experimentation under controlled laboratory conditions and numerical data analysis to draw an objective conclusion. Experimentation also enables investigation of the effect of several variables at different functional levels on the performance of the framework thereby validating its applicability and efficacy. The resulting research methodology is presented below.

3.6 Research methodology

The research methodology followed is summarised in Figure 20. This figure also maps each step of the methodology to the corresponding thesis chapter.

3.6.1 Identification of research problem

This research is in the broad area of manufacturing informatics with a focus on providing deeper insights into manual manufacturing tasks with the purpose of extracting and decoding knowledge from those tasks. The problem addressed by this research is the lack of a cohesive framework for the capture, extraction, decoding and transfer, in short digitisation, of manual manufacturing task knowledge, especially hidden knowledge such as human skill.

3.6.2 Literature review

The literature review is conducted by examining the peer-reviewed journal and conference articles, thesis/dissertations, book chapters and web pages. It begins with an overview of the approaches used to digitise manual manufacturing task knowledge to understand the need, nature and importance of the research problem. Next, the methods used in the reported approaches are reviewed to identify the pros and cons of each and to find the gaps that could be filled by this research.

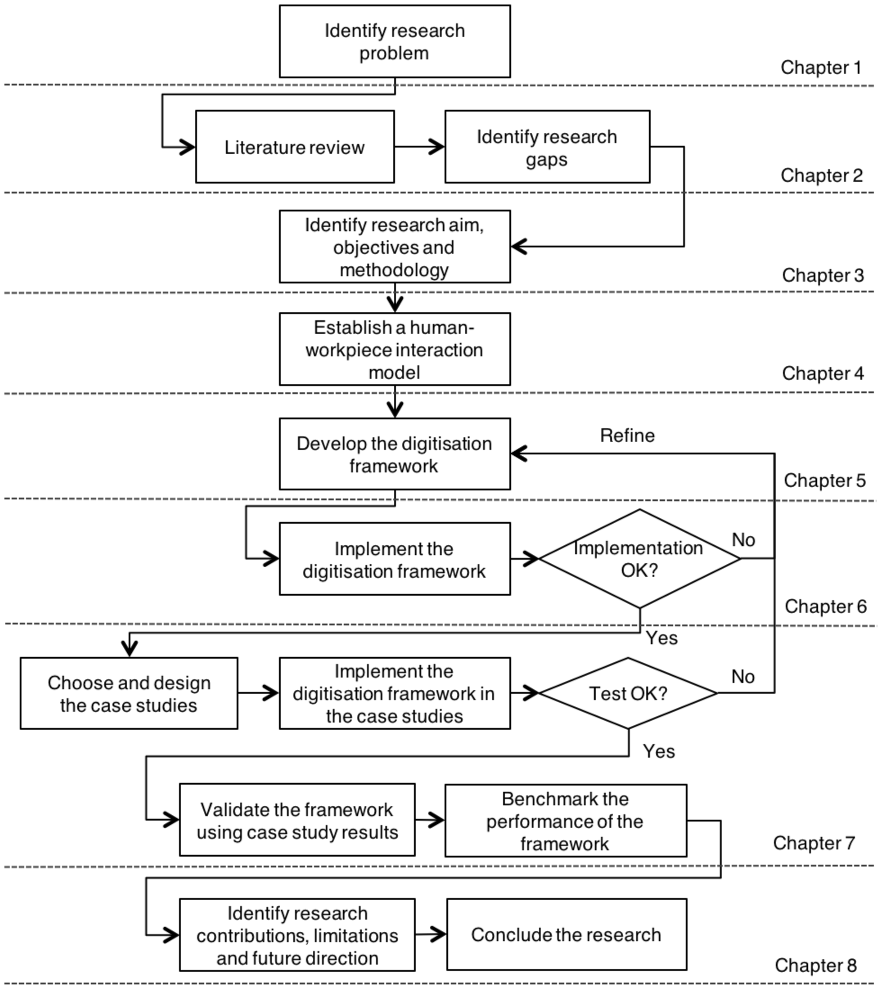


Figure 20: Research methodology and corresponding thesis chapter mapping

Simultaneously, the methods used in literature that apply to the research problem but are applied for different problems are also reviewed to identify their relevance in this research and their future trends. At this stage, the research activity is made up of two main tasks:

i. **Survey of literature**

An extensive literature survey is performed to identify the research trends and potential in manufacturing knowledge digitisation. It starts with identification of keywords in manufacturing knowledge digitisation such as ‘human motion/action capture’, ‘object/workpiece recognition and tracking’, ‘human-object interactions’, ‘human skill capture/ extraction/ acquisition/ digitisation’ and ‘knowledge capture/ extraction/ acquisition/ digitisation in manufacturing’. Based on the keywords, an extensive search of databases is conducted. The primary databases used for this search are Scopus and Science Direct. Google Scholar was used to get access to some papers that were not available in the databases. The articles are then filtered to select only relevant papers by screening the abstracts. The selected papers are fully reviewed and analysed to gain a clearer picture of the relevant research landscape.

ii. **Identification of research gaps**

The research gaps in the area of digitisation of manufacturing knowledge digitisation are identified from the above literature survey.

3.6.3 Identification of research aim, objectives and scope

The research aim is identified based on the research problem. The research objectives are developed according to the research gaps found in literature. This ensures that the proposed research is aligned with the current research trends. The research aim is specifically formulated to solve the research problem and is broken down into specific objectives to approach the research as a phased activity. The research scope explains the technical and practical boundaries within which the research is conducted.

3.6.4 Establishment of a human-workpiece interaction theory

In this research, there is a need to conceptualise and develop a basic human-workpiece interaction theory that could provide a strong foundation to build the framework for digitisation of manual manufacturing task knowledge because of a lack of such a theory in literature. However, there are seminal theories related to human behaviour in manufacturing environments that could be complementary to this research and therefore those theories are used to reinforce the human-workpiece interaction theory.

3.6.5 Development of the proposed digitisation framework

The framework for digitisation of manual manufacturing task knowledge is developed to fulfil the need for a cohesive methodology to capture, model, extract and decode manufacturing knowledge from manual manufacturing tasks. The approach taken is to identify the key steps in informatics data processing from literature and to bring these steps together in a logical sequence to implement key functions of the digitisation process. Each function is detailed with the relevant inputs, outputs, methods and dependencies.

3.6.6 Implementation of the proposed digitisation framework

The framework is proposed to be implemented using a simplified lab-scale assembly task. Each function of the framework is applied according to the methods, tools and dependencies identified and the outcome of one function is the input to the next in the designed sequence. The main goal in this stage is to evaluate the comprehensiveness of the framework to digitise manufacturing task knowledge and identify major problems or obstacles in the implementation of the framework. The issues identified in the evaluation are fed back to the development stage for the framework to be refined accordingly.

3.6.7 Validation of the framework using case studies

Once the proposed framework delivers its main goal of digitising manual manufacturing task knowledge and all the major methods and tools proposed in the framework are refined and implemented successfully, the framework is tested and validated using real-life-like use cases. Another objective at this stage is to leverage the variations in the chosen tasks to test those methods and tools that were not used during the implementation stage.

The research at this stage is made up of two major activities:

1. **Choosing and designing the case studies**

   The case studies were chosen and designed to test and validate the framework to evaluate its effectiveness, cohesiveness and generality.

Importantly, the choice was dictated by the limitations of the framework and its methods and tools as they currently stand in this research.

2. **Testing the framework using the case studies**

   This task is similar to the implementation stage of the framework where all the functions of the framework are executed to digitise the manufacturing knowledge embedded within the tasks in the case studies. Any technical or practical issues with the framework are noted and fed back to the development stage to further refine the framework. Finally, a real manufacturing use case is chosen to confirm the findings of this research with respect to framework implementation. In this study, the constituents of manufacturing knowledge extracted are compared with the knowledge that already exists for the task as reported in literature. This comparison is expected to demonstrate whether or not the framework is capable of extracting both explicit and tacit manufacturing knowledge from actual, real-world manual manufacturing tasks.

3.6.8 Discussions and conclusions

Finally, the contributions made by this research to knowledge are identified by reviewing the outcomes against the proposed research aim and objectives. The research contributions are assessed to determine whether or not they bridge the research gaps. Following that, the limitations of this research are identified, based on which the future research directions are discussed. This facilitates the continuation of this research in order to overcome the limitations and enhances the research outcomes for the benefit of the manufacturing research and industrial community.

3.7 Chapter summary

This chapter states the research hypothesis, aim and objectives to develop a framework for digitisation of manual manufacturing task knowledge. The scopes of key stages of research are defined and the proposed research methodology to deliver the aim and objectives of this research are presented in detail.

CHAPTER 4

4 HUMAN-WORKPIECE INTERACTION THEORY

This chapter proposes the concept of human-workpiece interactions to represent a manual manufacturing task. It describes the hypothesis behind the concept and explains the seminal theories from literature that have been used to develop the proposed human-workpiece interaction (HWI) theory. The key concepts from this chapter are also presented in two journal papers and a conference paper (See 'List of Publications').

This chapter aims to achieve the following objectives:

- ❖ Present the research hypothesis and propose the basic HWI concept.
- ❖ Explain the adoption of Rasmussen's Skills-Rules-Knowledge (S-R-K) framework, Rasmussen's decision ladder concept and Gibson's theory of object affordances to develop the HWI theory.
- ❖ Describe the use of the proposed HWI theory for human task analysis.

4.1 Basic human-workpiece interaction theory

This research postulates that all manual manufacturing tasks involving a human and a workpiece can be represented by a series of interactions between a human and the workpiece. These interactions are characterised by the highly cognitive nature of manual manufacturing tasks in which human actions on the workpiece/s are guided by (a) constant human observation of the workpiece/s' states during the task, (b) human perception of the progress of the workpiece/s in real-time based on the observations and (c) dynamic human reasoning and decision-making based on the perceptions. The human action itself is a function of human dexterity, which is the manner in which the human uses his body to perform the task. The combination of cognitive ability and dexterity, also known as human skill varies from one human to another thereby causing varying quality of the same task when performed by different humans.

In this chapter, the HWI concept and resulting theory is proposed. The theory is constructed at its basic level and is built up to an advanced level by reinforcing it with the three seminal theories presented in section 2.4. The methodology of advancing the theory is shown in Figure 21.

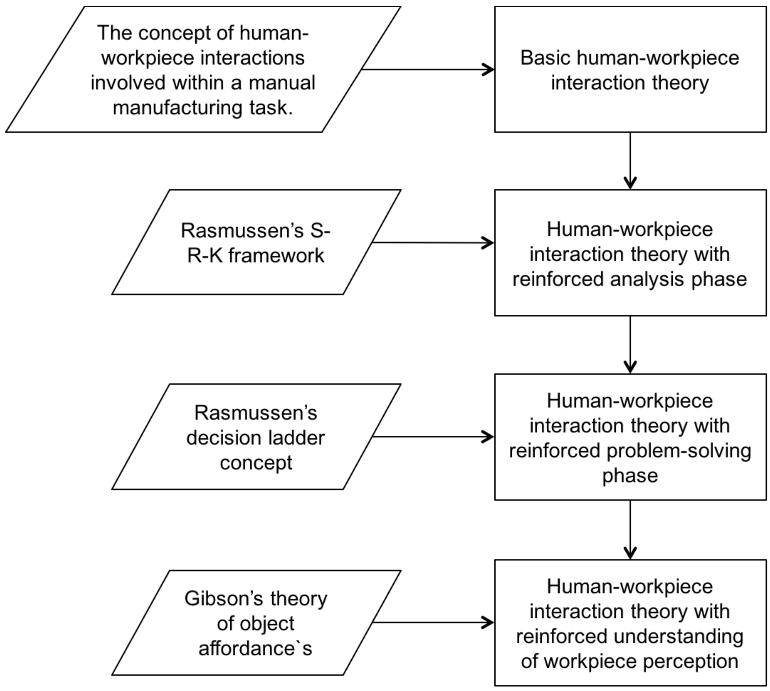


Figure 21: Methodology for the advancement of the HWI theory

The basic HWI theory is shown in Figure 22. This theory consists of two main entities, namely, the human and the workpiece. Here the workpiece denotes either one or multiple workpieces involved in the task such as objects to be manipulated or processed and tools used for manipulation or processing. The task may belong to any of the 3 major manufacturing categories, namely, machining, assembly or inspection.

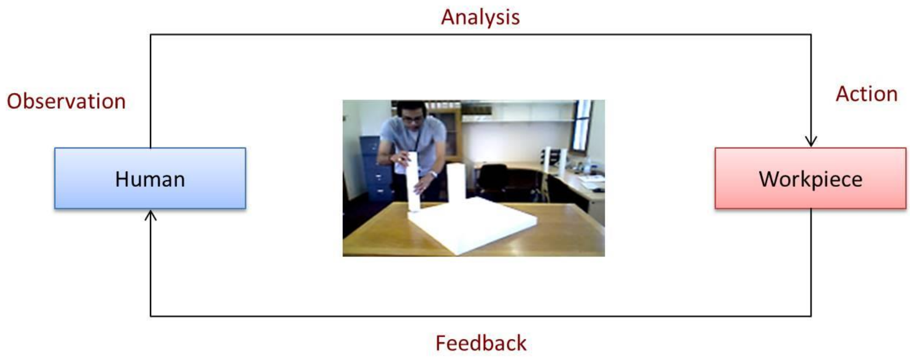


Figure 22: Basic HWI theory

According to the theory, every human action results in a change of state of the workpiece and this change of state is instantaneously and implicitly conveyed to the human via visible, audible or tactile feedback. The human observes this feedback and perceives its implication on the progress of the manufacturing task based on his/her intellect, skill and past experiences with the task. After analysing this, the human makes a choice on the next action to take on the workpiece such that the task is steered towards successful completion. The chosen action is then executed and the action-feedback-reaction loop is repeated until the task is successfully completed.

It is evident from this theory that there is a direct co-relation between human action and the workpiece state and therefore the manifestation of a human action at any instance during the task can be seen as the effect of that action on the workpiece.

4.2 Advancement of the HWI theory

The basic HWI theory introduced in section 4.1 is based on the research hypothesis. While the action and feedback segments of the human-workpiece interaction loop are self-explanatory, the analysis segment needs detailed investigation. This is because while human action and workpiece feedback is visible during task observation, the analysis of the feedback is not. While human skill is largely believed to manifest via his/her physical actions during the task, it also manifests significantly via the plans and choices that the human makes before and during the task.

Therefore, the analysis aspect of the theory is reinforced by three landmark theories in human behaviour and activity analysis, namely, Rasmussen's Skill-Rules-Knowledge (S-R-K) framework, Rasmussen's Decision Ladder and Gibson's concept of object affordances. Though these theories have been proposed for developing human-machine and human-product interfaces, and assessment of human errors and industrial accidents in manufacturing systems, this research has advanced the theory by applying these concepts to understand and model human-workpiece interactions that occur during manual manufacturing tasks. The interaction models in turn enable the digitisation of the manufacturing knowledge associated with the tasks, especially tacit knowledge such as human skill. There are no references in literature that have proposed the advancement and application of these seminal theories in digitisation of manual manufacturing task knowledge.

4.2.1 Application of the Rasmussen's S-R-K framework to advance the HWI theory

According to the S-R-K framework (Section 2.4.1), the human behaviour within a system varies according to the state of the system within which the human operates, the level of skill and expertise of the human, and the type of information available from the system to the human. The same can be applied to any manual manufacturing task in which a human manipulates or processes a workpiece or a set of workpieces. Examples of such manual manufacturing tasks in the industry are complex assemblies, 3D part polishing, spray-painting, composite fibre layup, among others. A typical manual assembly task is considered to explain the application of the S-R-K framework to the proposed HWI theory.

Rule-based behaviour: If the assembly task is well defined and has a structured set of rules to be applied for all the operations within the task, the human actions to complete the assembly are rule-based. For each action within the assembly, the human identifies the state of the workpiece/s by observing and analysing specific visible or tactile signs, such as workpiece dimensions and orientation of parts and then chooses an appropriate rule from a set of rules to respond in order to progress the assembly task.

Skill-based behaviour: A long and repetitive practise of such rule-based behaviour in the assembly task leads to the human acquiring a level of expertise and competency such that the behaviour changes from being rule-based to skill-based. In the skill-based behaviour, the human responds to the changes in the workpiece during the assembly task almost automatically with appropriate actions in relation to the workpiece information - perceived as signals. The human no longer consciously observes and analyses the specific signs but subconsciously takes note of those signs as continuous time-space signals and produces a reflex to them using actions that are guided by task sequence and muscle memory from the past. These signals provide information about the current state of the workpiece and allow the experienced user to respond with sub-conscious actions.

Knowledge-based behaviour: Growing expertise and competence also allows the human to solve new or unforeseen problems within the assembly task by elevating to the highest knowledge-based behaviour. In this kind of behaviour, the human relies on intelligent reasoning, pattern matching from experiences of the past, generation of hypotheses and attempting to verify them to predict workpiece states to solve an unforeseen problem. For this, the human consciously perceives and analyses the symbols that represent the problem within the assembly task, such as a missing part or wrongly assembled parts, and takes the most suitable corrective actions based on his/her experience of understanding the task. These problem-solving procedures tend to be memorised if solved multiple times and the human actions within those procedures are converted from being knowledge-based to rule-based subsequently.

It is therefore inferred that the analysis aspect of the basic HWI theory can be expanded into skill-based, rule-based and knowledge-based segments with each of these segments influenced by the S-R-K framework. The HWI theory reinforced by the S-R-K framework is shown in Figure 23.

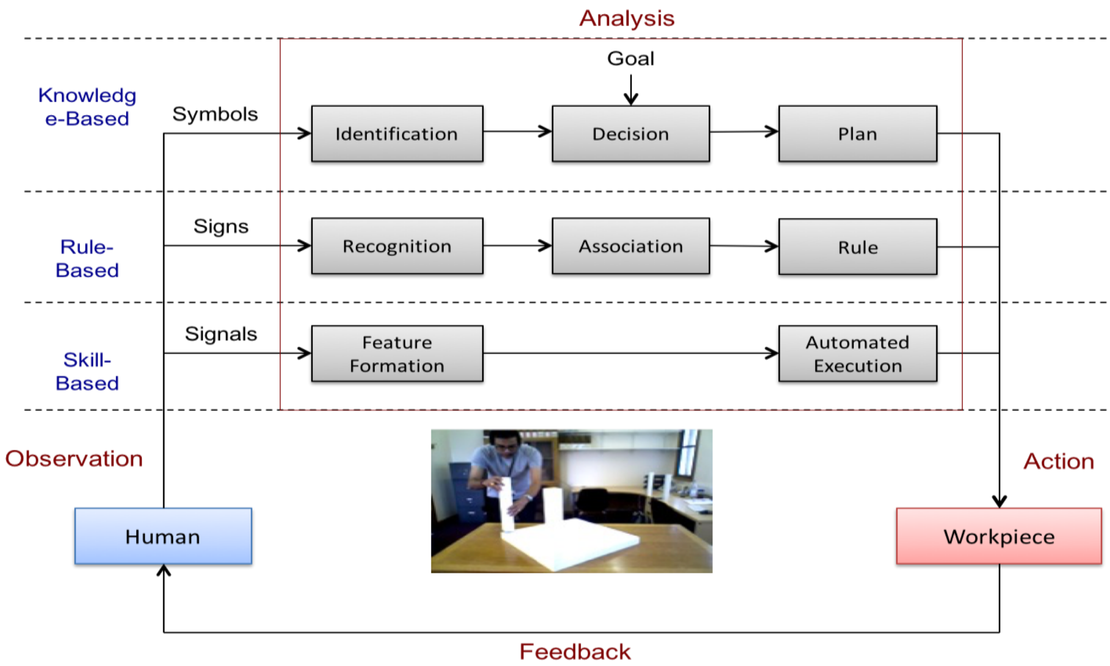


Figure 23: Application of the S-R-K framework to the HWI theory

Signals, signs and symbols

In the S-R-K reinforced theory, every human action is followed by feedback from the workpiece on its state of progress. Depending on the level of behaviour displayed by the human at any given point during the task, this feedback can be perceived as signals, signs or symbols. At the skill-based level, the workpiece feedback is perceived as continuous time-space signals followed by forming the feature of the intended workpiece outcome in mind. The sensorimotor instinct then takes over to sub-consciously execute the next intended action on the workpiece.

At the rule-based level, when the human must follow a standard operating procedure to execute the task, the workpiece feedback is perceived as signs.

These signs are recognised and associated with a pre-defined pattern to pick a relevant rule from the stored set of rules to progress the task.

In certain situations when unforeseen problems occur, the human behaviour is classified as knowledge-based and at this level the workpiece feedback is perceived as symbols that indicate a problem. Based on the intended final goal, past experiences of the task, and generating and testing new hypotheses, the human plans a set of action and executes them to solve the problem. Depending on the success of the solution, the workpiece either successfully progresses to completion or displays symbols of persisting with the same problem or occurrence of a new one. A successfully solved problem is memorised, the plan that solved the problem is stored as a rule and the next time the same problem occurs, the human operates at the rule-based level. The more experienced a human gets at a task, the more likely it is that he or she performs the task at the skill-level rather than at the rule-based or knowledge-based levels.

4.2.2 Application of the Rasmussen's Decision Ladder concept to advance the HWI theory

In the proposed HWI theory, the human behaviour during the analysis phase of the interaction is segregated according to Rasmussen's S-R-K framework. Though decision-making is present at all 3 levels, it is most prominent in the knowledge-based level. At this level, the human attempts to solve new and unforeseen problems by making decisions based on what he/she observes, what he/she knows from past experiences, what the overall goals of the task are and by anticipating the consequences of his/her decisions on the workpiece and the task at large. At the skill-based level, decision-making is almost absent because the skilled human performs his actions sub-consciously based on sensorimotor inputs and only observes the feedback from the workpiece sub-consciously. At the rule-based level a certain degree of decision-making exists but those decisions are simple and are merely there to choose rules that govern every subsequent action on the workpiece. Therefore, in order to understand the decision-making process better at the knowledge-based level, the

Rasmussen's Decision Ladder is used and the main four steps of the ladder are adopted (Figure 24).

In the '*Activation'* step, the need for response is activated after the human detects a problem with the workpiece during one of the interaction loops, for example, a wrongly assembled part, categorised as an abnormal condition.

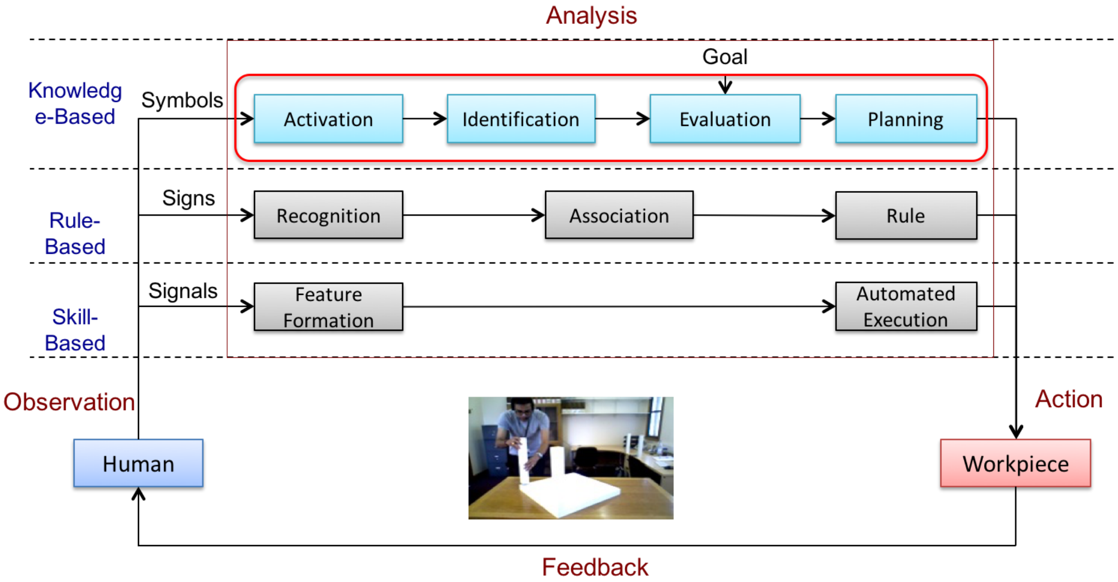


Figure 24: Rasmussen's decision ladder applied to the HWI model

In the '*Identification'* step, the workpiece state is identified via the various symbols that are observed by the human as part of the feedback from the workpiece. Identification of the workpiece state allows the human to compare the abnormal state with the expected normal state at this point of the task and therefore to decide on the next course of action to solve the problem. In the '*Evaluation'* step, the human evaluates all the possible solutions that he/she could implement on the workpiece to solve the problem based on his/her past experience or even generating new hypotheses and verifying them. The evaluation gives the human the ability to choose between competing solutions keeping the overall goal of the task in mind. Finally, in the '*Planning'* step, the human simulates the solution in his/her mind to visualise the target state and plan the necessary next action to take the workpiece a step closer towards problem resolution. This is followed by the actual execution of the chosen action. The subsequent perception of the new state of the workpiece will indicate if the problem is completely solved. If the problem persists or a new one surfaces, then the decision ladder is repeated at until the problem is completely solved.

4.2.3 Application of Gibson’s concept of object affordances to advance the HWI theory

Skilled humans perform complex manual manufacturing tasks on a routine basis. They have a vast repertoire of expert actions to execute the task governed by the overall goal of the task. Each chosen action exerts its effect on the workpiece changing its state either positively or negatively depending on whether the workpiece is progressing towards successful completion or otherwise respectively. For example, a skilled spray-painter knows the precise painting gestures and the right amount of paint release to ensure a uniform coating of paint on the vehicle body and a skilled polisher knows the correct amount of pressure to exert and the optimum tool paths to follow to achieve the desired surface finish. In order to decipher these human skills, it is necessary to capture the basic human actions and the corresponding changes undergone by the workpiece in real-time. The proposed HWI theory enables the digitisation of manual manufacturing task knowledge with an aim to extract, decode and reproduce human skills associated with the task.

Why use object affordances?

The method used to capture the human-workpiece interactions is daunting considering that both human action capture and workpiece progress tracking must be performed simultaneously in real-time within a shopfloor environment that may not be conducive with varying complexities of workpieces and human gestures. Therefore, there is a need to simplify the methods by considering any known aspects of the task into the capture algorithms. For example, while tracking the progress of a wheel loading task in automotive assembly, the object recognition algorithm is greatly simplified if the fact that the wheel is always circular is considered.

In order to consider the known aspects of the workpiece during the manual task, this research draws inspiration from the seminal concept of object affordances introduced by Gibson (1979). Gibson suggests that the object affordances are action possibilities available to an individual to execute on an object depending on his/her action capabilities. The dominance of the visual sensory system over others and people's ability to pick up affordances implies that human actions are aided by visual feedback from the workpiece during the task. However, other perceptual systems such as tactile feedback (for assembly tasks) or audible feedback (for machining tasks) cannot be ignored and must be considered as viable inputs to the human for associating affordances to workpieces during the task.

If all the known affordances associated with the task workpiece/s are considered, the task capture module can be tuned to pick up those affordances easily. This way, there is no need to write complex object recognition and tracking algorithms to observe and analyse the entire workpiece through the duration of the task. Also, different parts of the workpiece could be observed and tracked based on particular phases of the task. Again this information can be fed prior to the task capture itself so that the sensors capturing the workpiece can focus on those parts only thereby reducing computational load. Understanding of workpiece affordances in a manufacturing setting is feasible because (a) the manufacturing task follows a known sequence of steps except during abnormal problem-solving sessions and (b) the workpieces are engineering artefacts and therefore have drawings and Computer Aided Design (CAD) models to take reference from to develop simpler recognition and tracking algorithms.

Affordances are context-dependent and define the relationships between the human and the workpiece during the task. Therefore, workpiece affordances (e.g. rough surfaces afford polishing, or a hole affords insertion of a peg) can be empirically determined by multiple observations of the tasks and eventually predicted by analysing the properties and features of workpieces. Once the workpieces are associated with affordances, wrong action-effect combinations may indicate a problem and alert the human to operate in the knowledge-based realm to solve the problem.

In the proposed HWI theory, the human considers workpiece affordances in the analysis phase when the next action on the workpiece is being planned as shown in Figure 25. Therefore, at the skill-based level, the object affordances are subconsciously taken into account because the workpiece feedback unambiguously conveys the appropriate affordance.

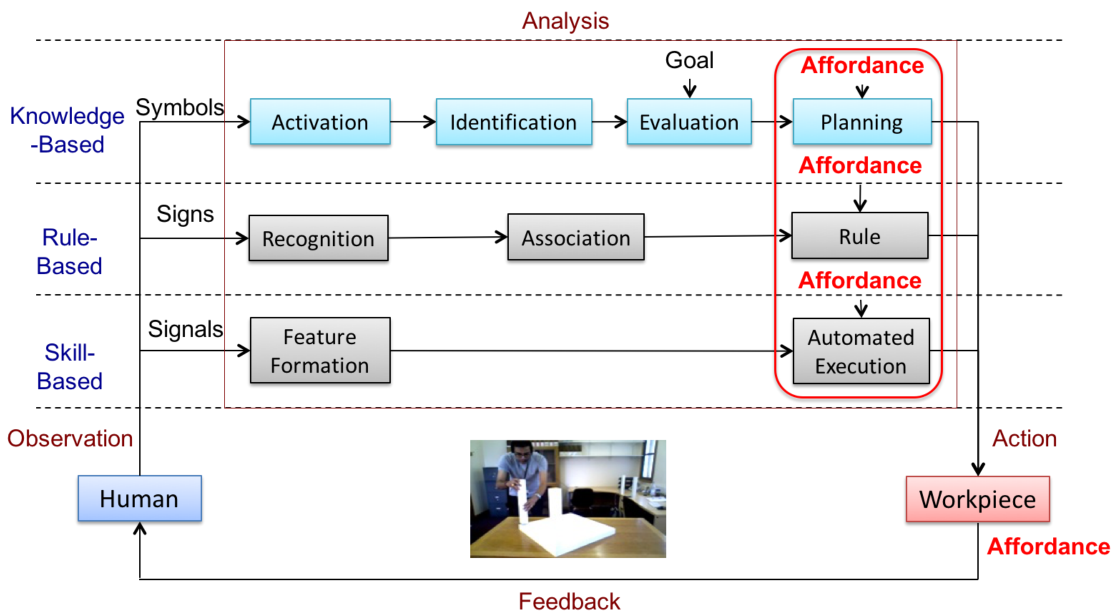


Figure 25: Application of Gibson's theory of object affordances to the HWI theory

At the rule-based level, the human consciously considers the workpiece affordance in order to confirm that he/she has chosen the correct rule that governs his/her next action on the workpiece. However, at the knowledge-based level, due to a problem in the task, the workpiece feedback may indicate multiple affordances from which the human has to choose one and plan his/her problem-solving action based on the chosen affordance.

Starting from the simple HWI theory, this research has applied Rasmussen's S-R-K framework to segregate the analysis phase of the theory into 3 different categories of human behaviour during the manufacturing task. The frame of reference for the decision-making steps within the knowledge-based level is taken from Rasmussen's decision ladder. Finally, Gibson's concept of object affordances is used to suggest that the human visualises workpiece feedback as an affordance and as a factor to choose the next most appropriate task action. The use of this advanced HWI theory will inform the development of the proposed framework for digitising manual manufacturing task knowledge.

4.2.4 Application of HWI theory for human task analysis

This research proposes that the advanced HWI theory can support a potent approach to acquire detailed insights into the cognitive structure of the manual manufacturing tasks and therefore provide significant information for human task analysis, especially the Cognitive Work Analysis (CWA) component. The CWA itself consists of five components, namely, Work Domain Analysis (WDA), Strategy Analysis (SA), Control Task Analysis (CTA), Social Organisation and Cooperation Analysis (SOCA) and Worker Competency Analysis (WCA) (section 2.4.4) out of which only the SA, CTA and WCA are considered in this research. WDA and SOCA are not considered because of the use of high-level abstraction frameworks and the study of human factors in these analyses respectively requiring a qualitative research approach. This is outside the scope of this quantitative research.

Strategy analysis: By capturing the human actions on the workpiece and mapping them to corresponding workpiece changes in time and space, the task strategy used by the human can be identified and analysed. This is the SA component of CWA. There are two classes of methods to perform SA, namely, formative methods and empirical methods. Generally, the formative methods work on inferring the strategies from interviewing the humans involved in the tasks rather than extracting the strategies by observing the humans perform the tasks. In the empirical method, workers in different task settings are observed and analysed, workers' verbal reports are analysed and other related information about the task such as task constraints are collected. This research therefore uses the empirical method.

Control task analysis: Each workpiece progress is associated with the human action immediately preceding it in time, and the action itself is based on the workpiece feedback received prior to the action. Therefore, the decisions that the human makes based on the workpiece feedback during the task can be inferred by analysing the action choices of the human during the task. This is the CTA component of CWA.

Worker competency analysis: Within the captured HWI data, human actions during the task can be extracted in digital form. The effectiveness of these actions in the task depends on the competency of the worker. This enables the evaluation of worker competency and the comparison of competencies between two or more workers for the same task. CWA further enables the following capabilities:

i. Ergonomic analysis of a manual-manufacturing task for improving workplace health and safety.
ii. Error analysis of manufacturing tasks for reducing human-induced industrial accidents.
iii. Human skills transfer process for up-skilling the workforce.

This research proposes to use the empirical method of cognitive work analysis (CWA) to extract the manufacturing task knowledge comprising human action skills, human reaction/control skills and task strategies. This method is designed and executed in accordance with a framework proposed in this research.

4.3 Chapter summary

This chapter proposes the basic HWI concept and explains how the HWI theory is developed by using seminal theories from literature such as Rasmussen's S-R-K framework, Rasmussen's decision ladder and Gibson's theory of object affordances. The HWI theory acts as a guide for the development of the framework for digitisation of manual manufacturing task knowledge that has the potential to facilitate CWA. The proposed framework is presented in the next chapter.

CHAPTER 5

5 FRAMEWORK FOR DIGITISATION OF MANUAL MANUFACTURING TASK KNOWLEDGE

This chapter proposes a framework for digitisation of manual manufacturing task knowledge. It describes the major steps, methods and tools used in the proposed framework. The framework is also presented in a journal paper (See 'List of Publications').

This chapter aims to achieve the following objectives:

- ❖ Define and explain the terms of reference used in the digitisation framework.
- ❖ Propose the design and structure of the digitisation framework.
- ❖ Propose the major steps used in the framework and explain the inputs, outputs, methods and dependencies of each step.

5.1 Terms of reference

Framework is defined in Merriam Webster as “a basic conceptual structure or a frame of reference for a set of ideas or facts in order to build a system”. This research proposes to develop a cohesive process, enabled by ICT, to digitise manufacturing knowledge associated with manual manufacturing tasks. A framework is therefore required to provide a structure for this process to define its inputs, methods, dependencies and outputs.

5.1.1 Framework structure

The digitisation framework is used to support an ICT platform and therefore each of its proposed components is organised according to the four basic elements of informatics data flow (Figure 26).

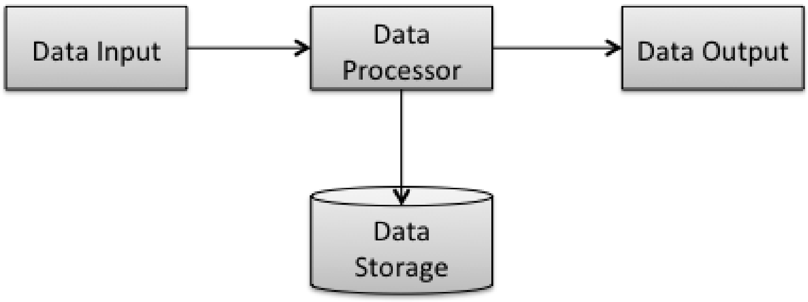


Figure 26: Basic elements of each ICT component of the framework

Data Input: At the overall framework level, the primary data input to the ICT platform comes from gaming interface devices, such as the Microsoft Kinect™, that capture and record human action and changing workpiece configurations during a manual-manufacturing task into digital data. Secondary inputs are taken from the experts executing the manual task in the form of verbal cues during the task to note certain critical sub-tasks, published task guidelines, engineering drawings of the workpieces and manufacturing environment, and technical task manuals if available.

Data Processor: The input data is processed using a set of software programmes that run standard and bespoke algorithms including third party Application Programming Interfaces (APIs) for human skeletal motion processing, depth image processing to recognise and track changes in workpiece, audio processing, modelling and analysis of processed data, and mining of manufacturing knowledge from the models produced.

Data Storage: Input data, processed data and produced models are stored and archived in appropriate digital data storage units such as Comma Separated Values (.CSV) files, spreadsheets, and databases for searchable and on-demand access.

Data Output: The input, processed and stored data can be queried to extract the manufacturing knowledge embedded within this data which can be disseminated in various forms using multiple media depending on the application. Various manufacturing knowledge constituents can be extracted, such as task strategy, problem-solving approach, expert gestures during workpiece manipulation, body motion mechanics, etc. This knowledge can be reproduced using multiple media.

5.1.2 Digitisation

Digitisation is defined in Merriam Webster as "a process of converting analogue information into digital information". This definition has a wide scope and therefore there is a need to define a suitable scope of digitisation implemented in this research.

As explained in section 4.1, any manual manufacturing task can be represented as a series of human-workpiece interactions in which the human using his skills, knowledge and experience manipulates a workpiece from its initial state to its final desired state. The manufacturing knowledge associated with this task is embedded within these interactions and the manufacturing environment at large. To extract this knowledge, the interactions and the environment have to be in a form that can be capable of being analysed. The interactions and the manufacturing environment are physical in nature and the human decisions that affect the interactions and the environment are abstract in nature. Therefore both these elements need to be digitised so that the resultant digital data can be processed and analysed using ICT algorithms, methods and tools to extract the manufacturing task knowledge.

5.1.3 Manual manufacturing

A manual-manufacturing task is one that does not involve the use of any mechanised or automated hardware or software tools for the primary operation of manipulating engineering workpieces. In order to compensate for the lack of automation, the human performing the manual task uses his set of skills including but not limited to his intellect, sophisticated cognition, dexterity, adaptability, smart decision making and the ability to learn from past experiences and training. Manual manufacturing entails a wide spectrum of tasks belonging to the 3 main families of manufacturing operations, namely, machining, assembly and inspection.

5.1.4 Manufacturing knowledge

The knowledge embedded within or associated with a manual-manufacturing task is made up of two main components, namely, the explicit component and a tacit (hidden) component. The explicit component includes standard operating procedure manuals, engineering drawings, manufacturing environment layouts and constraints, etc. and is well documented. The tacit component includes human action skills and human reaction or decision making skills that affect the task and is not documented because of the difficulty in capturing and expressing this knowledge.

The human action and control skills manifest themselves during the task in the form of human-workpiece interactions that are responsible for successful implementation of the task and involve the following key knowledge constituents:

i. Task strategy planned prior to the task and adjusted during the task.
ii. Motion characteristics and mechanics that make up human actions to manipulate workpiece/s during the task.
iii. Effects of specific human actions on specific workpiece configurations and their time-space dependencies during the task.
iv. Expert human gestures to solve unforeseen and unexpected problems that occur during the task.

This research strives to extract and present the above 4 aspects of hidden manufacturing knowledge in a digital form that can be accessed on demand, and consumed using multiple media as required by the end application.

5.2 Development of the framework

The proposed framework for digitisation of manual manufacturing task knowledge comprises 6 main steps as illustrated in Figure 27. Each step of the proposed framework includes its own data input, data processing methods, internal and external dependencies, and data output. The data output of one step is the primary data input to the next step. The following section illustrates and describes each step of the framework.

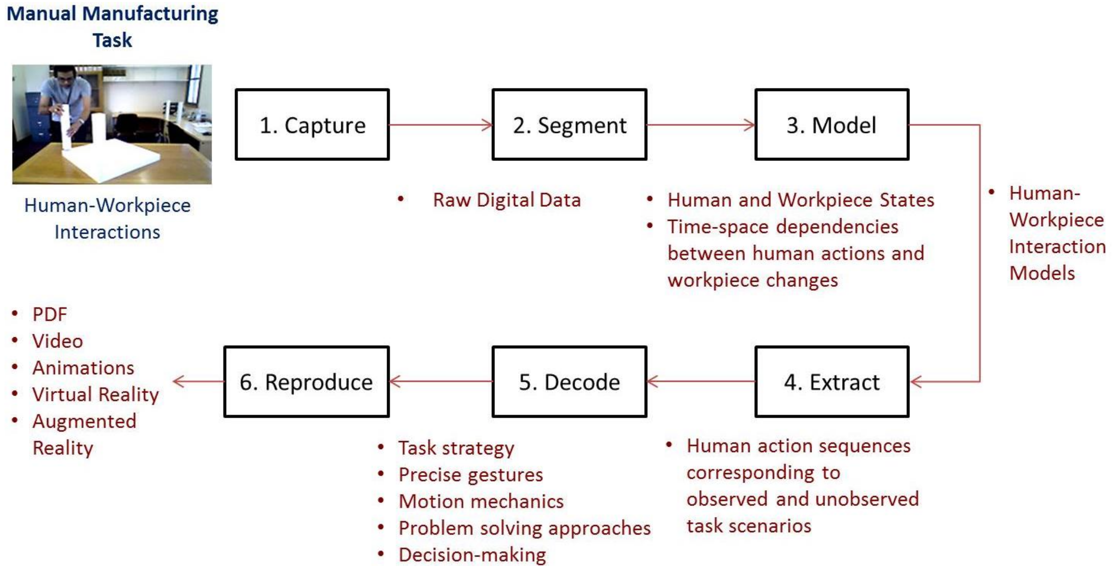


Figure 27: Framework for digitisation of manual manufacturing task knowledge

Step 1: Capture

Purpose: To capture and record the human-workpiece interactions occurring during a manual manufacturing task within a manufacturing environment into digital data (Figure 28).

Description: In order to capture the human-workpiece interactions, it is necessary to track both the human actions and the changes undergone by the workpiece simultaneously in real time for the entire duration of the manual task. Human actions involve complex skeletal motion such as translational and rotational hand gestures, motion mechanics, body postures, grip-release movements, etc. These human actions directly affect the workpiece causing it to change shape, dimensions, colour, orientation, etc. depending on the task being performed.

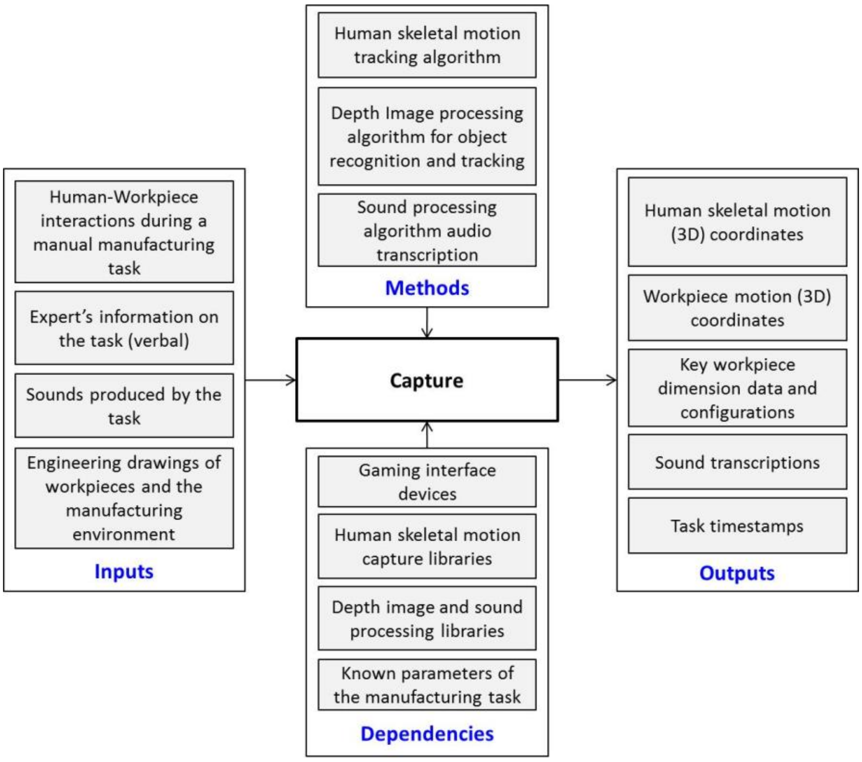


Figure 28: The 'Capture' step and its elements

Inputs: Apart from capturing the actual manual manufacturing tasks (the primary input to this step), secondary inputs about the task such as descriptions of critical human actions and verbal indication of when problems are being solved during the task can be obtained from the expert performing the task. In certain tasks, the sounds produced by the workpiece or the tool indicate the progress of the task, such as in machining or polishing. This input can be obtained by recording these sounds along with capturing the human-workpiece interactions. Finally, inputs such as engineering drawings of the workpiece and critical components of the manufacturing environment help in scoping, structuring and simplifying the process of capturing the task.

Methods: Human action is captured by using one or more Kinect sensors. The Kinect sensor generates RGB and depth image streams of the 3D scene that is being captured. The associated software libraries, both standard and bespoke, are used to simultaneously extract human skeletal motion from the images and the changes made to the workpiece. Sounds generated during the task and spoken words by the human while performing the task are also recorded using stereo microphones and processed into transcriptions using standard audio processing and speech-to-text libraries.

Dependencies: The accuracy of the human-workpiece interaction capture depends heavily on the resolution of the Kinect. While the human skeletal tracking is a mature functionality and comes standard with the Kinect, object recognition and tracking must be custom developed and is constrained by its relatively low resolution (640 x 480 pixels). Also, the human skeletal tracking performance is affected by occlusions; i.e. when the device does not get an unobstructed view of the human body. Therefore, skeletal motion tracking reliability depends on the scale of the occlusion during the task. Finally, all the known aspects of the task such as the physical boundary of the task, exclusion of highly unlikely scenarios of the task and predictable configurations of the workpiece during the task can be considered in the capture algorithms and simplify the task capture step. These dependencies limit the level of task complexity that can be reliably captured.

Outputs: Tracking the human skeletal motion results in a digital data stream of 3D coordinates (x, y, z) of the 20 skeletal joints of the human body for the entire duration of capture. Recognising and tracking the workpiece results in a digital stream of 3D coordinates (x, y, z) of workpiece positions, numerical dimensions, shapes in the form of 2D silhouettes and workpiece configurations in the form of constituent parts. Transcriptions of sound data form a part of the digital data output if sound was captured during the task. All these digital outputs have a timestamp associated with them indicating the exact time of capture for each data point.

<u>Step 2: Segment</u>

Purpose: To segment the raw data generated in step 1 into discrete human action states and workpiece states that collectively and sequentially represent the entire series of human-workpiece interactions captured during the manual manufacturing task (Figure 29).

Description: The human-workpiece interactions captured in step 1 are recorded as continuous raw digital data. In order to extract and decode the manufacturing knowledge embedded within this data, it needs to be filtered to remove unwanted data or noise and segmented into distinguishable states that can be modelled and analysed using an appropriate state-machine based learning technique. Therefore, the segmentation step segregates the raw and continuous human-workpiece interaction data into discrete human action states and workpiece states, associating the sound transcriptions with the corresponding states based on the recording timestamps.

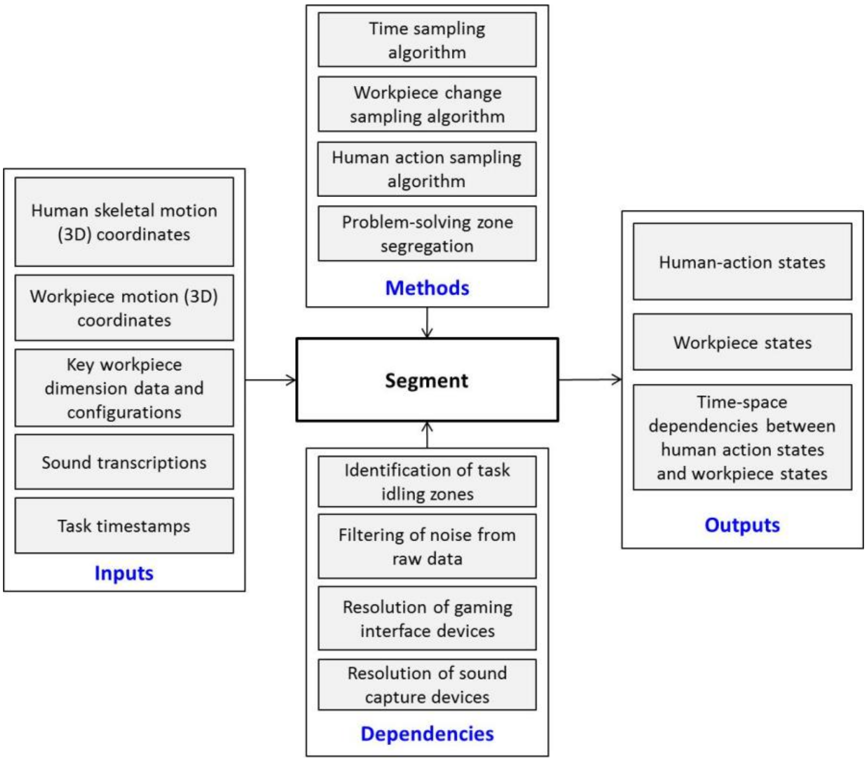


Figure 29: The 'Segment' step and its elements

Inputs: The output of step 1 consisting of human skeletal motion data, workpiece motion, dimension and configuration data, sound transcriptions and task timestamps form the input for the segmentation step. Task timestamps provide the fundamental link between the human, workpiece and sound data because all these 3 sources of data are recorded simultaneously during the task capture step.

Methods: Manual manufacturing tasks differ from each other based on how the workpiece changes during the tasks. The segmentation method changes depending on whether the workpiece undergoes a gradual change or big changes at regular or irregular intervals, whether the trajectories of human actions during the task are smooth, spasmodic or cyclical and whether the human mainly operates in the skill-based, rule-based or knowledge-based levels during the task. This research considers 3 categories of tasks based on the above conditions.

In the first category, the workpiece changes are gradual in nature for the duration of the task and the human uses his skill-based behaviour much more than rule-based or knowledge-based behaviours. An example of such as task is manual polishing in which the surface of the workpiece gradually changes from rough to smooth. For this category of tasks, the human-workpiece interaction data is segmented using time sampling.

In the second category, the human action is primarily made up of big and abrupt changes to motion trajectories, such as in complex assemblies where constituent parts need to be assembled on the workpiece using precise but big gestures seen among the otherwise smooth trajectories. In these tasks, the human uses a combination of rule-based and knowledge-based behaviour much more than skill-based behaviour. For this category of tasks, the human-workpiece interaction data is segmented using trajectory-change sampling.

In the final category, the workpiece undergoes prominent changes to its position, dimension, shape or configuration or a combination of these characteristics as opposed to the gradual workpiece changes seen in the first category. The human action trajectories are usually cyclical for these tasks. An example of such a task is composite layup where repetitive human actions are required to manipulate the composite ply and lay it over a complex mould. In these tasks, the human uses a combination of rule-based and knowledge-based behaviour much more than skill-based behaviour. For this category of tasks, the human-workpiece interaction data is segmented using workpiece-change sampling.

In addition to segmenting the data into discrete human action and workpiece states, it is also necessary to identify and isolate problem-solving sessions that have occurred within the human-workpiece interactions during a manual task. Isolation of such sessions will enable further analysis, extraction and decoding of problem-solving strategies including the choice of corrective actions made by the expert while solving the problem.

However, before segmentation, the continuous raw data needs to be filtered to remove any unwanted data. This filtering is done by removing the high frequency noise from human skeletal motion and idling periods during the task that do not contribute to the progress of the task. A simple threshold comparison algorithm is used to filter out such noise and the threshold values are obtained empirically and/or in consultation with task experts. The three segmentation techniques are described below.

i. **Time sampling:** This is the simplest form of segmentation where the continuous human skeletal motion data and workpiece progress data is segmented every 'n' units of time. The unit used in this research is 'seconds' and the value of 'n' is chosen based on the frequency of small changes in human actions that result in incremental changes in the workpiece.

ii. **Trajectory sampling:** In this technique, continuous human skeletal motion data is analysed and segmented at points where an abrupt change in gesture such as change in motion direction, angles between body parts, acceleration and/or speed is detected. The continuous workpiece data is also segmented at the same points. Specific joints of the human skeleton are chosen depending on the parts of the body that are predominantly used to perform the task. For example, in an assembly task, abrupt changes in only hand gestures are used to segment human motion data. A threshold to determine the level of abruptness in motion is set prior to the implementation of the segmentation step. The threshold value is arrived at on the basis of empirical means and/or in consultation with the task experts.

iii. **Workpiece sampling:** In this technique, the workpiece characteristics such as position, dimensions, shape and configuration are analysed to find points where significant changes in any of these characteristics have occurred. The continuous workpiece data is segmented at these points to obtain discrete workpiece states. At the same points, corresponding human action data is also segmented to obtain discrete human action states. As in the trajectory-based segmentation technique, a threshold value is chosen to determine the level of workpiece change that is considered significant enough for segmentation. This threshold value is estimated based on empirical data and/or in consultation with the task experts. For example, in the composite layup task, the workpiece is divided into several sectors and the change in ply surface topography changes significantly from being not laid to fully laid in those sectors. The segmentation is performed every time the laminator progresses from one sector to another.

iv. **Segregation of problem-solving sessions:** One key objective of this research is to identify problem-solving sessions within the captured human-workpiece interactions and analyse these sessions further to extract and decode the knowledge-based behaviour used by the human to solve the problem. There are two ways investigated in this research to identify and isolate problem-solving sessions within the captured human-workpiece interaction data.

    In the first method, the corresponding human and workpiece states generated from the human-workpiece interaction data for two or more runs of the same task are compared. The presence of additional states in some runs of the task indicates peculiar human behaviour and unexpected changes to the workpiece during those runs. It is important to isolate and analyse such states because in some cases they could be representations of problem-solving sessions and for some it could be noise (false data captured as a result of low resolution of the Kinect or unwanted data that was missed by the filtering process).

In the second method, verbal instructions are issued by the task expert to the capture step to note his/her set of actions as problem-solving actions. A verbal indication to start noting the problem-solving session is processed using a standard speech-to-text function and all human skeletal motion and workpiece-change data captured from this point on is flagged as a problem-session session until a verbal instruction to stop is issued by the expert. This way there is no need to compare human and workpiece states of task runs to isolate problem-solving sessions. Both these methods have been investigated in this research.

Dependencies: The performance of the segmentation step depends on the quality of human-workpiece interaction data being captured and recorded. This quality is further dependent on the resolution of the Kinect and the sound capturing microphones. Higher the resolution less is the noise in the captured data. With the primary human action and workpiece-change data coming from the Kinect, its low image resolution introduces a significant amount of noise in the captured data.

Other factors that affect the performance of the segmentation step are the thresholds used for segregating human and workpiece data in the trajectory-change-based and workpiece-change-based segmentation techniques. Realistic values of these thresholds must be obtained based on past task capture data and/or with practical inputs from task experts based on their task knowledge. Unrealistic values will result in incorrect generation of human and workpiece states. A higher than realistic threshold will result in missed states and loss of knowledge whereas a lower than realistic threshold will generate redundant states introducing more complexity, which does not necessarily mean more accuracy in the subsequent modelling step.

Outputs: Two main outputs are generated from the segmentation step, namely the discrete human action states comprising human skeletal motion data within each state and discrete workpiece states comprising the positional, dimensional, shape and configurational changes within each state. Out of the several ways to represent human action states, one way is to use Therbligs.

Therbligs are mainly used in human motion study in work environments, such as in an assembly line to evaluate task productivity. For certain tasks, Therbligs can be investigated to represent human action states and one such investigation is conducted in this research. A third but not a discrete output is the time-space dependencies between human action states and corresponding workpiece states based on a common time and space of capturing the manual-manufacturing task.

Step 3: Model

Purpose: To model the human-workpiece interactions represented as discrete human and workpiece states for each observation of a manual manufacturing task (Figure 30).

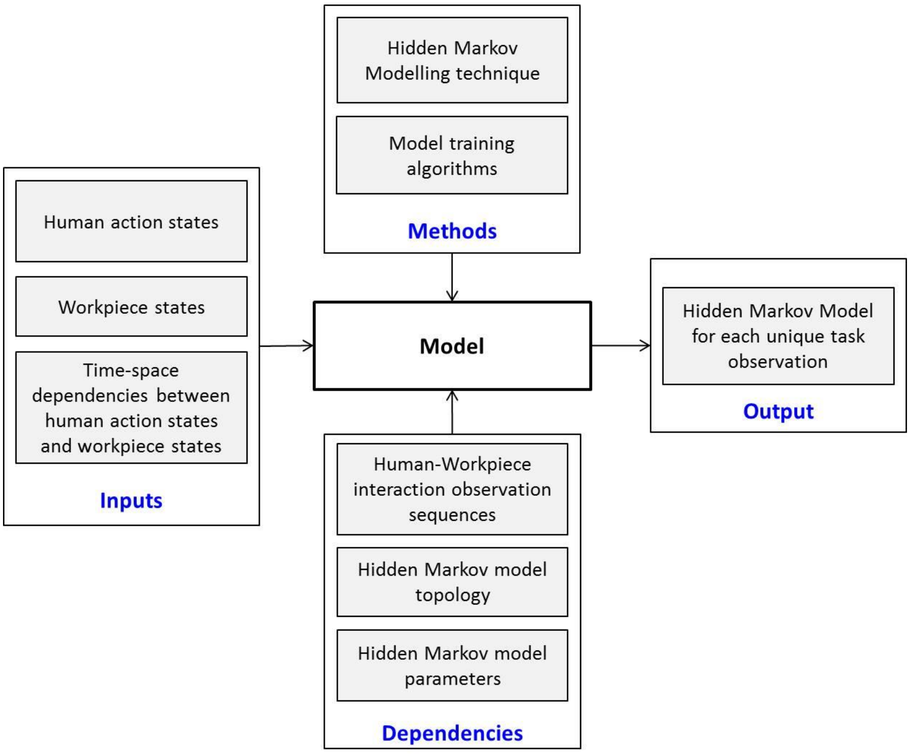


Figure 30: The 'Model' step and its elements

Description: The discrete human action states and workpiece-change states generated in step 2 are non-deterministic in nature, which means that the same states may not be generated every time the same manual manufacturing task is captured. Because of the manual nature of the task, it is not performed in exactly the same manner even by the same human and small differences between the tasks may result in different states. Therefore, in order to analyse these states to extract and decode the manufacturing knowledge embedded within them, a non-deterministic discrete state space modelling technique can be used.

Inputs: The inputs for the modelling step are the discrete human action and workpiece states as well as the time-space dependencies between these states as generated in step 2.

Methods: There are two popular methods to model time series data of multivariate observations, which in this case are the human-workpiece interactions, namely Hidden Markov Modelling (HMM) or Conditional Random Fields (CRF). Both are graphical models that describe how the input states of a system are connected, the probabilities of input states to emit output states and enable the prediction of a sequence of input states given a sequence of output states.

HMM is a generative modelling method in which the input states are independent of each other except for the Markov assumption that the next input state is dependent only on the previous input state, thereby modelling a system as a directed graph with a specific state sequence. There is no relationship between the output states themselves. The transition probabilities between input states and emission probabilities between input and output states are fixed and time-invariant.

CRF is a discriminative modelling method in which the input states depend on other inputs states and also affect them not necessarily in any sequence, thereby modelling a system as an undirected graph. The Markov assumption could be made a part of these dependencies. The output states however are independent of each other like in the case of HMM. However, the main difference between CRF and HMM is that in CRF the transition probabilities between input states and the emission probabilities between input and output states are conditional and not fixed. Therefore, more context-dependent variables can be modelled in CRF, which is not possible in HMM. Therefore, CRF seems to be a better choice in terms of the richness of detail that can be modelled for a time-series system (Sutton and McCallum, 2011).

Despite CRF being a superior modelling method, HMM is used in this work because of two primary reasons. First, HMM can be used to model a system in which input states are hidden and the relationship between them is not completely known whereas in a CRF, the relationships must be defined as a conditional function. Second, even when the state transition and emission probabilities are not known, an HMM model can be trained by using only the observed output states using unsupervised learning algorithms, whereas unsupervised learning in CRF is less known in literature. Sutton and McCallum (2011) state: “For any particular data set, it is impossible to predict in advance whether a generative or a discriminative model will perform better. Finally, sometimes either the problem suggests a natural generative model, or the application requires the ability to predict both future inputs and future outputs, making a generative model preferable.” Given the above arguments and the new approach of this research to extract knowledge from tasks by modelling the human-workpiece interaction within the tasks, HMM was chosen as a practical attempt.

HMM is used to model the human-workpiece interactions that are segmented into discrete human action and workpiece states. An HMM is a stochastic machine learning tool used to model and analyse systems that can be represented as state machines that display the Markov condition; the current state of the system depends only on the previous ‘n’ states of the system. In addition to the Markov condition, another factor in an HMM is that the set of states that affect the system are hidden (not observable directly) whereas the effects of those states on the system are observable. The transitions between the hidden states and the observation states are represented stochastically, i.e. by assigning probabilities for hidden state transitions and for observation states caused by the hidden states. These probabilities are assumed to be time-invariant, i.e. the probabilities do not change over the duration of the task and therefore it is vital that these probabilities are estimated empirically. These probabilities are initially assigned in the HMM model on random basis and/or in consultation with the task experts and may not be optimal. Therefore, the HMM models must be trained in order to generate optimal probabilities so that the human-workpiece interactions could be modelled as accurately as possible. Training is conducted using an expectation-maximisation (EM) algorithm that optimises the probabilities of the HMM to the local optimum. A detailed introduction to HMM is presented later in this chapter (section 5.3).

Dependencies: Modelling of human-workpiece interactions depends on the primary characteristics of the HMM models, namely, the topology of the model determined by the number of hidden states (human action states) and the number of observation states (workpiece states) and the model parameters comprising the hidden state transition probabilities and probabilities of the hidden states emitting the observation states. Training efficacy of the HMM depends on the observation sequence used for the training. Each unique observation sequence results in a unique set of optimised HMM parameters due to the training process converging to a local optimum rather than a global one and therefore the selection of observation sequences for HMM training requires some attention.

Outputs: As a result of this modelling step, one hidden Markov model is generated for each unique observation of a sequence of workpiece states, i.e. unique task scenarios. Therefore, several HMMs could be generated for one task where each HMM model represents a different scenario within the same task. This step completes the digitisation process of the manual manufacturing task where each unique observation is modelled using an HMM model. These models could be saved in .CSV files and can be retrieved on demand for manufacturing knowledge extraction and decoding.

Step 4: Extract

Purpose: To analyse all the HMM models generated in the modelling step for a task and extract the human action sequences responsible for any given task scenario (sequence of workpiece change states) whether or not the scenario is observed (Figure 31).

Description: After the digitisation of a manual manufacturing task into digital human-workpiece interaction models using HMM, the next step is the analysis of these models in order to decode the manufacturing knowledge embedded within them. However, before manufacturing knowledge is decoded for a task, the human action sequences responsible for different scenarios of the task must be identified and extracted.

Inputs: The input to the analysis step is the set of HMM models generated in step 3 for the digitised manual manufacturing task.

Methods: As described in the modelling step, all unique sets of human-workpiece interactions observed during several runs of the manual manufacturing task are modelled using hidden Markov modelling.

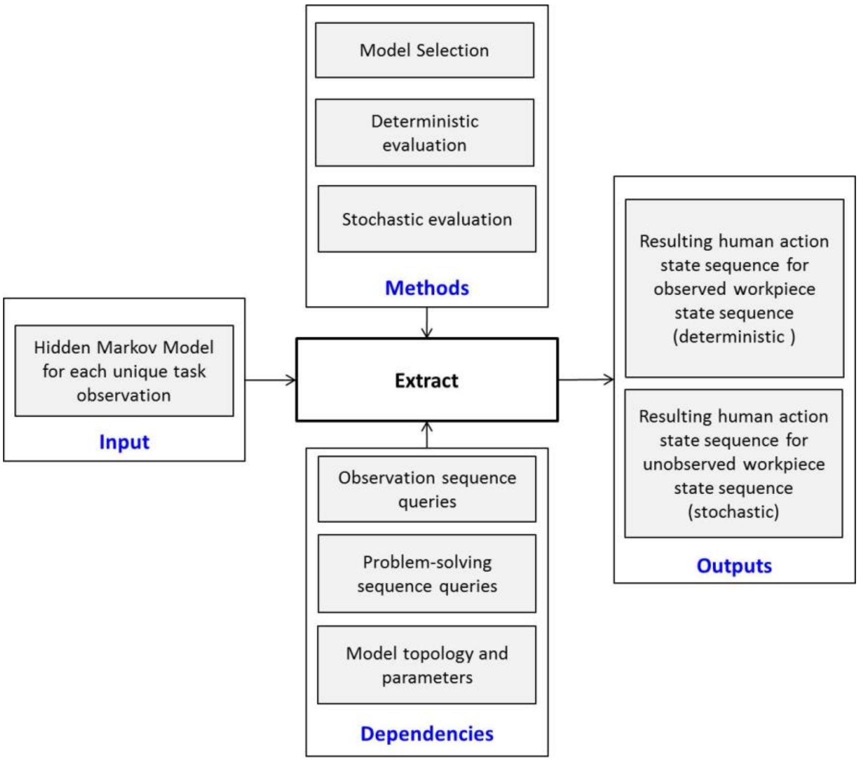


Figure 31: The 'Extract' step and its elements

Therefore, each task is associated with several HMM models. One of the main constituents of manufacturing knowledge is the sequence in which actions are selected and executed by the human during the task. Therefore, it is necessary to first extract the human action sequences responsible for specific workpiece state sequences representing various task scenarios.

Since each task is represented by multiple HMM models, one for each task scenario, a method to select the right HMM model for a given task scenario is needed. The selection method is deterministic if the given workpiece observation sequence (task scenario) has been observed before and is already represented by an HMM model. The right HMM model is therefore selected by merely comparing the given workpiece observation sequence with all observation sequences represented by their corresponding HMM models. The selection method is non-deterministic if the given workpiece sequence has not been observed before and an HMM model for it does not exist. In this case the HMM model that is most likely to represent the given workpiece observation sequence is stochastically obtained. Once the HMM model is identified for the given task scenario, the next step is to extract that human action sequence that is most likely responsible for that scenario.

Dependencies: The selection of the right HMM model to extract the human action sequences for a given task scenario (workpiece observation sequence) depends on the task scenario itself. This implies that a strong dependency exists between the workpiece states to human action states, as is the case with any manual manufacturing task. The same is true for task scenarios involved in problem-solving sessions. Also, the HMM model topology as discussed in section 5.3.2 influences the extraction of human action state sequences because the number of hidden states and observed states could be different for different task scenarios.

Outputs: The output of the extraction step is the sequence of human action states that is obtained by deterministic or stochastic means for the generation of the given task scenario represented by its workpiece change sequence. Human action sequences for different combinations of workpiece change sequences can be extracted in order to cover all task scenarios and decode the manufacturing knowledge associated with these scenarios in the next ‘decoding’ step.

Step 5: Decode

Purpose: To decode the different manufacturing knowledge constituents associated with the human action sequences extracted in step 4 (Figure 32).

Description: Manufacturing knowledge is embedded within the actions performed by the human during the task. There are several constituents of this knowledge such as task strategy, timing and nature of human gestures, mechanics of human body motion and workpiece manipulation techniques used during the execution of the task. These knowledge constituents can be decoded from the human action sequences extracted in step 4.

Inputs: Human action sequences responsible for given workpiece change sequences during a manual-manufacturing task.

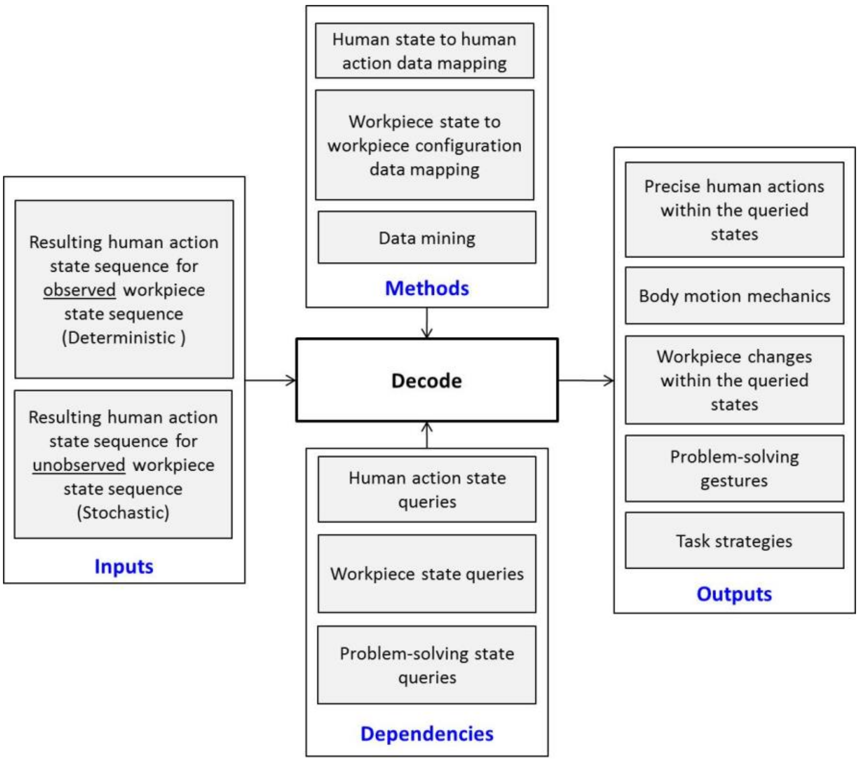


Figure 32: The 'Decode' step and its elements

Methods: From the sequence of human action states and the resulting workpiece observation states, each state must be mapped to its actual human skeletal motion data and workpiece progress data respectively. From these two sets of data the knowledge constituents are decoded. For example, from the human action sequence, the strategy adopted by the human to execute the task can be obtained. Similarly, from the human skeletal motion data, motion mechanics such as speed and acceleration of human hand gestures, angle between upper and lower limbs, glance angle, body orientations, etc. can be obtained. The resolution of a problem during the task can also be decoded by observing the effects of unusual actions taken during those sessions on the workpiece and the contribution of those actions to successfully solve the problem.

Dependencies: In order to decode the knowledge constituents, the right human action sequence for the given workpiece observation sequence (query) must be provided.

Outputs: The output of the decoding step is a set of manufacturing knowledge constituents decoded from the human action sequences. This data being in digital form can then be converted into multiple media so that it could be reproduced and transferred in the next step of the framework.

Step 6: Reproduce

Purpose: To reproduce the manufacturing knowledge constituents extracted and decoded using the first 5 steps of the framework (Figure 33).

Description: The manufacturing knowledge associated with the captured manual manufacturing task can be reproduced by rendering the knowledge constituents decoded in the previous step. The human actions required to execute the task successfully can be reproduced graphically using tools such as animations with the relevant knowledge constituents annotated within the animation during the corresponding times of the task.

Inputs: Knowledge constituents such as precise human gestures, mechanics of movements performed, and task strategies adopted during the execution of the task.

Methods: Graphical rendering of the execution of the task including human actions and workpiece progress can be reproduced using multiple media. The simplest form of rendering is 2D animation on the computer screen annotated with the knowledge constituents such as task strategy in the form of action sequence, human's body postures, motion speed and acceleration values, problem-solving gestures, etc.

Dependencies: In order to reproduce a manual manufacturing task with knowledge constituents annotated, the original capture step must record all task details such as all human actions and the corresponding changes to the workpiece. However, due to various constraints of the 'Capture' step such as low resolution of the Kinect, it is difficult to capture the workpiece changes in certain orientations, especially if the workpiece is complex. Also problems such as occlusions do not allow the capture of all human actions in the task making some portions of the task not reproducible.

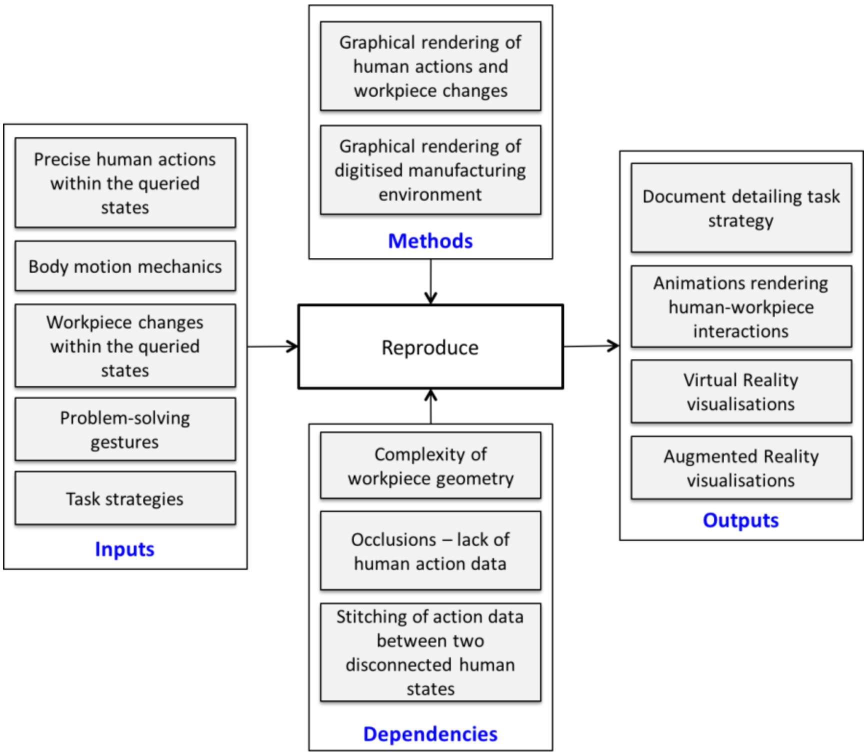


Figure 33: The 'Reproduce' step and its elements

When disconnected actions form part of a human action sequence, appropriate stitching methods are required to connect the disconnected actions together to render the unified human action sequence.

Outputs: The knowledge extracted and decoded using this framework can be stored in the form of documents such as task manuals or as videos of animations demonstrating the task. Since all the knowledge constituent data is digital, virtual and augmented reality tools could be used to reproduce the knowledge in virtual task space for task simulation or augmenting this knowledge over the real manufacturing environment in real task space. Such reproduction of knowledge helps in applying the digitisation framework to applications such as ergonomic analysis, human error analysis, skill training and to inform the design and process of automating manual manufacturing tasks.

5.3 Introduction to Hidden Markov Models

5.3.1 Need for modelling

Digitisation of manufacturing knowledge associated with a manual manufacturing task involves not only the capture and record of human-workpiece interactions in digital form but also the representation of this digital data in a manner that can be analysed and reproduced as required. Another objective of the digitisation process is to extract the hidden or implicit aspect of manufacturing knowledge such as human skill and decode this knowledge so that it can be explicitly documented and passed on from one human to another.

Human-workpiece interactions during a manual manufacturing task can be considered as a sequential time series of human actions and changes in workpiece configurations. In the segmentation phase, this continuous human-workpiece interaction data is segmented into discrete human action states and workpiece observation states, in which each human action state is made up of continuous human motion data. In these interactions, the changes undergone by the workpiece, denoted by the workpiece states are distinctly observable whereas the human action states, embedded with action and control skills, are not distinctly observable. The HWI theory also illustrates that any human action on the workpiece is influenced by the outcome of the workpiece from the previous human action.

Therefore, a modelling tool is needed that can represent a time series of human-workpiece interaction data over long and repeated observations of a manual manufacturing task and one that can also take into account the hidden nature of human actions and the dependence of human action on the outcome of the workpiece based on the previous human action. The modelling tool must also be able to identify the human action sequences responsible for all the possible workpiece observation sequences in order to extract and decode the different constituents of manufacturing knowledge.

These requirements form the basis of selecting a semi-supervised machine learning technique that can represent (1) multiple long time series of multivariate data as a state machine, (2) hidden input states that influence the observable output states and (3) incorporate the dependencies of hidden input states over the previous output states and the previous hidden input states. The human-workpiece interactions can be considered as a Markov process where the next human action state is dependent only on the previous human action state. Because the human action states are considered hidden, Hidden Markov Modelling (HMM) is a suitable modelling tool.

HMM, a stochastic framework for modelling a time series of multivariate observations is a widely used tool to analyse and predict time series phenomena. There is a large volume of literature produced on the design and use of HMM to a broad range of pattern recognition tasks. A classical reference to HMM can be found in Rabiner and Juang (1986). The first practical application of HMM is based on the work of Rabiner (1989) for speech recognition. Since then, HMM has been used to model and analyse time series data including, Electro-Cardiogram (ECG) analysis (Coast et al., 1990), face detection and recognition (Nefian and Hayes, 2000), gene finding in DNA (Cawley and Pachter, 2003), stock market forecasting (Hassan et al., 2007) and human activity detection (Sung et al., 2012) among others.

An HMM model is a statistical Markov model in which the system being modelled is assumed to be a Markov process with unobserved (hidden) states. A Markov process is a process, which moves from one state to the next depending only on the previous ‘n’ states. This process is called ‘n’ order Markov process where ‘n’ is the number of past states affecting the choice of the next state. The simplest Markov process is the first order process where the choice of the next state depends only on the previous state (n = 1).

In a regular Markov model, the state is directly visible to the observer, and therefore the state transition probabilities are the only parameters. In a hidden Markov model, the state is not directly visible, but output, dependent on the state, is visible. The transitions between hidden states are governed by a Markov chain and the emissions from each state are governed by a distinct probability distribution. The principal use of HMM is the determination of the most likely sequence of hidden states that could have generated a given sequence of observations. This problem usually does not have an exact solution therefore the idea is to find the hidden state sequence that would have generated the observed sequence with the highest probability.

In speech recognition, the HMM try to match a pattern of sound frequencies to predefined words based on the highest probability of a word matching a sequence of sound frequencies (Rabiner, 1989). HMM is also been widely used for Human Activity Recognition (HAR) from a continuous series of video or static image data. In HAR, human motion features such as body silhouettes and joint angles and human motion characteristics such as motion speed and acceleration are extracted using techniques such as Principal Component Analysis (PCA) and these motion features are mapped to predefined activities such as walking, running, jumping, idling, etc. to recognise human activities within a video or image sequence. HMM is used to match the human motion feature patterns to the activity datasets and select the most probable activity that would match the motion feature pattern (Uddin et al., 2010).

The main ability that this research aims to enable is to extract human action sequences that are responsible for specific workpiece state change sequences during a manual manufacturing task. The workpiece state changes are considered observable due to these being conspicuous whereas the human action states are considered hidden because within these actions are implicit human skills that are not directly observable. Therefore, HMM is used to model the hidden human states and observable workpiece states and map any workpiece state sequence to a human action sequence to extract and decode manufacturing knowledge associated with the manual tasks.

5.3.2 Definition of HMM

The structure of an HMM model contains states and observations. An HMM model is defined as set (S, O, π, A, B), where $S = \{s_1, s_2 \ldots, s_n\}$ is a finite set of $'n'$ states, $O = \{o_1, o_2 \ldots, o_m\}$ is a vocabulary of 'm' possible observation symbols or states, $\pi = \{\pi_i\}$ are the initial state probabilities, $A = \left\{a_{i_j}\right\}$ is the state transition matrix, $B = \{b_i(o_k)\}$ is the emission matrix.

$\lambda = (\pi, A, B)$ is used to denote an HMM model where

π_j is the probability that the system starts at state *j* at the beginning

a_{ij} is the probability of going to state *j* from state *i*

$b_i(o_k)$ is the probability of "generating" symbol o_k at state *i*

It is assumed that the state machine emits a symbol and starts to jump to a new state at the same time. Time t is discrete and starts with 1. Each probability in the state transition matrix and in the emission matrix is time invariant, i.e. the matrices do not change over time as the system evolves. This condition is often unrealistic but reasonably acceptable in this research because the manufacturing task is a known sequence of workpiece observations and the probabilities of workpiece state transitions and workpiece observations as a result of human actions would not normally change over the duration of a task.

Because HMM is a stochastic modelling technique, each model must adhere to the following obvious probability constraints:

$$\sum_{i=1}^{n} \pi_i = 1$$

$$\sum_{j=1}^{n} a_{ij} = 1 \; for \; i = 1, 2, \dots, n$$

$$\sum_{k=1}^{m} \boldsymbol{b_i(o_k) = 1 \; for \; i = 1, 2, \dots, n \; where \; n = number \; of \; hidden \; states \; and \; m = number \; of \; observation \; states}$$

[Equation 5-1]

In common applications of HMM, the internal states of the model are not observable, thus the states are said to be hidden. Only the emitted symbols (observations) can be observed. The goal is to extract some information about the internal states from the model parameters and emitted symbols. The emitted symbols in this case are the observable workpiece states and the internal states are the human action states in which implicit constituents of manufacturing knowledge are hidden.

Example of an HMM model: Consider the following state machine with two hidden states 'A' and 'B' and each state emitting one of the two observation symbols '0' or '1'.

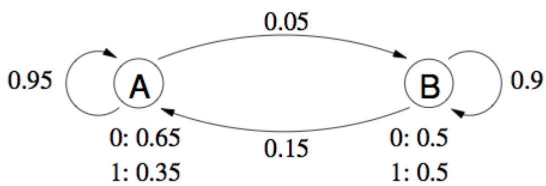


Figure 34: Example state machine

The HMM model that represents the above state machine (Figure 34) is $\lambda = (\pi, A, B)$ in which input states $S = \{A, B\}$ and observations states$V = \{0, 1\}$. The HMM model parameters are:

- State transition matrix $A = \begin{pmatrix} 0.95 & 0.05 \\ 0.1 & 0.9 \end{pmatrix}$

- Emission matrix $B = \begin{pmatrix} 0.65 & 0.35 \\ 0.5 & 0.5 \end{pmatrix}$

- Initial probability matrix $\pi = (\, 0.5 \quad 0.5)$

5.3.3 Standard HMM implementation

There are three fundamental problems that HMM can be applied to solve. HMM is of practical interest here mainly because all these problems can be solved in reasonably quick time for long sequences. The first two are pattern recognition problems whereas the third is a model learning problem. The 3 problems are: (1) Finding the probability observing a given observation state sequence given an HMM model ('evaluation problem'), (2) finding the hidden state sequence that most likely generated the given observation state sequence ('decoding problem') and (3) re-estimating the HMM model parameters so that the model is optimised given an observation state sequence ('learning problem').

Problem 1: Evaluation

Given the HMM model $\lambda = (\pi, A, B)$ and a sequence of observation states O, find the probability of observing the observation state sequence from the HMM λ also denoted as $P(O|\lambda)$.

Consider a situation in which a system is modelled using multiple HMM models each representing a different system scenario, i.e. a different sequence of observation states. Therefore, given an observation state sequence, the evaluation problem aims to find the HMM model that is most likely to represent that sequence. Such a problem is encountered in speech recognition where a large number of HMM models exist, each representing a particular word. An observation sequence is formed from the phonetics of a spoken word, which is recognised by identifying the most probable HMM model representing the word.

In this research, all unique workpiece state sequences (task scenarios) observed for a manual manufacturing task will be modelled using distinct HMM models. Given a task scenario, the evaluation problem is to find the HMM model that most likely represents it. Once the HMM model is found, the manufacturing knowledge embedded within the human action states within the model can be extracted and decoded.

The evaluation problem is solved by using the 'Forward' or 'Backward' algorithm. Both algorithms are similar in nature and are used to compute the probability of an observation state sequence given an HMM model. Since the 'Forward' algorithm is used in this research, it is described below.

Forward Algorithm

This algorithm consists of 3 main steps, namely, initialisation to find the initial partial probability, forward recursion to find intermediate partial probabilities and termination to find the final probability as the sum total of all partial probabilities.

The given observation sequence is $O = \{o_1, o_2, \dots, o_T\}$, a is the probability in state transition matrix A and b is the probability in the observation emission matrix B.

Initialisation: Determine the first partial probability $\alpha_t(i)$ for occurrence of state i at time $t = 1$.

$$\alpha_1(i) = \pi_i b_i(o_1) \; for \; 1 \le i \le n$$

[Equation 5-2]

Forward recursion: For each time step t, the partial probability α_t is determined for each state i. This partial probability is the sum of all probabilities of all possible forward paths leading to that state. Recursively, a partial probability of a state at time t can be determined by using partial probability of that state at time $t-1$.

$$\alpha_{t+1}(i) = \left(\sum_i \alpha_t(i) a_{ij}\right) b_j(o_{t+1}) \; for \; 1 \le i \le n, 1 \le j \le n \; and \; 1 \le t \le T-1$$

$$where \; n = number \; of \; hidden \; states \; and \; T = total \; time$$

[Equation 5-3]

Termination: The final probability $P\,(O \mid \lambda)$ of the given observation state sequence O given the HMM model λ is the sum of all partial probabilities α computed in the previous two steps for time $t = T$.

$$P(O|\lambda) = \sum_{i} \alpha_T(i)\ for\ 1 \leq i \leq n$$

[Equation 5-4]

By running the 'Forward' algorithm on all HMM models, the model that returns the highest final probability $P(O|\lambda)$ is the one that most likely represents a given observation state sequence.

Problem 2: Decoding

Given the HMM $\lambda = (\pi, A, B)$ and the observation sequence $O = \{o_1, o_2, \dots, o_T\}$ the goal is to compute the most likely sequence of hidden states that produced the observation sequence O, i.e. to extract the hidden state sequence $S = \{s_1, s_2, \dots, s_T\}$ which maximises $P(S|O)$.

In this research, it is essential to find the sequence of human action states that are responsible for any given workpiece state sequence so that the hidden aspect of manufacturing knowledge associated with the task, such as human action and reaction skills, can be extracted and decoded.

The Viterbi algorithm can be used for this purpose, which is similar to the 'Forward' algorithm. The difference in the Viterbi algorithm is that in the final step, instead of summing up the partial probabilities, the highest probability among the partial probabilities is chosen. Then the hidden state sequence is obtained by back tracking the path that travels through the highest partial probability from $t = T\ to\ 1$.

Viterbi Algorithm

This algorithm consists of 3 main steps, namely, initialisation to find the initial partial probability, forward recursion to find intermediate partial probabilities storing each state with the highest intermediate partial probability and termination to find the final probability which is the highest of all partial probabilities.

The given observation sequence is $O = \{o_1, o_2, \dots, o_T\}$, a is the probability in state transition matrix A and b is the probability in the observation emission matrix B.

<u>Initialisation:</u> Determine the first partial probability $\delta_t(i)$ for occurrence of state i at time $t = 1$.

$$\delta_1(i) = \pi_i b_i(o_1)\ for\ 1 \leq i \leq n$$

[Equation 5-5]

<u>Forward recursion:</u> For each time step t, the partial probability δ_t is determined for each state i. This partial probability is the highest of all probabilities of all possible forward paths leading to that state. Recursively, a partial probability of a state at time t can be determined by using partial probability of that state at time $t - 1$.

$$\delta_t(i) = \max_i\left(\delta_{t-1}(i)a_{ij}\right) b_j(o_t)\ for\ 1 \leq i \leq n, 1 \leq j \leq n\ and\ 2 \leq t \leq T$$

$$where\ n = number\ of\ hidden\ states\ and\ T = total\ time$$

[Equation 5-6]

For backtracking the best path at the end, the state associated with the highest partial probability is stored in an array at each step t.

<u>Termination:</u> The final probability $P\,(o_T\mid\lambda)$ of the given observation state sequence O given the HMM model λ is the highest of all partial probabilities δ computed in the previous two steps for times $t = T$.

$$P(o_T|\lambda) = \max_i\left(\delta_T(i)\right) for\ 1 \leq i \leq n$$

[Equation 5-7]

The state s that is associated with the highest final probability $P(o_T|\lambda)$ is then added as the last state to the array of states that represents the best state path that was constructed in the forward recursion phase. Therefore, for a given observation sequence O, the most probable hidden state sequence S is determined.

Problem 3: Learning

The evaluation and decoding problems involve either a measurement of an HMM model's relative suitability or an estimate of what the underlying model is representing, i.e. to determine what might have `really happened'. It can be seen that both the problems depend upon foreknowledge of the HMM parameters - the state transition matrix A, the emission matrix B, and the initial vector π. There are, however, many circumstances in practical situations in which these are not directly quantifiable, and have to be estimated. Usually, the HMM topology is relatively well designed (by an expert or obtained heuristically) but reasonably true determination of the state transition and observation emission probabilities is needed. The process of re-estimating the transition and emission probabilities of an HMM model is known as 'the learning problem'.

Given the observation state sequence O based on which the HMM model λ is generated, learning is an iterative process in which the parameters of the model are re-estimated by optimising them with each learning cycle (iteration) until the parameters cannot be optimised anymore, i.e. when a convergence is reached. The convergence is reached when $P(O|\lambda)$, the probability of the model λ to emit the observation state sequence O, does not increase with any more iterations. There are 2 main learning algorithms that are commonly reported in literature, namely, 'Viterbi Training' also known as 'Segmental K-Means' and 'Baum Welch', also known as 'Forward-Backward' algorithm.

<u>'Viterbi Training' algorithm:</u> In this algorithm, the parameters A and B of the HMM model $\lambda = (\pi, A, B)$ are adjusted to maximise $P(O, I|\lambda)$ where I is the sequence of hidden states that is responsible for emitting the observation state sequence O. Firstly, this algorithm requires the hidden state sequence I to be known beforehand to re-estimate the HMM model parameters. Secondly, by re-estimating the parameters based on the specific combination of the hidden state sequence I and the observation state sequence O, the resulting HMM model

becomes highly selective for that combination and loses its ability to predict hidden state sequences for other observation state sequences.

<u>'Baum-Welch' algorithm:</u> In this algorithm, the parameters A and B of the HMM model $\lambda = (\pi,\ A,\ B)$ are adjusted to maximise $P(O|\lambda)$ where O is the sequence of observation states based on which the HMM model λ was generated. Unlike the Viterbi algorithm, the Baum-Welch algorithm does not require the hidden state sequence I to be known beforehand. It also computes $P(O|\lambda)$ by summing up the probabilities $P(O, I|\lambda)$ for all possible hidden state sequences I rather than a specific sequence. Therefore, the resulting HMM model does not lose its ability to predict hidden state sequences for other observation state sequences.

Choice between the two learning algorithms

It is important to note that none of the two learning algorithms give globally optimum HMM model parameters but converge to a local optimum instead (Dempster et. al, 1977, Boodidhi, 2011). This means that for each observation state sequence and for each new set of starting HMM parameters, both the algorithms will converge to new optima resulting in a different resulting HMM model each time.

The Viterbi training algorithm requires the HMM model to be initialised with reasonably appropriate parameters rather than with random numbers. To arrive at the optimised HMM model, this algorithm does not consider all possible hidden state sequences. Therefore, it is not computationally expensive and executes relatively faster (Rodrıguez and Torres, 2003). Therefore, in cases where the HMM model can be appropriately initialised, the hidden state sequence that emits the observation state sequence is known and the HMM model is required to run in real-time (such as in speech recognition), the Viterbi training method is preferred.

The Baum-Welch algorithm does not need any model initialisation but just non-zero random values as starting HMM model parameters. Also, the algorithm exhaustively uses all the available data to produce robust and optimal estimates of model parameters and is therefore computational expensive and slower. In this research, as the hidden state sequences for all observation state sequences may not be known and the HMM is not required to run in real-time the ‘Baum Welch’ algorithm is preferred and used.

Baum-Welch algorithm explained

The Baum-Welch algorithm (Baum et al., 1970) consists of 4 main steps, namely, forward recursion, backward recursion, determination of temporary probability variables and re-estimating of the revised HMM parameters π, A and B. Before the algorithm is executed, the HMM model $\lambda = (\pi, A, B)$ is initialised with random values for parameters π, A and B. The algorithm updates these parameters iteratively until convergence is reached.

<u>Initialisation:</u> Determine the first partial probability $\alpha_1(i)$ for occurrence of state at time $t = 1$ and the last partial probability $\beta_T(i)$ at time $t = T$

$$\alpha_1(i) = \pi_i b_i(o_1)\ for\ 1 \leq i \leq n$$

$$\beta_T(i) = 1$$

[Equation 5-8]

<u>Forward Recursion:</u> For each time step t, the partial probability α_t is determined for each state i. This partial probability is the sum of all probabilities of all possible forward paths leading to that state. Recursively, a partial probability of a state at time t can be determined by using partial probability of that state at time $t - 1$.

$$\alpha_{t+1}(j) = \left(\sum_i \alpha_t(i) a_{ij}\right) b_j(o_{t+1})\ for\ 1 \leq i \leq n, 1 \leq j \leq n\ and\ 1 \leq t \leq T - 1$$

$$where\ n = number\ of\ hidden\ states\ and\ T = total\ time$$

[Equation 5-9]

<u>Backward Recursion:</u> For each time step t, the partial probability α_t is determined for each state i. This partial probability is the sum of all probabilities of all possible forward paths leading to that state. Recursively, a partial

probability of a state at time t can be determined by using partial probability of that state at time $t-1$.

$$\beta_t(i) = \sum_j \beta_{t+1}(j) a_{ij}\, b_j(o_{t+1})\ for\ 1 \le i \le n, 1 \le j \le n\ and\ 1 \le t \le T-1$$

$$where\ n = number\ of\ hidden\ states\ and\ T = total\ time$$

[Equation 5-10]

<u>Determination of temporary probability variables:</u> Using the forward and backward partial probabilities, the temporary probability variables γ and ξ are calculated as follows:

$$\boldsymbol{\gamma_t(i)} = \frac{\boldsymbol{\alpha_t(i)\beta_t(i)}}{\boldsymbol{\sum_{j=1}^{n} \alpha_t(j)\beta_t(j)}}$$

$$\xi_t(i,j) = \frac{\alpha_t(i) a_{ij} \beta_{t+1}(j) b_j(o_{t+1})}{\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_t(i) a_{ij} \beta_{t+1}(j) b_j\,(o_{t+1})}$$

[Equation 5-11]

where $\boldsymbol{\gamma_t}(\boldsymbol{i})$ is the probability of the system being in state i at time t, given the observation state sequence O and HMM λ and $\xi_t(ij)$ is the probability of the system being in state i and j at times t and $t+1$ respectively given the observation state sequence O and HMM λ.

<u>Re-estimation of updated HMM parameters:</u> Using the temporary probability variables γ and ξ, the revised HMM parameters π', A' and B' are computed:

$$\bar{\pi} = \gamma_1(i)$$

$$\overline{a_{ij}} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t\,(i)}$$

$$\bar{b_i}(k) = \frac{\sum_{t=1}^{T} \delta_{o_t,o_k} \gamma_t(i)}{\sum_{t=1}^{T} \gamma_t(i)}$$

[Equation 5-12]

Note that the equation for $\overline{b_\iota}(k)\ the$ summation in the nominator is only performed over observation states correspond to o_k.

Using the revised π', A' and B', a new iteration of the algorithm is executed until convergence is achieved.

5.3.4 Proposed application of HMM in the framework

The main goal of this research is to extract and decode the manufacturing knowledge associated with manual manufacturing tasks, especially the hidden aspects of that knowledge such as human action and reaction skills. By observing and segmenting the human-workpiece interactions during the task, the human action states (hidden states) and workpiece states (observation states) are modelled into an HMM model $\lambda = (\pi,\ A,\ B)$. All unique task scenarios including the ones with problem solving sessions are represented by their individual HMM models $(\lambda_1, \lambda_1, \dots, \lambda_N\ for\ N\ unique\ task\ scenarios)$. For each HMM, the parameters are assigned using inputs from the task experts or randomly when such inputs are not available. The 3 problems of evaluation, decoding and learning are applied to this research in the following order for digitisation of each task:

1. Learning

HMM models representing the unique scenarios of a task are optimised using the Baum-Welch algorithm before being used to extract and decode the manufacturing knowledge from that task. This ensures that the models are reasonably true in representing their respective task scenarios.

2. Evaluation

Given a task scenario with its unique workpiece state sequence, the 'Forward' algorithm is used to find the HMM model that most likely represents that scenario. For example, if a task is represented by 2 HMM models, one for a normal scenario and one for a problem-solving scenario, it is expected that for a given workpiece state sequence resembling a normal task scenario, the normal task scenario HMM will be returned by the algorithm. Once the most likely HMM model is identified, it could be used to extract and decode the task knowledge.

3. Decoding

From the identified HMM model, the Viterbi algorithm is used to determine the most likely sequence of human action states (hidden states) that are responsible for the given sequence of workpiece states (observation states). Once the sequence of human action states is identified, the knowledge constituents within those states such as strategy adopted by the human, mechanics of human body motion, action choices made by the human in response to the changing configurations of the workpiece, etc. can be extracted. Once all this information is available, the human action and reaction skills can be made explicit and reproduced using multiple media such as animations for skill demonstration or converted to workpiece manipulation and control programmes to develop an automation solution for the task.

5.4 Chapter summary

This chapter proposes the design of the framework for digitisation of manual manufacturing task knowledge and provides the major steps, methods and tools needed to develop the framework. It also highlights the inputs, outputs and dependencies of each step, an implementation of which is presented in the next chapter. A brief introduction to HMM and its application in modelling human-workpiece interactions is also presented.

CHAPTER 6

6 IMPLEMENTATION OF THE DIGITISATION FRAMEWORK

This chapter presents a step-by-step implementation of the proposed framework for digitisation of manual manufacturing task knowledge and task environment. The digitisation process of the manufacturing task environment described in this chapter is presented in two journal papers and a conference paper (See 'List of Publications').

This chapter aims to achieve the following objectives:

- ❖ Explain the rationale behind the choice of example task and the framework implementation process.
- ❖ Describe the experiment setup needed for framework implementation.
- ❖ Present the framework implementation process in a step-by-step manner detailing the methods, tools and techniques used.

6.1 Overview

The proposed digitisation framework described in the previous chapter (Chapter 5) is implemented using both off-the-shelf and bespoke ICT methods and tools for human action and workpiece progress capture during a manual task. This combination makes this research unique in the current literature landscape and the low cost point of using gaming interface sensors makes it potentially attractive for the industry. The 6 steps of the framework implemented to digitise the manufacturing knowledge embedded within two simplified, lab-scale manufacturing tasks and the results are presented and discussed.

6.2 Choice of example task

The proposed digitisation framework starts by capturing human-workpiece interactions and the task environment data and then progresses to represent this data into digital models. These models are subsequently analysed to extract and decode the manufacturing knowledge embedded within the task and the extent of knowledge that can be extracted depends on the amount and quality of task data captured. Data capture is therefore a vital step in the framework and it is implemented in this research using gaming interface technology such as Kinect sensors.

These sensors are primarily meant for the gaming industry and therefore do not have the capabilities required to capture complex manufacturing tasks, such as high fidelity, high resolution imaging. At the same time, the advantages of these sensors, such low cost, portability, effective full-body human skeletal tracking and availability of the 3D imaging are compelling enough to be used in this framework. Therefore, Lego blocks assembly task is chosen as an example of a simplified lab-scale manufacturing task. In this task, the workpieces individually and when assembled have simple geometries which makes their real-time recognition and tracking quite simple and reliable despite the low resolution (640 x 480 pixels) of the current generation of Kinect sensors. The human actions required to execute this assembly task are also not complex and therefore can be reliably captured in real-time.

6.3 Task description

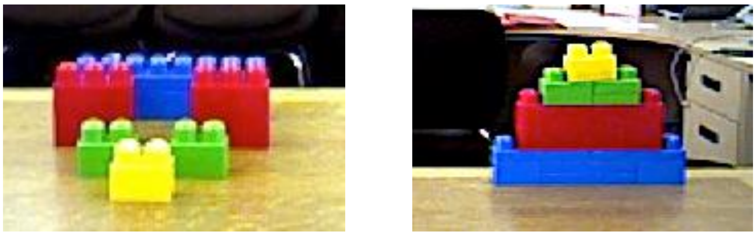
6.3.1 Task elements

There are 3 main elements in the implementation of the digitisation framework, namely, the human, the workpiece and the task.

Human: The human performing the assembly task is assumed to be an expert based on his skills, training and past experiences of performing the task repeatedly. He is assumed to know the solutions to all the known problems in the task and have the wherewithal to solve new and unforeseen problems that might arise during the task.

Workpiece: The workpiece or the set of workpieces in this task are the blocks of Lego in standard shapes, sizes and colours that are assembled together to form a specific structure and in a sequence best known to the task expert.

Task: The task involves manoeuvring the workpieces from their initial positions to the assembly zone and assembling them together to form the final desired shape. The human uses his two hands to grasp, manoeuver and place the workpieces in the best-known sequence. The following figure (Figure 35) illustrates the workpieces used in the task and the final assembled structure, which is the final goal of the task.



(a) (b)

Figure 35: (a) Task workpiece components and (b) Final assembled workpiece

6.3.2 Task setup

The task setup consists of the task expert who performs the task, the workpiece components for assembly and a workstation (the table) on which the assembly is performed (Figure 36). The task is captured by a Kinect sensor, mounted on a tripod at a height of 1.5m from the floor and at a distance of 1.5m from the assembly workspace. The task environment is tightly controlled by ensuring that no other human apart from the one implementing the task is present, lighting conditions in the room are maintained unchanged for the duration of the task to avoid the effects of ambient light change on workpiece recognition and tracking, no sunlight is allowed to enter the room to avoid interference with the Kinect sensor and only the selected Lego blocks were manipulated during the task without the introduction of new ones.

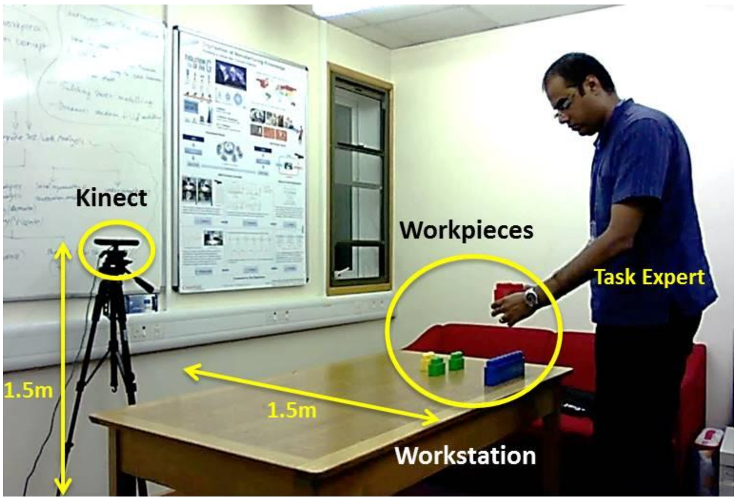


Figure 36: Task setup for digitisation of assembly task

6.3.3 Task rules

Two rules have been formulated and programmed into the task capture step to minimise uncertainty in the framework implementation process.

i. The assembly is executed within the pre-defined virtual box on the workstation. This rule cuts down the time taken by the task capture function by focussing the workpiece recognition and tracking function on a small area rather than the entire 3D space in the visible view of the Kinect sensor.
ii. Only one block is handled and manoeuvred at a time. This rule cuts the complexity of the real-time workpiece progress tracking function by recognising and tracking only one object at a time based on its colour and its changing spatial positions within the virtual box.

6.4 Framework implementation process

The 6 steps of the framework are implemented sequentially as described in section 5.2. Multiple runs of the task are performed to capture various scenarios under normal and problem solving conditions. Unique observations are selected, modelled and analysed to extract and decode the manufacturing knowledge embedded within the task.

Step 1: Capture

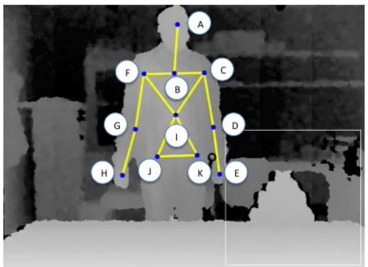
The Kinect sensor records the entire task performed by the human and the data produced by the sensor is processed using standard and bespoke software functions to capture and digitise human actions and the effects of those actions on the workpiece during the task. Sound is not captured in this implementation. Task capture is also sometimes referred to as task observation depending on the context in which it appears in this thesis.

Tracking human actions

A continuous stream of RGB and depth image frames (Figure 37) is produced by the sensor at the rate of 30 fps. A standard human skeletal tracking function, provided by an open source library called 'OpenNI', is used to track and record the 3D spatial positions of the 11 upper body joints of the human while performing the task. The lower body joints are ignored because they are occluded by the assembly workstation. For each image frame generated by the sensor, the skeletal joint positions are acquired using the following functions:

- kinect.getJointPositionSkeleton(user_id, SimpleOpenNI.SKEL_HEAD, head)
- kinect.getJointPositionSkeleton(user_id, SimpleOpenNI.SKEL_NECK, neck)
- kinect.getJointPositionSkeleton(user_id, SimpleOpenNI.SKEL_LEFT_SHOULDER, l_shoulder)
- kinect.getJointPositionSkeleton(user_id, SimpleOpenNI.SKEL_LEFT_ELBOW, l_elbow)
- kinect.getJointPositionSkeleton(user_id, SimpleOpenNI.SKEL_LEFT_HAND, l_hand)
- kinect.getJointPositionSkeleton(user_id, SimpleOpenNI.SKEL_RIGHT_SHOULDER, r_shoulder)
- kinect.getJointPositionSkeleton(user_id, SimpleOpenNI.SKEL_RIGHT_ELBOW, r_elbow)
- kinect.getJointPositionSkeleton(user_id, SimpleOpenNI.SKEL_RIGHT_HAND, r_hand)
- kinect.getJointPositionSkeleton(user_id, SimpleOpenNI.SKEL_TORSO, torso)
- kinect.getJointPositionSkeleton(user_id, SimpleOpenNI.SKEL_LEFT_HIP, l_hip)
- kinect.getJointPositionSkeleton(user_id, SimpleOpenNI.SKEL_RIGHT_HIP, r_hip)

For example, in the first function the user being tracked is identified by the 'user_id' variable, the skeletal joint to be tracked is specified by the 'SimpleOpenNI.SKEL_HEAD' and the joint position is returned in the vector variable 'head'. The 3D coordinates (x, y, z) for each joint position are stored in the sequence of capture thereby acquiring the 3D motion data for the upper body of the human while he performs the assembly task.



(a)

(b)

Figure 37: (a) Depth image frame and (b) RGB image frame

Tracking workpieces

A bespoke depth and RGB image processing function is developed for workpiece recognition and tracking. This function looks for the presence of groups of screen pixels with the 4 workpiece colours (red, green, blue and yellow) simultaneously in the RGB image streams sent by the Kinect sensor. The search area for this function is limited to the virtual assembly box which is 200 screen pixels in length (x), 200 screen pixels in height (y) and 1000mm in depth (z). Once a coloured pixel block is identified, its 2D boundary and centre point are computed and highlighted on the screen as shown in Figure 38. By acquiring the boundaries and centre points of all the workpieces for each image produced by the sensor, the positions of individual workpieces as well as the progress of the assembled workpiece structure can be tracked in real-time.

Figure 38: Workpiece components identified and tracked

A sequence of images in which human skeletal motion and workpiece components are tracked during the assembly task are shown below. In Figure 39, the task is successfully completed without any problems (normal task scenario) whereas in Figure 40, a wrong assembly sequence is corrected by the task expert while successfully completing the task (problem solving scenario).

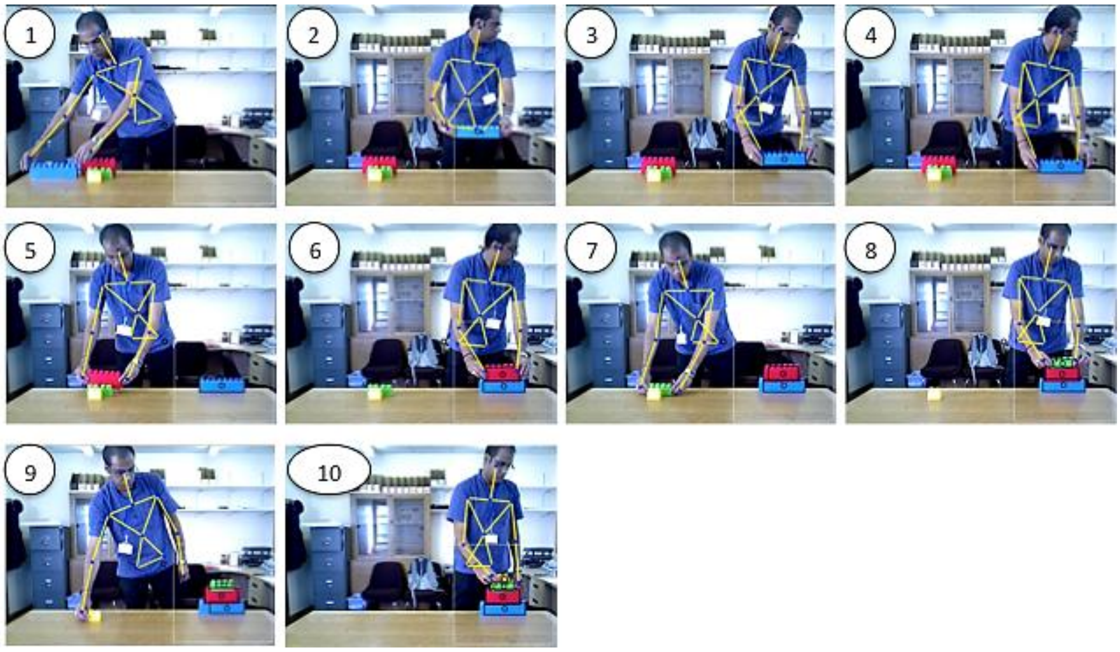


Figure 39: Sequence of images captured (normal scenario)

Figure 40: Sequence of images captured (problem-solving scenario)

Output

3D coordinates of human skeletal motion, 2D coordinates of workpiece positions and the assembly structure configurations are recorded in a spreadsheet along with the timestamps at which they were captured. The timestamps, stored as frame numbers, are the vital link between the human actions (represented by skeletal motion) and the corresponding workpiece changes (represented by individual workpiece positions and change in assembly structure configuration). A snapshot of the output spreadsheet is shown in Figure 41.

Frame	Red X	Green X	Blue X	Yellow X	Head X	Head Y	Head Z	Neck X	Neck Y	Neck Z	L_Hand X	L_Hand Y	L_Hand Z
1	320	320	320	320	531.09	69.04	2116.77	501.88	119.68	2185.85	386.58	241.52	2282.50
2	320	320	320	320	531.06	69.00	2109.73	501.36	119.80	2184.02	385.84	241.38	2284.12
3	320	320	320	320	530.83	68.86	2106.71	501.06	119.82	2182.97	384.62	240.44	2286.30
4	320	320	320	320	530.44	69.04	2106.20	500.52	119.89	2182.93	383.99	239.77	2285.05
5	320	320	320	320	529.88	68.21	2110.95	499.96	119.25	2184.38	383.66	239.42	2284.76
6	320	320	320	320	529.16	67.75	2113.34	499.29	118.81	2185.90	384.30	240.09	2287.74
7	320	320	320	320	528.41	67.53	2115.53	498.78	118.55	2186.75	383.80	241.26	2290.96
8	320	320	320	320	527.94	67.43	2122.12	498.90	118.51	2189.48	384.43	241.41	2291.24
9	320	320	320	320	526.46	66.89	2119.41	498.07	118.17	2189.27	383.87	242.02	2292.03
10	320	320	320	320	524.87	66.47	2124.06	497.08	117.91	2191.01	383.57	242.48	2292.58
11	320	320	320	320	523.18	65.94	2125.88	496.09	117.71	2191.98	383.38	242.73	2293.30

Figure 41: Snapshot of the spreadsheet that stores the captured raw data

Continuous data charts

The human skeletal motion data and workpiece change data can be visualised in the form of charts. Even before analysing the raw data any further, these charts provide useful information about the overall nature of the task and in some cases also about the sequence of actions that lead to task progress. The charts are shown in Table 1.

Table 1: Charts showing human skeletal motion (arm and elbow) and workpiece tracking from raw captured data

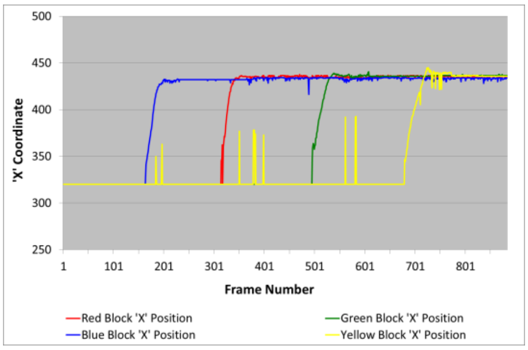


Chart 1: X-axis movement of the four workpiece blocks (red, green, blue and yellow)

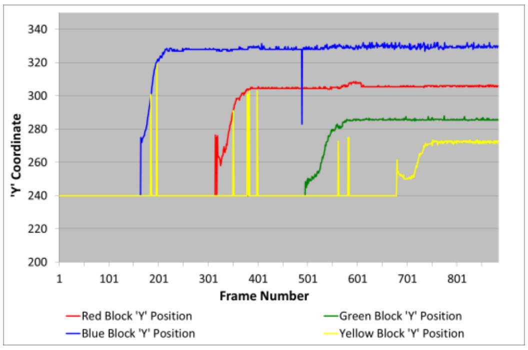


Chart 2: Y-axis movement of the four workpiece blocks (red, green, blue and yellow)

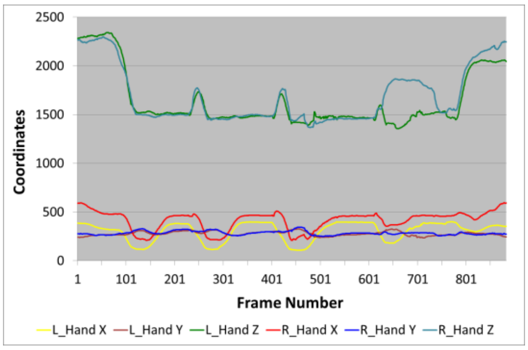


Chart 3: X, Y and Z axis movements of left and right hand joints of the human

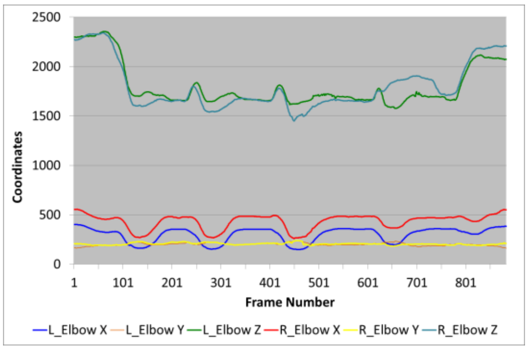


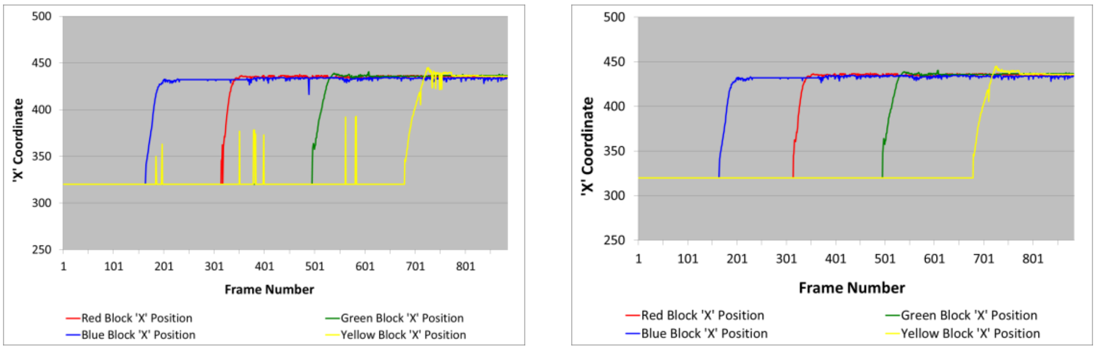
Chart 4: X, Y and Z axis movements of left and right elbow joints of the human

Filtering

The raw data contains noise which is unwanted data that has crept into the capture step due to errors in human skeletal tracking and/or workpiece tracking which in turn is due to factors such as low resolution of the Kinect sensor and occlusions affecting skeletal tracking. In this example, human skeletal tracking data does not have noticeable noise and the amplitude of the noise present will not have a considerable effect on the segmentation step. Therefore, this data is not filtered before segmentation.

However, in workpiece change data high amplitude noise can be seen in the tracking charts. These data points that abruptly increase in amplitude in comparison with the preceding and succeeding data and at unexpected positions during the task are filtered out using a simple threshold comparison algorithm. Using this algorithm, noise can be filtered online while the raw data is captured or offline, after the raw data is captured. In this example, the filtering is done offline and the results are shown in Table 2

Table 2: Workpiece tracking charts - before and after filtering



Workpiece tracking chart before and after filtering

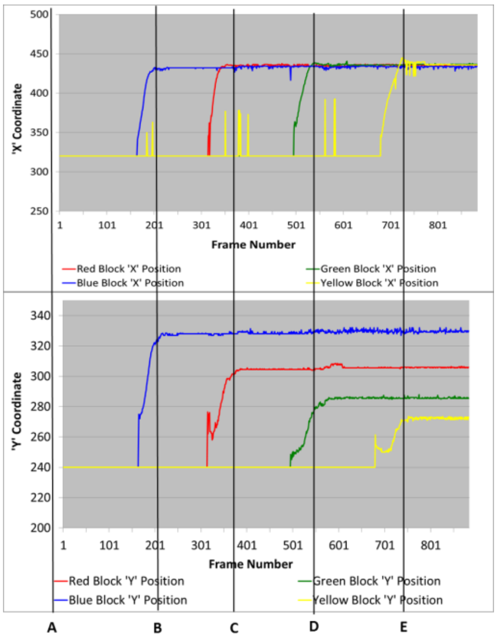
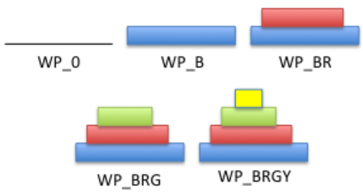
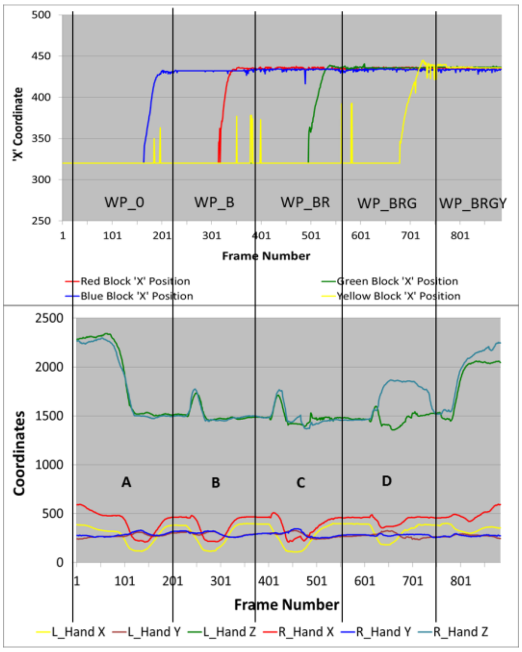
<u>Step 2 – Segment</u>

The filtered human action and workpiece change data is continuous in nature and must be segregated into discrete states to facilitate the hidden Markov modelling of the human-workpiece interactions.

According to the framework, segmentation of raw human action and workpiece data can be done by 3 techniques, namely, time-sampling, trajectory-sampling and workpiece-sampling. In this example, the workpiece undergoes prominent changes in shape and structure with every cyclical action of the human and therefore the workpiece-sampling segmentation technique is used.

In this technique, the workpiece tracking data is segmented at points where any one of the workpiece characteristics such as position, shape, dimension or configuration undergoes an abrupt change. In this example, these segmentation points are clearly seen in the workpiece tracking charts and both the workpiece tracking data and human skeletal motion data is segmented at these points. Workpiece states are generated at these segmentation points whereas human action states are the period between two workpiece states (Table 3).

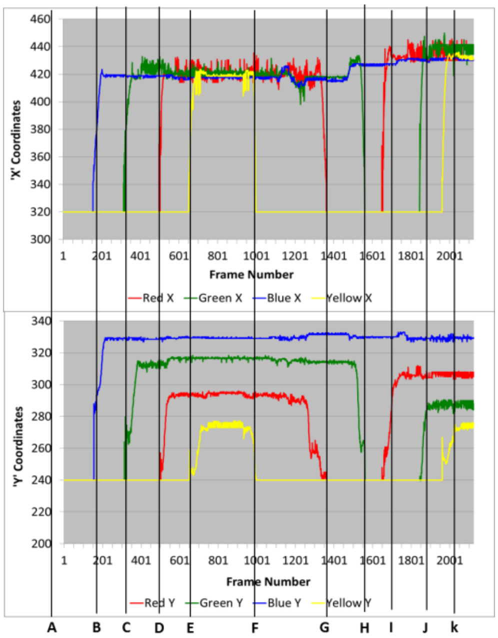
Table 3: Generation of human action and workpiece states (normal scenario)

<table>
<tr>
<td>
</td>
<td>
<table>
<tr><th>Label</th><th>Workpiece State</th></tr>
<tr><td>A</td><td>WP_0</td></tr>
<tr><td>B</td><td>WP_B</td></tr>
<tr><td>C</td><td>WP_BR</td></tr>
<tr><td>D</td><td>WP_BRG</td></tr>
<tr><td>E</td><td>WP_BRGY</td></tr>
</table>

</td>
</tr>
<tr><td>Workpiece tracking data segmentation</td><td>Resulting workpiece states</td></tr>
<tr>
<td>
</td>
<td>
<table>
<tr><th>Label</th><th>Workpiece State</th></tr>
<tr><td>A</td><td>H_A_B</td></tr>
<tr><td>B</td><td>H_A_R</td></tr>
<tr><td>C</td><td>H_A_G</td></tr>
<tr><td>D</td><td>H_A_Y</td></tr>
</table></td>
</tr>
<tr><td>Human tracking data segmentation</td><td>Resulting human action states</td></tr>
</table>

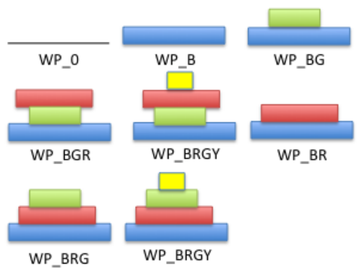
In order to capture another task scenario, a problem solving session was introduced into the task execution. The Lego blocks are assembled in the wrong sequence and subsequently corrected. The continuous human-workpiece interaction data from this task scenario is also filtered and segmented. Because of the presence of a problem-solving session in this particular task observation, more workpiece and human action states are generated as compared to the normal task scenario (Table 4).

The generated workpiece and human action states are discrete in nature. Each workpiece state is a snapshot of the workpiece at that particular point in time during the task. A human action state however is not a snapshot of the human pose at the point of segmentation but a repository of continuous human motion tracking data responsible for bringing the workpiece from its previous state to the next. But for discrete time space modelling purposes, the human action state is considered as a discrete state.

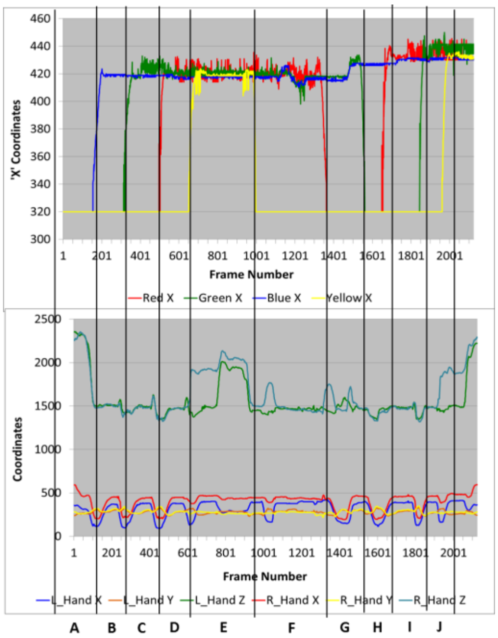
Table 4: Human action and workpiece states (problem-solving scenario)



Label	Workpiece State
A	WP_0
B	WP_B
C	WP_BG
D	WP_BGR
E	WP_BGRY
F	WP_BGR
G	WP_BG
H	WP_B
I	WP_BR
J	WP_BRG
K	WP_BRGY



Workpiece tracking data segmentation	Resulting workpiece states



Human tracking data segmentation

Label	Human State
A	H_A_B
B	H_A_G
C	H_A_R
D	H_A_Y
E	H_D_Y
F	H_D_R
G	H_D_G
H	H_A_R
I	H_A_G
J	H_A_Y

Resulting human action states

Step 3: Model

The discrete human action and workpiece states are now ready to be modelled using HMM. As described in section 5.3.4, the fundamentals of HMM are adopted from literature as-is but their interpretation is construed according to the needs of representing human-workpiece interactions in this framework. This interpretation of HMM is novel and is presented below.

HMM generation

HMM topology: A bespoke function is developed that takes the workpiece and human action states as inputs and generates the HMM topology (S, O, π, A, B), where $S = \{s_1, s_2 \dots, s_n\}$ is a set of $'n'$ human action states, $O = \{o_1, o_2 \dots, o_m\}$ is a set of 'm' workpiece states, $\pi = \{\pi_i\}$ are the initial state probabilities, $A = \left\{a_{i_j}\right\}$ is the state transition matrix, $B = \{b_i(o_k)\}$ is the emission matrix (Section 5.3.2).

The dimensions of π, A and B are dependent on the number of workpiece and human action states generated by the 'Segment' step. Initial probability values for these matrices were assigned in consultation with the task expert.

Unique HMM models: Each unique task observation is represented using its own HMM model. Therefore, ‘n’ different task captures will generate ‘n’ different HMM models and the goal is to model as many task scenarios so that knowledge embedded within the task can be extracted and decoded to the fullest extent. In this example, two unique observations are made and therefore two unique HMM models were generated; one for the normal task scenario and one for the problem-solving task scenario (Figure 42 and 40).

The workpiece state sequence observed for (A) - normal scenario is

$$O_A = \{WP\_B,\ WP\_BR,\ WP\_BRG,\ WP\_BRGY\}$$

The workpiece state sequence observed for (B) – problem-solving scenario is

$$O_B = \{WP\_B,\ WP\_BG,\ WP\_BGR,\ WP\_BGRY,\ WP\_BGR,\ WP\_BG,\ WP\_B,\ WP\_BR,\ WP\_BRG,\ WP\_BRGY\}$$

HMM models λ_A and λ_B are generated for the two unique workpiece state sequences O_A and O_B.

π	H_A_B	H_D_B	H_A_R	H_D_R	H_A_G	H_D_G	H_A_Y	H_D_Y
	0.7	0.02	0.08	0.02	0.07	0.02	0.07	0.02

A	H_A_B	H_D_B	H_A_R	H_D_R	H_A_G	H_D_G	H_A_Y	H_D_Y
H_A_B	0.01	0.20	0.56	0.01	0.10	0.01	0.10	0.01
H_D_B	0.60	0.01	0.10	0.03	0.10	0.03	0.10	0.03
H_A_R	0.10	0.01	0.01	0.20	0.56	0.01	0.10	0.01
H_D_R	0.10	0.03	0.60	0.01	0.10	0.03	0.10	0.03
H_A_G	0.10	0.01	0.10	0.01	0.01	0.20	0.56	0.01
H_D_G	0.10	0.03	0.10	0.03	0.60	0.01	0.10	0.03
H_A_Y	0.20	0.03	0.20	0.03	0.20	0.03	0.01	0.30
H_D_Y	0.10	0.03	0.10	0.03	0.10	0.03	0.60	0.01

B	WP_0	WP_B	WP_BR	WP_BRG	WP_BRGY
H_A_B	0.01	0.70	0.10	0.10	0.09
H_D_B	0.96	0.01	0.01	0.01	0.01
H_A_R	0.01	0.01	0.96	0.01	0.01
H_D_R	0.21	0.76	0.01	0.01	0.01
H_A_G	0.01	0.01	0.01	0.96	0.01
H_D_G	0.26	0.26	0.46	0.01	0.01
H_A_Y	0.01	0.01	0.01	0.01	0.96
H_D_Y	0.14	0.14	0.15	0.56	0.01

Figure 42: HMM model λ_A for normal scenario

π	H_A_B	H_D_B	H_A_R	H_D_R	H_A_G	H_D_G	H_A_Y	H_D_Y
	0.7	0.02	0.08	0.02	0.07	0.02	0.07	0.02

A	H_A_B	H_D_B	H_A_R	H_D_R	H_A_G	H_D_G	H_A_Y	H_D_Y
H_A_B	0.01	0.20	0.56	0.01	0.10	0.01	0.10	0.01
H_D_B	0.60	0.01	0.10	0.03	0.10	0.03	0.10	0.03
H_A_R	0.10	0.01	0.01	0.20	0.56	0.01	0.10	0.01
H_D_R	0.01	0.03	0.05	0.01	0.05	0.77	0.05	0.03
H_A_G	0.10	0.01	0.10	0.01	0.01	0.01	0.75	0.01
H_D_G	0.10	0.03	0.10	0.03	0.60	0.01	0.10	0.03
H_A_Y	0.20	0.03	0.10	0.03	0.20	0.03	0.01	0.40
H_D_Y	0.05	0.03	0.05	0.03	0.05	0.03	0.75	0.01

B	WP_0	WP_B	WP_BG	WP_BGR	WP_BGRY	WP_BR	WP_BRG	WP_BRGY
H_A_B	0.01	0.93	0.01	0.01	0.01	0.01	0.01	0.01
H_D_B	0.93	0.01	0.01	0.01	0.01	0.01	0.01	0.01
H_A_R	0.01	0.01	0.01	0.47	0.01	0.47	0.01	0.01
H_D_R	0.05	0.05	0.85	0.01	0.01	0.01	0.01	0.01
H_A_G	0.01	0.01	0.47	0.01	0.01	0.01	0.47	0.01
H_D_G	0.05	0.65	0.01	0.01	0.01	0.25	0.01	0.01
H_A_Y	0.01	0.01	0.01	0.01	0.47	0.01	0.01	0.47
H_D_Y	0.05	0.10	0.10	0.40	0.01	0.18	0.15	0.01

Figure 43: HMM model λ_B for problem-solving scenario

Optimisation of HMM parameters: The HMM model parameters (probabilities of π, A and B) are assigned in consultation with the task expert. However, these may not be optimum values for the particular task scenario. Therefore, optimisation is required so that the HMM model can represent the task scenario with reasonable trueness. In other words, $P\ (\lambda_A \mid O_A)$ and $P\ (\lambda_B \mid O_B)$ must be maximised.

The optimisation of HMM parameters also known as 'the learning problem' is explained in section 5.3.3. In this example, the two HMM models λ_A and λ_B are put through the Baum Welch algorithm to optimise their parameters and following are the results.

For λ_A, the probability of observing O_A is $P\ (\lambda_A \mid O_A) = 3.85e - 3$

After the first Baum-Welch Iteration: $P_1(\lambda_A \mid O_A) = 1.72e - 4$

Since $P_1 < P$, the parameters of λ_A cannot be optimised beyond their initial values. This means that the initial estimation of parameters by the task expert were reasonably true for the model to represent the normal task scenario.

For λ_B, the probability of observing O_B is $P\ (\lambda_B \mid O_B) = 3.23e - 11$

After the first Baum-Welch Iteration: $P_1(\lambda_B \mid O_B) = 2.18e - 10\ returning\ P1 > P$

After the second Baum-Welch Iteration: $P_2(\lambda_B \mid O_B) = 9.96e - 14$

Since $P_2 < P_1$, convergence was reached after the first iteration itself and the parameters of λ_B cannot be optimised any further. λ_B with its new optimised parameters will be used in the subsequent steps of the framework for task scenario B (Figure 44).

P	H_A_B	H_D_B	H_A_R	H_D_R	H_A_G	H_D_G	H_A_Y	H_D_Y
	9.08E-01	4.20E-04	7.09E-03	1.05E-03	1.51E-03	7.41E-02	2.01E-03	5.73E-03

A	H_A_B	H_D_B	H_A_R	H_D_R	H_A_G	H_D_G	H_A_Y	H_D_Y
H_A_B	3.00E-03	2.59E-02	4.42E-01	8.61E-02	3.27E-01	5.43E-03	1.05E-01	5.68E-03
H_D_B	2.13E-01	8.32E-04	2.76E-01	8.46E-03	6.71E-02	4.09E-03	2.60E-02	4.05E-01
H_A_R	1.51E-02	7.36E-04	2.10E-03	1.37E-01	6.62E-01	4.40E-04	1.70E-01	1.28E-02
H_D_R	2.45E-03	1.41E-03	1.50E-02	3.56E-03	9.69E-03	7.81E-01	4.24E-03	1.83E-01
H_A_G	2.30E-01	6.53E-04	1.81E-01	1.70E-03	2.97E-03	3.55E-03	4.90E-01	9.06E-02
H_D_G	9.00E-03	1.26E-03	7.88E-01	1.74E-02	9.82E-02	3.95E-02	3.38E-02	1.23E-02
H_A_Y	6.81E-02	8.72E-04	5.12E-01	1.60E-03	9.84E-03	4.69E-03	4.70E-03	3.98E-01
H_D_Y	7.39E-03	9.69E-03	1.27E-02	1.50E-01	1.32E-01	3.49E-03	6.77E-01	8.24E-03

B	WP_0	WP_B	WP_BG	WP_BGR	WP_BGRY	WP_BR	WP_BRG	WP_BRGY
H_A_B	0.00E+00	8.59E-01	3.75E-03	8.86E-03	1.25E-01	9.49E-04	4.22E-04	1.81E-03
H_D_B	8.78E-01	1.50E-02	2.38E-03	4.61E-02	2.50E-02	2.47E-02	1.58E-03	5.88E-03
H_A_R	0.00E+00	3.48E-03	2.74E-02	5.17E-01	1.68E-04	4.50E-01	7.40E-05	1.37E-03
H_D_R	2.47E-02	1.03E-02	9.52E-01	3.24E-03	1.20E-02	4.98E-04	1.19E-03	5.29E-04
H_A_G	0.00E+00	1.04E-03	5.03E-01	2.08E-02	1.50E-02	5.61E-04	4.60E-01	1.75E-04
H_D_G	0.00E+00	8.93E-01	2.85E-02	1.02E-02	2.64E-03	6.49E-02	1.72E-04	9.02E-04
H_A_Y	0.00E+00	5.40E-03	1.08E-01	8.68E-03	4.20E-01	2.84E-03	1.25E-04	4.55E-01
H_D_Y	0.00E+00	1.18E-02	1.48E-02	9.54E-01	2.32E-03	6.27E-03	1.07E-02	4.29E-04

Figure 44: HMM model λ_B with optimised parameters

Convergence to local optimum: Consider another workpiece state sequence O_C, which is another way of solving the problem in task scenario B.

$O_C = \{WP\_B,\ WP\_BG,\ WP\_B,\ WP\_BG,\ WP\_BGR,\ WP\_BGRY,\ WP\_BGR,$

$WP\_BG, WP\_B,\ WP\_BR,\ WP\_BRG,\ WP\_BRGY\}$

This workpiece observation sequence differs slightly from O_B and the difference is highlighted in red font in O_C. The HMM model λ_B is again put through the Baum-Welch algorithm to see if it converges to λ_C again with the new observation sequence O_C.

For λ_B, the probability of observing O_C is $P\ (\lambda_B \mid O_B) = 2.93e - 12$

After the first Baum-Welch Iteration: $P_1(\lambda_B \mid O_C) = 1.71e - 10\ returning\ P1 > P$

After the second Baum-Welch Iteration: $P_2(\lambda_B \mid O_C) = 3.05e - 13$

Since $P_2 < P_1$, convergence was reached after the first iteration itself and the parameters of λ_B cannot be optimised any further. Hence λ_B with new optimised parameters is henceforth called λ_C (Figure 45). This proves that the Baum-Welch algorithm converges to a different optimum every time a new observation sequence is used for optimisation.

P	H_A_B	H_D_B	H_A_R	H_D_R	H_A_G	H_D_G	H_A_Y	H_D_Y
	0.9088358	6.32E-04	0.01614751	0.00118913	7.75E-04	0.06364647	0.0026376	0.00613693

A	H_A_B	H_D_B	H_A_R	H_D_R	H_A_G	H_D_G	H_A_Y	H_D_Y
H_A_B	7.57E-04	2.46E-01	3.76E-01	2.20E-01	1.28E-01	4.06E-03	2.28E-02	3.15E-03
H_D_B	9.67E-01	1.61E-04	5.59E-03	3.15E-03	2.94E-03	1.15E-02	1.91E-03	7.36E-03
H_A_R	2.67E-02	9.38E-04	8.20E-04	1.27E-01	6.43E-01	2.69E-03	1.93E-01	5.89E-03
H_D_R	3.54E-02	1.37E-03	2.95E-02	1.75E-03	4.63E-03	8.46E-01	2.02E-03	7.90E-02
H_A_G	2.73E-01	7.89E-04	1.95E-01	1.31E-03	2.81E-03	2.11E-02	4.09E-01	9.73E-02
H_D_G	1.09E-02	2.71E-01	1.33E-02	1.01E-01	5.58E-01	1.84E-03	3.67E-02	6.71E-03
H_A_Y	8.77E-02	6.85E-03	5.16E-01	8.35E-03	2.15E-02	6.54E-03	6.24E-03	3.47E-01
H_D_Y	1.25E-02	1.72E-02	1.29E-02	1.51E-01	1.43E-01	6.82E-03	6.46E-01	1.03E-02

B	WP_0	WP_B	WP_BG	WP_BGR	WP_BGRY	WP_BR	WP_BRG	WP_BRGY
H_A_B	9.22E-05	9.73E-01	4.11E-03	5.01E-03	1.64E-02	4.13E-05	2.20E-04	9.94E-04
H_D_B	9.30E-01	1.25E-03	6.30E-02	2.57E-03	1.43E-03	1.42E-03	6.19E-05	3.05E-04
H_A_R	2.55E-03	1.34E-02	2.04E-02	5.26E-01	1.69E-04	4.36E-01	3.58E-05	1.28E-03
H_D_R	3.16E-03	6.51E-03	9.81E-01	1.12E-03	7.04E-03	4.34E-05	5.92E-04	2.53E-04
H_A_G	4.68E-03	1.51E-03	5.90E-01	8.81E-03	1.48E-02	4.94E-05	3.80E-01	1.34E-04
H_D_G	1.38E-03	9.69E-01	3.51E-03	1.23E-02	1.17E-03	1.17E-02	5.66E-05	3.92E-04
H_A_Y	2.65E-03	1.87E-02	7.84E-02	1.11E-02	4.22E-01	1.14E-03	8.04E-05	4.66E-01
H_D_Y	2.81E-03	2.79E-02	2.67E-02	9.29E-01	3.08E-03	1.06E-03	8.72E-03	4.24E-04

Figure 45: HMM model λ_C with optimised parameters

Thus the output of the modelling step is a set of unique HMM models with optimised parameters representing their corresponding workpiece observation sequences. Each model therefore embodies the human-workpiece interactions that are involved in their respective task scenarios. These models are now ready to be analysed in the knowledge extraction and decoding steps.

HMM evaluation – picking the right model

The HMM model that represents a particular task scenario can be queried/analysed to extract the constituents of manufacturing knowledge embedded within that scenario. Each task can have multiple models as in this case $(\lambda_A, \lambda_B\ and\ \lambda_C)$ therefore, for a given task scenario it is necessary to pick the right model for analysis. This is the case for 'the evaluation problem' explained in section 5.3.3 in which the 'Forward' algorithm is used to find the model that returns the maximum probability to represent a given task scenario.

Consider a task scenario represented by the workpiece observation sequence

$$O_{Q1} = \{WP\_B,\ WP\_BR,\ WP\_BRG,\ WP\_BRGY\}$$

By deterministic evaluation the sequence in O_{Q1} is compared with that in O_A, O_B and O_C and a match is found with O_A. Therefore, λ_A is picked for the given observation sequence O_{Q1}.

Now, consider a task scenario represented by the workpiece observation sequence

$$O_{Q2} = \{WP\_B,\ WP\_BG,\ WP\_B,\ WP\_BG,\ WP\_BGR,\ WP\_BGRY,\ WP\_BGR,$$

$$WP\_BG,\ WP\_B,\ WP\_BR,\ WP\_BRG,\ WP\_BRGY\}$$

In this case, deterministic evaluation fails to pick a model because the given observation sequence O_{Q2} does not match any of the observation sequences O_A, O_B and O_C. Therefore, by stochastic evaluation $P(\lambda_A \mid O_{Q2})$ is compared with $P(\lambda_B \mid O_{Q2})$ and $P(\lambda_C \mid O_{Q2})$ and the model with the highest probability is picked. Therefore, by using the 'Forward' algorithm:

$$P\ (\lambda_A \mid O_{Q2}) = 5.47e - 14$$

$$P\ (\lambda_B \mid O_{Q2}) = 3.23e - 13$$

$$P\ (\lambda_C \mid O_{Q2}) = 1.70e - 10$$

Since $P\ (\lambda_C \mid O_{Q2}) > P\ (\lambda_B \mid O_{Q2}) > P\ (\lambda_A \mid O_{Q2})$, HMM model λ_C is picked for the given observation sequence O_{Q2}.

<u>Step 4: Extract</u>

Once the best HMM model is identified for a given task scenario, the sequence of human action states responsible for that scenario can be extracted using the Viterbi algorithm as described in section 5.3.3. This algorithm uses both forward and backward recursive methods to arrive at the most likely sequence of human action states. Once the human action state sequence is extracted, detailed analysis can be performed on the individual states in the sequence to decode the manufacturing knowledge embedded within them.

In the previous step, for observation sequence O_{Q1}, HMM model λ_A is identified that most likely embodies the task scenario represented by O_{Q1}.Using the 'Viterbi' algorithm, the most likely sequence of human actions H_{Q1} that could produce O_{Q1} is identified.

Therefore, $H_{Q1} = \{H\_A\_B,\ H\_A\_R,\ H\_A\_G,\ H\_A\_Y\}$

Note that H_{Q1} is the exactly the same as O_A , the original observation sequence that λ_A was based on. This shows that the parameters for λ_A were appropriately assigned.

However, this phenomenon may not be seen for other observation sequences due to the stochastic nature of the 'Viterbi' algorithm. Such a case is presented when the workpiece observation sequence O_{Q2} is considered. From the previous step, HMM model λ_C is identified that most likely embodies the task scenario represented by O_{Q2}. Again using the 'Viterbi' algorithm, the most likely sequence of human actions H_{Q2} that could produce O_{Q2} is identified.

Therefore, $H_{Q2} = \{H\_A\_B,\ H\_A\_G,\ H\_D\_G,\ H\_D\_R,\ H\_D\_Y,\ H\_A\_Y,$
$H\_D\_Y,\ H\_D\_R,\ H\_A\_B,\ H\_D\_B,\ H\_A\_B,\ H\_A\_R,\ H\_A\_G,\ H\_A\_Y\}$

Note that H_{Q2} is not captured during any of the task scenarios but is determined stochastically. This case demonstrates that even for task scenarios that have not been observed and captured, the 'Extract' step is able to output the most likely human actions that could have produced that scenario. Therefore, various task scenarios can be simulated and the human response to those scenarios can be predicted to gain a deeper insight into complex manual tasks.

Step 5: Decode

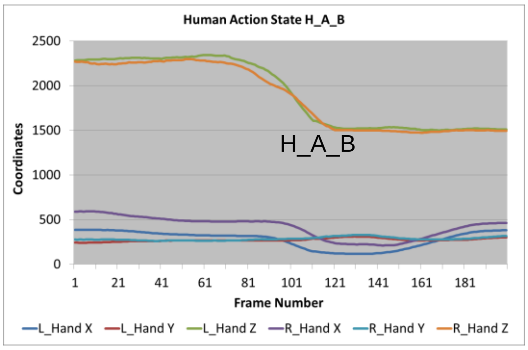
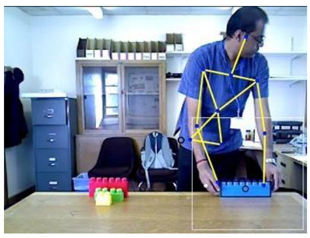
Given a task scenario, steps 4 and 5 have shown that the right HMM model that best represents that scenario can be picked and the human action sequence responsible for that scenario can be extracted. This sequence is made up of human action states that correspond directly to the changes the workpiece has undergone in the given task scenario. In this step, these human-workpiece interactions are further analysed to decode the manufacturing knowledge embedded within them.

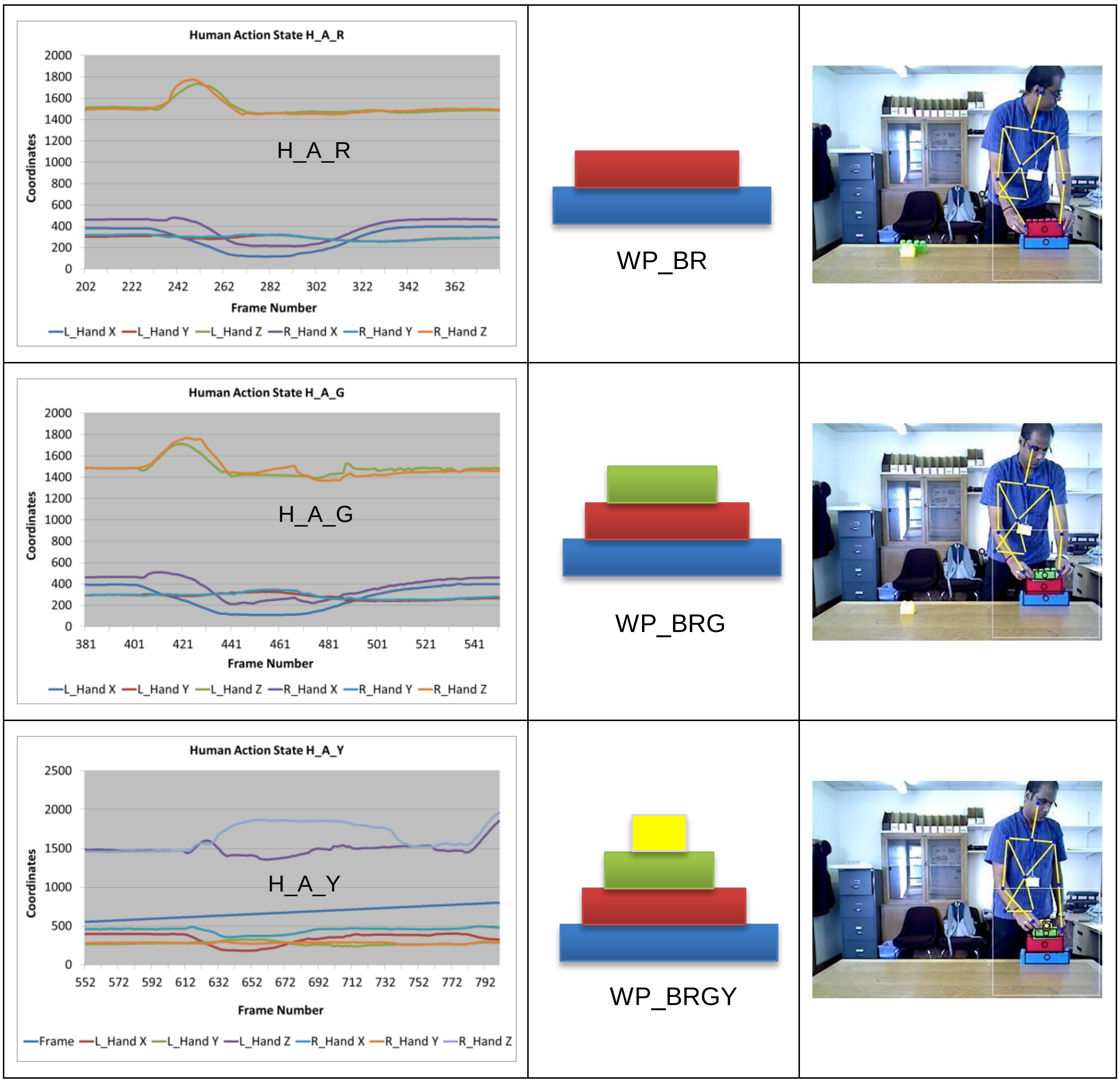
There are several constituents of manufacturing knowledge such as the task strategy adopted, nature and spatial characteristics of gestures made, mechanics of motion performed during the gestures, action choices made in response to task situations and workpiece manipulation techniques. These constituents can be decoded from the raw human action data and workpiece tracking data stored within the states that are extracted.

Extraction of states from the human action sequence

Consider the output sequence $H_{Q1} = \{H\_A\_B,\ H\_A\_R,\ H\_A\_G,\ H\_A\_Y\}$ extracted from HMM model λ_A from the workpiece observation sequence $O_{Q1} = \{WP\_B,\ WP\_BR,\ WP\_BRG,\ WP\_BRGY\}$. The human action data within the states of H_{Q1} is pulled out from the raw human motion data and mapped to the corresponding workpiece states from the observation sequence (Table 5).

Table 5: Human action data mapped to corresponding workpiece states

Human Action State	Workpiece State	Observed Task Status
	 WP_B	



Decoding of manufacturing knowledge

Multiple constituents of manufacturing knowledge can be decoded from the extracted states and human action data. These constituents include the following but are not limited to:

- Task execution strategy
- Nature and spatial characteristics of human gestures while working on the workpiece
- Workpiece manipulation techniques
- Mechanics of human movements such as body bending angle, angles between upper and lower arms, body orientation with respect to the workpiece, etc.

Task execution strategy

The task execution strategy can be decoded by determining the sequence of human actions responsible for the given sequence of workpiece states from the most likely HMM model. From the human action sequence, the following knowledge can be decoded:

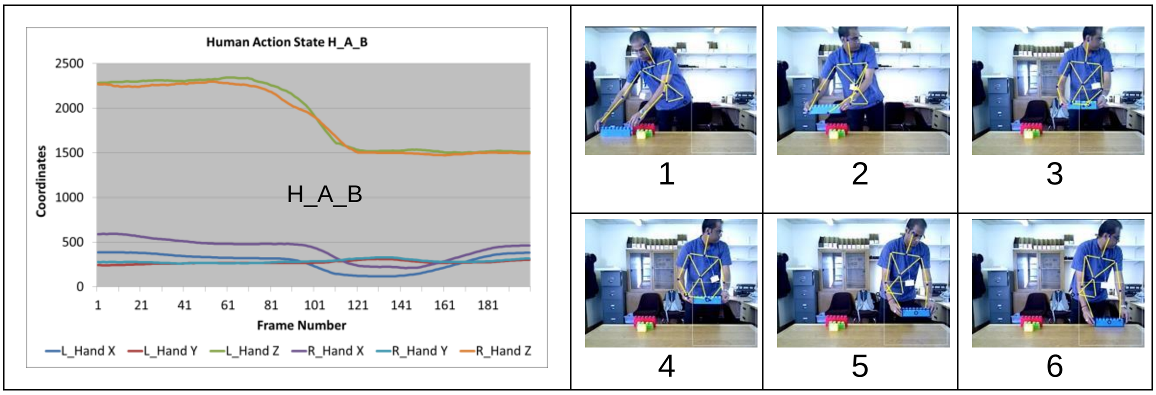
- Plan and approach of task execution by breaking the task down into sub-tasks i.e. the action states.
- Sequence of execution of the sub-tasks to achieve the main task. This sequence depends on the task scenario being queried.
- Selections made during the task to choose specific actions from a repertoire of actions available to the human to successfully complete the task. In this example, the repertoire of human actions H is $\{H\_A\_B,\ H\_A\_G,\ H\_D\_G,\ H\_D\_R,\ H\_D\_Y,\ H\_A\_Y,\ H\_D\_Y,\ H\_D\_R,\ H\_A\_B,\ H\_D\_B,\ H\_A\_B,\ H\_A\_R,\ H\_A\_G,\ H\_A\_Y\}$

  From this set, the actions chosen are $H\_A\_B,\ H\_A\_R,\ H\_A\_G\ and\ H\_A\_Y$ and each one is executed at specific times during the task.

Nature and spatial characteristics of human gestures

A human action state in the digitisation framework is not a static snapshot of the human pose at time ‘t’ but a continuous series of human gestures from time t1 to t2, which manipulate the workpiece from its state at time t1 to its resulting state at time t2. Therefore, by visualising the human motion data within each extracted action state, the nature (trajectories and patterns) and spatial characteristics (3D spatial coordinates) of human gestures with respect to the changes in the workpiece can be obtained. The human motion data and corresponding workpiece change data for the extracted state H_A_B is shown as an example (Table 6).

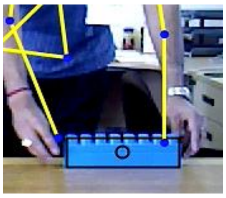
Table 6: Human actions and workpiece progress during state H_A_B

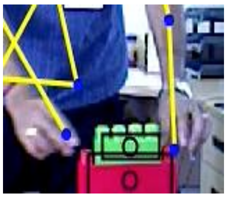


<u>Workpiece manipulation techniques</u>

Workpiece manipulation, such as grasping and release, techniques vary from task to task and person to person. However, for each task there is a 'correct' technique to grasp and release workpieces and these techniques can be learnt from observing how the experts do it. In this example, the human skeletal motion data and the workpiece configuration and position data from the extracted states can be visualised to extract workpiece grasp and release techniques. For the extracted action states $H\_A\_B,\ H\_A\_R,\ H\_A\_G\ and\ H\_A\_Y$, Table 7 shows how the expert manipulated each workpiece while performing his action during those states.

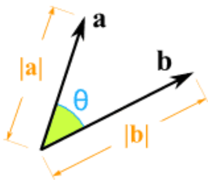
Table 7: Workpiece manipulation techniques illustrated for each extracted state

Human Action State	Workpiece Grasping Technique	Workpiece Release Technique
H_A_B		
H_A_R		

H_A_G		
H_A_Y		

Mechanics of human motion

Human motion data is complex and consists of the 3D spatial coordinates of 20 different skeletal joints that are tracked and recorded in the capture step. Useful insights can be drawn from this complex motion data.

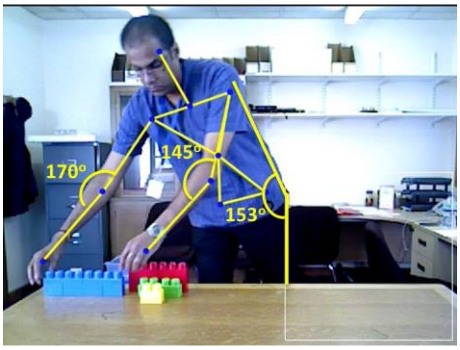
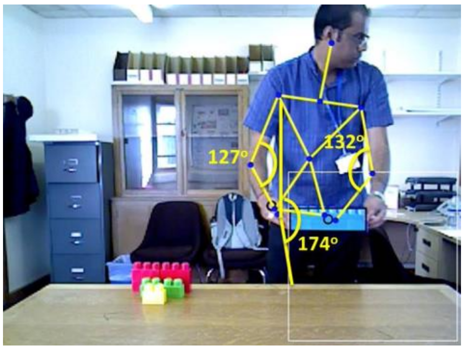
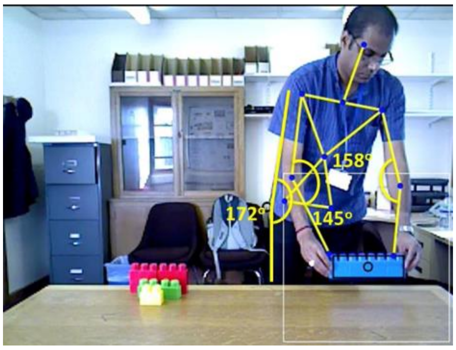
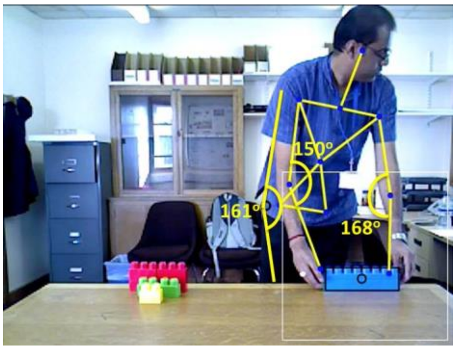


Any motion parameter that is important to understand the human's action skills can be computed from the joint coordinate data. For example, the angle between the upper and lower arms represented by vectors 'a' and 'b' can be obtained from the equation: $a \cdot b = |a| \times |b| \times \cos(\theta)$, where $|a|$ is the magnitude of vector 'a' and θ is the angle between them.

Motion mechanics such as body bending angle, angles between the upper and lower arms and motion speed and acceleration for each critical joint can be mathematically calculated using vector computing. This information is vital to understand the physical nuances of skill execution. A comparison of the mechanics computed for different people executing the same task can provide a skill metric to quantify skill levels of people and as a measure to gauge the efficacy of skill training processes. However, the comparison of motor skills is valid only between people with similar builds and body proportions. This is because a human with relatively shorter height might need to perform certain workpiece manipulations in a completely different manner as compared to a taller human in order to achieve the same task result. Motion data can also provide the kinematics information to design and develop an automation solution that could mimic human motion for a complex task.

Table 8 illustrates the computation of body bending angle and the angles between upper and lower arms for 4 random human poses chosen from the action states.

Table 8: Human motion mechanics extracted from motion data

	
• Angle between upper & lower arm (Left): 145^{o} • Angle between upper & lower arm (Right): 170^{o} • Body bending angle: 153^{o}	• Angle between upper & lower arm (Left): 132^{o} • Angle between upper & lower arm (Right): 127^{o} • Body bending angle: 174^{o}
	
• Angle between upper & lower arm (Left): 158^{o} • Angle between upper & lower arm (Right): 145^{o} • Body bending angle: 172^{o}	• Angle between upper & lower arm (Left): 168^{o} • Angle between upper & lower arm (Right): 150^{o} • Body bending angle: 161^{o}

Step 5: Reproduce

There are multiple ways as suggested in section 5.2 to reproduce the decoded manufacturing knowledge. One of the simplest ways to reproduce the task strategy is to tabulate the human action states and their corresponding workpiece observation states in chronological order. Using graphics-rich media such as immersive virtual environments, the task execution can be demonstrated using human avatars on virtual workpieces. Such a demonstration can be augmented with the knowledge constituents such as motion mechanics for better visualisation of the skills involved. Augmentation of this information can also be done on a real task environment using mixed reality technologies for in-view, hands-free access.

It is not within the scope of this research to reproduce manufacturing knowledge using virtual or mixed reality. 2D animation is used instead as a medium to reproduce the human-workpiece interactions with augmented information about the manufacturing knowledge constituents. An example of a basic animation of executing the Lego block assembly task is shown in Figure 46. A trained human should be able to learn from the animation, augmented with additional task knowledge such as motion mechanics and expected workpiece progress during the animation, in order to acquire the necessary skills to perform the assembly task successfully.

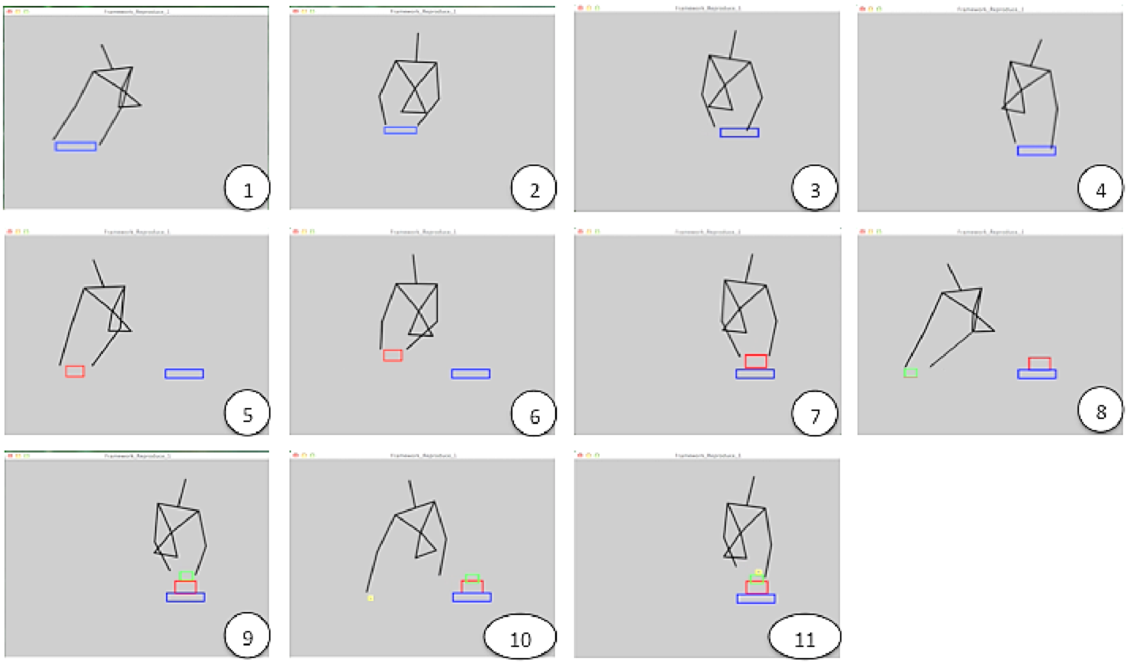


Figure 46: Human-workpiece interactions animated for a specific task scenario

6.5 Digitisation of manufacturing task environment

6.5.1 Overview

In the implementation of the framework described so far, the environment within which the assembly tasks were performed did not play a role in the digitisation process. The environment was controlled to remain unchanged during the course of task execution and therefore the 'Capture' step of the framework was not programmed to track any changes that may have taken place within it. However, it is necessary to investigate the digitisation of the manufacturing environment as well to capture any changes that might affect the task positively or negatively.

In this study, an effort is made to digitise the manufacturing environment of a manual assembly task from the automotive industry, namely the manual wheel loading operation that occurs in the trim and final assembly line in automobile production. Human actions within the task are not tracked to maintain a sharp focus on the digitisation of the task environment only. Therefore, a scenario is envisaged in which the wheel loading operation is automated by replacing the human operator with an automated wheel loading system and the purpose of implementing the 'Capture' step is to feed digitised information about the task environment to this automated system.

This study differs from the Lego block assembly study presented in section 6.1 in 4 main areas:

1. Only the 'Capture' step is implemented and not the whole framework.
2. Three Kinect sensors are used instead of just the one to digitise different parts of the task environment.
3. Real-life engineering workpieces, tools and activities are used to mimic the actual task.
4. The other ambient characteristics of the task environment, such as lighting are not controlled.

6.5.2 Choice of task

The manual wheel loading operation is chosen because:

1. It is a prime target for automation in the automotive industry and automation of complex manual tasks is one of the potential applications of the proposed framework.

2. It involves real-world engineering workpieces such as the wheel and the wheel hub and therefore the capability of the ‘Capture’ step to recognise and track them within an industrial-scale task environment can be tested.
3. It can be easily mimicked in the laboratory using industrial-scale tools.

6.5.3 Background

The automotive industry is one of the early adopters of automation for material handling, processing, assembly and inspection operations and continues to be highly automated (Gupta and Arora, 2009). However, there are a few operations in vehicle production that have not been automated yet such as those in the trim and final assembly line where the vehicle gets its seats, internal and external trims, and wheels. This is because the installation of components on a constant moving vehicle body is a complex task that is as yet best performed by skilled human operators (Choi et al., 2010).

While this study tests the ‘Capture’ step of the proposed framework to digitise a task environment, it does so in the context of enabling the automation of the wheel loading operation in the trim and final assembly line. Though the operation would seem straightforward for a human operator, it is one of the most complex manufacturing assembly activities to automate because it requires the human ability to accurately track the moving vehicle body that sways unpredictably on the conveyor line and to recognise in real-time the alignment features for successful assembly. Human operators perform wheel loading accurately and effectively using their multi-modal sensing abilities and acquired skills. These characteristics allow them to intelligently manoeuver the wheel towards the wheel hub and install it while constantly compensating for arbitrary motion deviations of the car body. They have to instinctively rotate the wheel if required to align the tapped bores on the wheel to the threaded studs on the wheel hub and swiftly react to take adaptive steps in case of unforeseen situations like unplanned conveyor halts (Figure 48). Therefore, it has been difficult to replace skilled human wheel loading operators with automated solutions. Chen et al. (2009) have indicated that the wheel loading operation alone can cost automotive manufacturers up to US$1.5 million a year thereby justifying the need to automate this operation. The potential threat of developing musculoskeletal disorders in operators, caused by manoeuvring heavy wheels in uncomfortable body postures during installation despite using weight compensation gantries (Figure 47), further reinforce the need for automation. A focused literature review is presented in Appendix A.

Figure 47: Manual wheel loading operation

The 'Capture' step provides a cost-effective method to track the motion characteristics of the moving vehicle body in real-time and simultaneously identify the misalignment between the to-be-loaded wheel and the wheel hub that receives the wheel. This data can be used by the automation solution to gather intelligence about the operation enabling it to successfully perform the wheel loading operation. An example of an automation solution could be an expert system that controls an industrial robot arm to align and load the wheels on to the moving vehicle body.

Wheel hub mounted on the vehicle axle | **Wheel** | **Car body on the conveyor line with wheel hubs installed (Turpen, 2012)**

Figure 48: A typical wheel hub, wheel and assembly line

6.5.4 Method

Mimicking the wheel loading operation

In order to collect live data pertaining to a wheel loading operation and given that it was not possible to do this in an actual production line, the key elements of the operation are mimicked in laboratory conditions.

An important element of the operation is the moving conveyor line carrying the vehicle body with the wheel hubs mounted on the front and back axles. Therefore, the motion of the wheel hub on the conveyor line is reproduced by mounting the wheel hub on to a robot arm and programming the robot arm to mimic typical conveyor line motion. The conveyor motion characteristics such as out-of-plane deviations are programmed as sinusoidal oscillations in the following five patterns:

1. Linear motion of the wheel hub along x-axis without any deviations at an average speed of around 67mm/s (Shi, 2008).

2. Linear motion of step 1 with stop-start movements to mimic the vehicle body jerks on the conveyor line.

3. Linear motion of step 1 with sinusoidal motion deviations along the vertical y-axis to mimic the bounce of the vehicle body on the conveyor line.

4. Linear motion of step 1 with sinusoidal motion deviations along the perpendicular z-axis to mimic the sway of the vehicle body on the conveyor line.

5. Linear motion of step 1 with sinusoidal motion deviations along both y-axis and z-axis to mimic the composite effect of both bounce and sway of the vehicle body on the conveyor line.

According to the data received from a Tier 1 manufacturer, a typical vehicle body in motion on a conveyor will deviate from linear motion with out-of-place oscillations of +/-10mm in amplitude and a frequency of 1Hz (Chen et al., 2010).

The second important element of the operation is the radial alignment between the wheel and the wheel hub so that the bores of the wheel are in the same angular position as the studs on the wheel hub at the time of loading. The misalignment scenarios are also reproduced during the experiments by positioning the wheel hub on the robot arm with varying angular positions.

Experiment setup

The wheel loading workstation, simulated in the laboratory, is divided into two motion sensing zones, namely, the far sensing zone and the near sensing zone. This is done to cover the entire 2.5m length of a typical wheel loading workstation. In the far sensing zone, the coarse motion of the moving wheel hub is tracked whereas in the near sensing zone, the motion characteristics are closely monitored. In the near sensing zone, the alignment features on the moving wheel hub are also recognised and their angular positions are measured.

Two Kinect sensors, one in the far sensing zone (called the ‘far sensor’) and one in the near sensing zone (called the ‘near sensor’), are used (Figure 49). The wheel hub is mounted on the robot arm with the studs facing the sensors. Comau NM-45, a 6-axis industrial robot arm with a maximum payload of 45Kg and position reproducibility of 0.06mm is used. The robot is programmed to mimic the 5 conveyor motion patterns listed above and each pattern is repeated 10 times to obtain multiple datasets to gauge reproducibility of results.

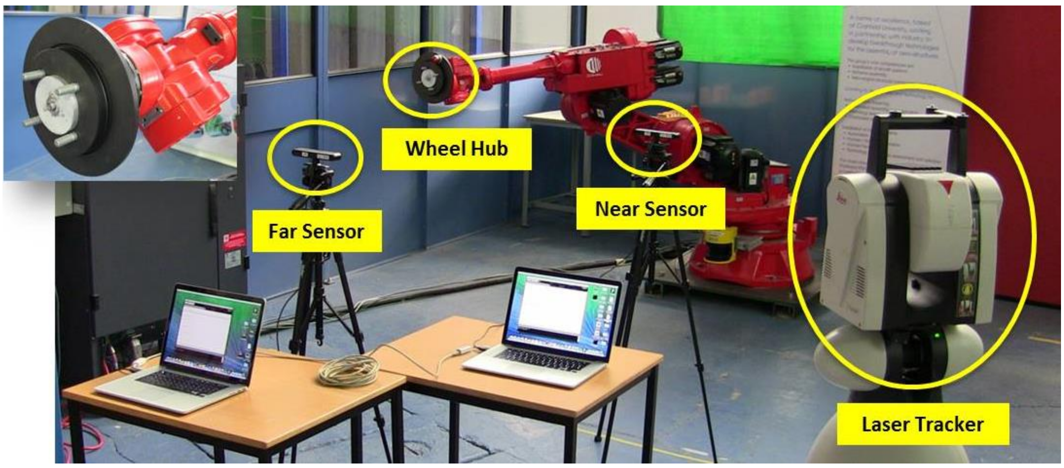


Figure 49: Wheel loading workstation setup simulated in the laboratory

Kinect sensor positioning

The Kinect sensors are reported to have maximum accuracy over the distance range of 1 to 3m from the sensor with an effective field of view of 54.0°

horizontal and 39.1° vertical (Dutta 2012). Khoshelham and Elberink (2012) have reported that the random error of measurement results increases quadratically with increasing distance from the sensor and reaches 40 mm at the maximum range of 5m. These inputs influenced the positioning of the Kinect sensors in the far and near sensing zones to cover the entire length of the wheel loading workstation with the combined frames of view of the two sensors.

Therefore, the far sensor was placed at a perpendicular distance of 2m from the moving wheel hub plane covering a horizontal field of view of about 2m. The near sensor was placed at a distance of 850mm covering a horizontal field of view of about 800mm. The two sensors are laterally separated by a distance of 1m to attain a 400mm of view overlap with each other and are placed at the same height as that of the moving wheel hub from the ground. The two sensors together cover an area of 2.4m of the workstation (Figure 50).

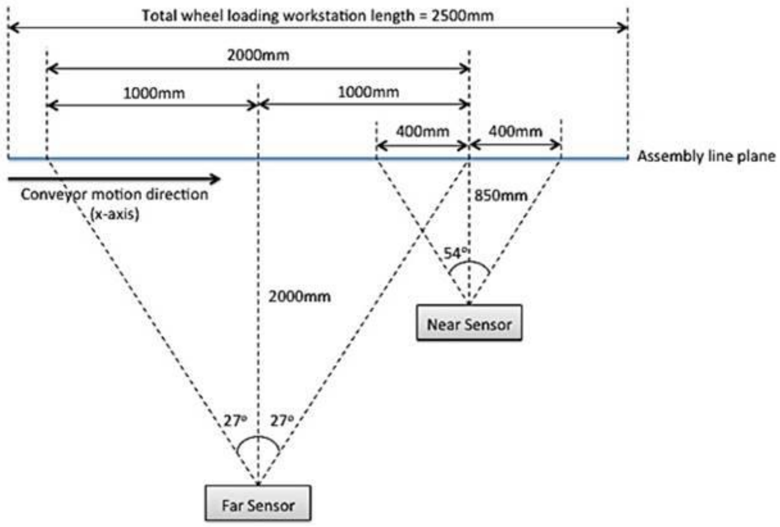


Figure 50: Kinect sensor positioning for far and near motion sensing

In addition to the Kinect sensors, a laser motion tracker (Leica Absolute Tracker AT402) is also used to track the motion of the moving wheel hub (Figure 49). The laser tracker uses a laser beam that is reflected off a reflector that is attached to the wheel hub to track its motion. It has a resolution of 0.1 μm, accuracy of +/- 10 μm and repeatability of +/- 5 μm making it a very accurate device for tracking motion and therefore is used to gauge the accuracy and precision of the proposed depth sensor based method.

The third Kinect sensor is placed directly in front of the wheel placed on the storage rack and at the same height as that of the centre of the wheel from the ground (Figure 51). This sensor recognises the alignment features on the wheel, which are the four tapped bores, and measures their angular positions.

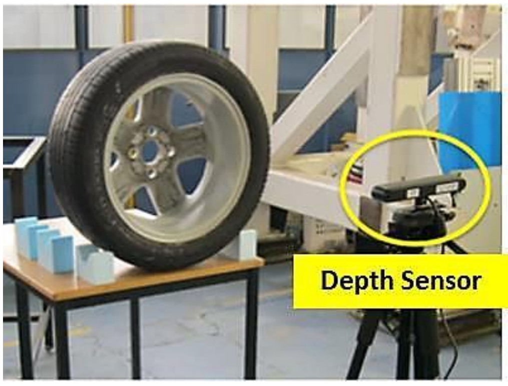


Figure 51: Alignment feature recognition of the stationary wheel

Sequence of events

This experiment imitates the wheel loading operation as it is performed in an actual automotive trim and final assembly line. The sequence of events reproduced is as follows:

a) The robot arm moves the wheel hub linearly across the workstation (along x-axis) for a distance of about 2.5m at an average speed of about 67mm/s. Typical conveyor motion deviations are programmed into the path as per the 5 patterns listed in section 3.1.

b) The wheel hub first enters the far sensing zone in which the far sensor tracks it and records its spatial position in all 3 axes. The speed of motion is also computed.

c) The wheel hub then enters the near sensing zone in which the near sensor tracks it and records its spatial position in all 3 axes. The speed of motion is also computed. The near sensor also recognises the alignment features, the 4 studs on the moving wheel hub, to record their angular positions.

d) The Kinect sensor placed in front of the stationary wheel recognises the alignment features, the 4 bores on the wheel to record their angular positions.

e) The data generated in each of the above steps enables the automated wheel loading solution to make critical decisions such as when, where and how to load the wheel onto the moving wheel hub and to dynamically correct the misalignments if any between the wheel and the wheel hub before loading. Any data that is out of the tolerance limits can be used to trigger an abort.

Motion tracking and feature recognition of the moving wheel hub

The algorithms used to track the moving wheel hub and recognise the angular positions of the alignment features of both the moving wheel hub and the stationary wheel are based on the comparison of depth values of the pixels that belong to the object with those of the background. In this manner, the object edge is located from each depth image and its centre point is computed. Since the sensor produces up to 30 depth image frames per second, the continuous computation of the centre point within these images results in tracking the motion of that object.

The moving wheel hub first enters the field of view of the far sensor in the far motion sensing zone. This zone covers the pre-loading area where the x, y and z positions of the centre of the wheel hub are tracked and its motion speed is constantly computed (Figure 52). Any deviations or disruptions in motion along any of the 3 axes are captured and recorded. Since the far sensor is placed at a relatively larger distance from the wheel hub motion plane, it can track a wider area but being less accurate is used to measure coarse motion characteristics.

The wheel hub then moves into the field of view of the near sensor in the near motion sensing zone. This zone covers the loading area and therefore the fine motion is tracked with more accuracy and precision than in the far sensing zone. In this zone, the alignment features of the moving wheel hub are also recognized (Figure 53a). The 4 studs located 90^{o} apart from each other at a pitch centre diameter of 108mm from the centre (Figure 53b) are recognised and their angular positions are measured in terms of the angular position of the stud located within the 90^{o} to 180^{o} quadrant.

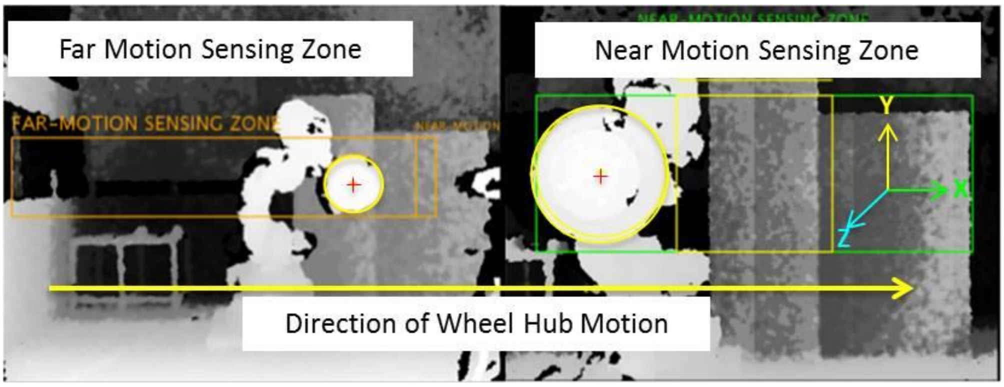
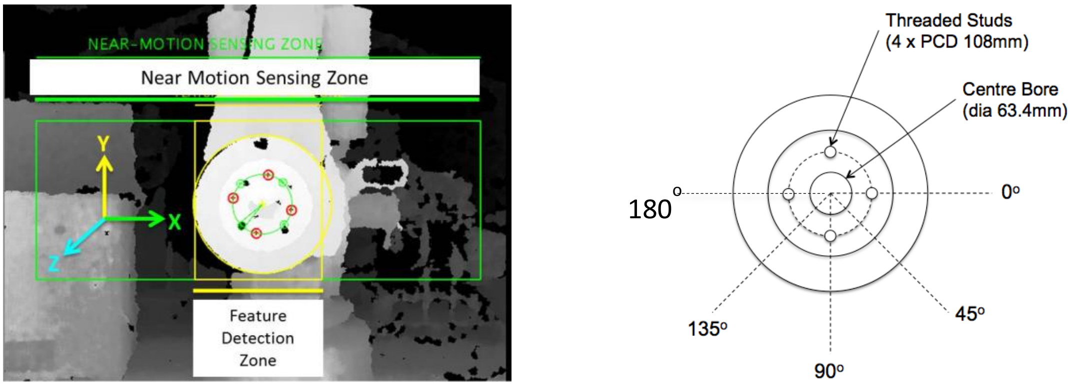


Figure 52: The moving wheel hub tracked in the two motion-sensing zones

Object and feature recognition of the stationary wheel

The depth sensor placed in front of the stationary wheel also uses the same edge detection algorithm to detect the wheel centre as the one used to detect the moving wheel hub centre (Figure 54a).

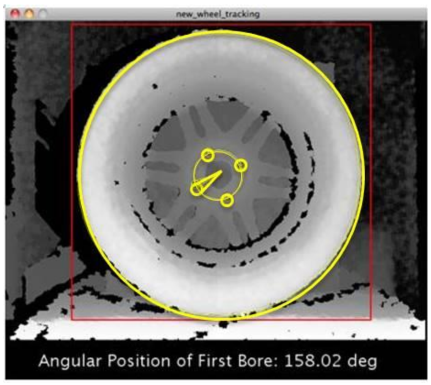


(a) (b)

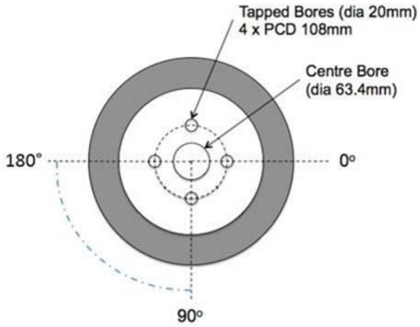
Figure 53: (a) Wheel hub tracking and feature recognition (b) Wheel hub drawing

The 4 bores located 90^o apart from each other at a pitch centre diameter of 108mm from the wheel centre (Figure 54b) are recognised and their angular positions are measured in terms of the angle of the bore located within the 90^o to 180^o quadrant.

The difference between the angular positions of the alignment features on the wheel and those on the wheel hub denote a misalignment (Figure 55) that needs to be corrected before loading can take place.



(a)



(b)

Figure 54: (a) Recognised wheel and wheel bores and (b) 2D wheel drawing

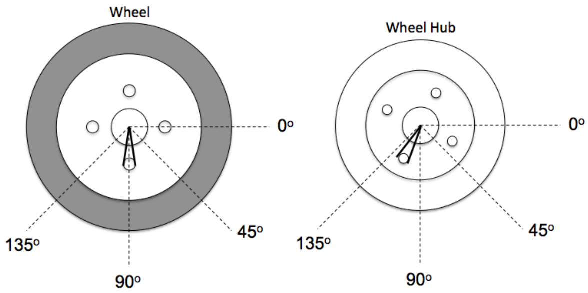


Figure 55: Misalignment of wheel and wheel hub features

6.5.5 Results

The results of the task environment digitisation are presented in this section in the following order:

1. Identification of wheel features and measurement of the angular positions of the wheel bores.
2. Motion tracking of the moving wheel hub and identification of the angular positions of the wheel studs for the programmed motion pattern no. 5, which is the linear motion along x-axis with sinusoidal deviation in y and z-axis, the most complex pattern. Tracking results for all motion patterns are presented in Appendix B.

Identification of wheel features and measurement of the angular positions of the wheel bores

The Kinect sensor captures depth images of the stationary wheel at the rate of up to 30 frames per second. From within each depth image, the 4 bores of the wheel are recognised and their angular positions, represented by the angle of the bore located within the 90^{o} to 180^{o} quadrant (the 'first bore'), are measured. To improve the accuracy of this method, the angle obtained is cumulatively averaged over 45 depth frames before it is recorded. 10 iterations of the experiment are conducted and the results are tabulated in Table 9.

Table 9: First wheel bore angle and its standard deviation (10 iterations)

Iteration	1	2	3	4	5	6	7	8	9	10	Mean	Standard Deviation
Angle	102.95	102.98	102.99	103.01	103.03	103.08	103.25	103.42	103.57	103.73	103.20	0.28

Motion tracking of the moving wheel hub and identification of the angular positions of the wheel studs

The far and near sensors track the motion of the wheel hub by continuously detecting the centre point of the hub and recording its x, y and z coordinates along with its speed in the direction of motion (x-axis). In the far sensing zone, the far sensor tracks the position and speed of the wheel hub whereas in the near sensing zone, the near sensor tracks its motion and identifies the angular positions of the studs of the moving wheel hub.

The motion tracking data obtained from the far and near sensors is compared to that obtained from the laser tracker that tracks the same motion. Since the laser tracker and the depth sensor are not synchronised during motion tracking, the two sets of data cannot be plotted and visualised on the same chart. The motion tracking results for the five simulated motion patterns are presented below. Since each motion pattern is run for 10 iterations, the wheel hub position and speed values are averaged over the 10 iterations.

Linear motion at 67mm/s along x-axis with deviations in y and z-axis

<u>In the far sensing zone</u>: Figure 56 and Figure 57 show the motion charts produced by the far sensor and the laser tracker for y-axis and z-axis deviations respectively. Since the oscillations are along the y-axis and z-axis, x-axis motion tracking chart is not shown.

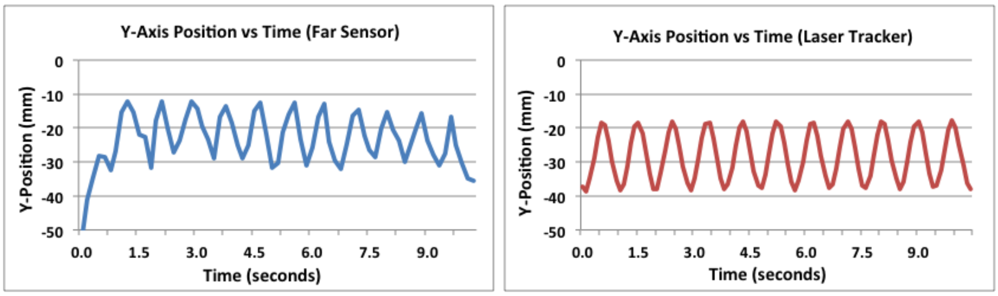


Figure 56: Wheel hub positions - far sensor and the laser tracker (y-axis)

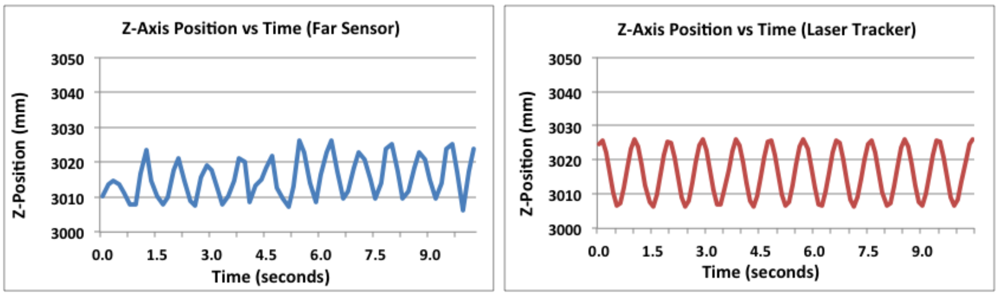


Figure 57: Wheel hub positions - far sensor and the laser tracker (z-axis)

<u>In the near sensing zone:</u> Figure 58 and Figure 59 show the motion charts produced by the near sensor and the laser tracker for y-axis and z-axis deviations respectively. Table 10 shows the angular positions of the wheel hub measured over 10 iterations for this motion pattern.

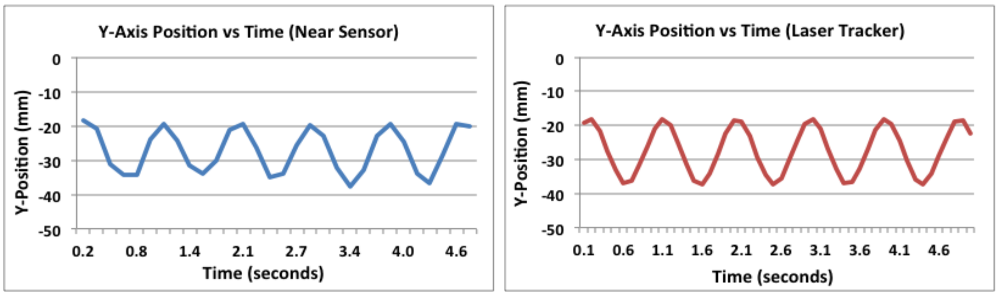


Figure 58: Wheel hub positions - near sensor and the laser tracker (y-axis)

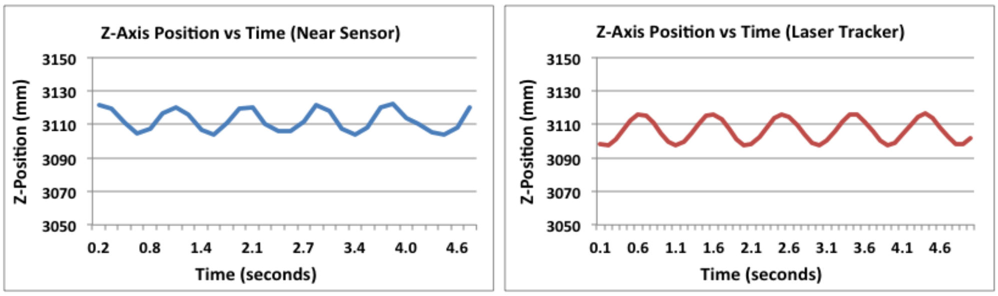


Figure 59: Wheel hub positions - near sensor and the laser tracker (z-axis)

Table 10: First wheel hub stud angle and its standard deviation (10 iterations)

Motion Pattern	Angular Position of First Wheel Hub Stud (degrees)											
	1	2	3	4	5	6	7	8	9	10	Mean	Std. Deviation
Linear motion in x-axis with deviations in y and z-axis	336.78	335.79	338.14	333.56	335.99	337.06	333.54	336.42	337.14	334.80	335.92	1.54

6.5.6 Performance and proposed improvements

Optimum sensor setup is vital for reliable performance of the wheel loading task digitisation process. This involves the distances at which the sensors are placed from the task plane, sensor face angles with respect to the task plane, the number of image frames averaged for error minimisation and the IR interference between the two Kinect sensors. All these aspects are studied and the analysis is presented in Appendix C.

The results show that the proposed Kinect sensor based task capture method is able to track the moving wheel hub and measure its motion characteristics in real-time. The use of a far and near motion sensing zone isolates the low and high accuracy needs of the wheel loading operation while being able to capture the entire workstation length. Despite the relatively low resolution of the Kinect sensor, placing it at a short perpendicular distance of 850mm from moving wheel hub plane, the resulting motion tracking error of 2.78mm is achieved.

For the use case in this study, the wheel bores are 20mm in diameter and the wheel hub studs are 12mm in diameter. Therefore, an assembly tolerance of 4mm is required for successful wheel loading irrespective of the motion patterns. The error in measuring motion deviation amplitude, especially in the crucial near motion sensing zone of 2.78mm is less than the required assembly tolerance. For the measurement of angular position of the wheel hub studs, the maximum standard deviation noted was 1.54^{o}, which is an equivalent of 1.46mm, is also less than the 4mm tolerance required. Therefore the proposed method is feasible to be implemented in wheel loading operations that use the specifications of the wheel and the wheel hub used in this study.

According to Chen et al. (2005), the minimum assembly tolerance used in the industry is 2mm. The maximum error recorded of 2.78mm for y-axis and z-axis deviations in this work renders the proposed method unsuitable for the industry in its current version. However, with the recent launch of the second generation of the Kinect sensor (Kinect V2) with improved depth resolution coupled with improved object detection algorithms, it is anticipated that the motion tracking error of less than 2mm would be achieved.

IR interference between the far and the near sensors compelled the motion sensing zone to be divided into mutually exclusive far and near motion sensing zones. The second generation of depth sensors are expected to be significantly less affected by IR interference and therefore a motion tracking setup with the far sensor tracking the moving wheel hub along the entire length of the loading workstation can be used. This setup will enable the far sensor to constantly track the moving hub for major disruptions whereas the near sensor can track the motion more precisely and determine misalignments more accurately.

In this study, the Kinect sensors are not re-calibrated and therefore object recognition and tracking quality degrades as the object moves away from the centre of the field of view of the sensor. A calibration method is needed to enhance the accuracy of coordinate mapping between the sensor coordinate system and the real world coordinate system and this is expected to enhance the accuracy of motion tracking and feature recognition.

Finally, the wheel hub mounted on the vehicle axle consists of additional components such as the brake disc callipers and in some cases the drum brake setup is installed. Therefore, the tracking method proposed here will need to be amended to recognise the wheel hub and recognise the alignment features in the presence of such components.

6.6 Chapter summary

This chapter presented the implementation of the proposed framework for digitisation of manual manufacturing task knowledge from the manual assembly of Lego blocks. This task was captured, segmented, modelled and the manufacturing knowledge constituents such as task strategy, precise human gestures, workpiece grasp and release techniques, and human motion mechanics were extracted, decoded and reproduced successfully. The chapter also presented how the framework and the task capture methods proposed within it could be implemented to capture and digitise the manufacturing environment within which the tasks occur. The next chapter presents the testing and validation of the framework using 3 case studies.

CHAPTER 7

7 VALIDATION

This chapter presents 3 case studies in which 3 different manual tasks are used to test the performance of the digitisation framework. The case studies are designed to test certain methods and tools from the digitisation framework that were not tested at the implementation stage. To validate the framework, it is benchmarked against an existing method of extracting task knowledge from a real-world composite layup task. Finally, through the case studies the breadth and depth of investigation conducted in this research is summarised.

This chapter aims to achieve the following objectives:

- Introduce the case studies.
- Describe the implementation of the framework for each case study and present the outcomes.
- Present the performance of the framework against the benchmark.
- Summarise the breadth and depth of investigation conducted in this research.

7.1 Overview

The proposed digitisation framework was developed and successfully implemented by digitising the knowledge embedded within a simplified assembly task and an assembly task environment (sections 6.4 and 6.5). A validation study will provide an understanding of the framework through different real-life-like task examples (case studies) and will cover additional features and functions of the framework that are not used in the implementation phase. The study will also gauge the efficacy of the framework by evaluating whether it can deliver the 5 functionality measures in the case studies.

Case studies provide an experimental platform for contextual analysis of a limited number of parameters. Researchers have used case studies as an important tool for many years across a broad spectrum of research disciplines. Yin (1984) defines case study research as “an empirical inquiry that investigates a contemporary phenomenon within its real-life context; when the boundaries between phenomenon and context are not clearly evident; and in which multiple sources of evidence are used”.

While most case studies are qualitative in nature, those adopted in this research are quantitative where the research data is gathered by experimental means. Three case studies are chosen, namely, (1) digitisation of pen assembly task knowledge, (2) digitisation of Ikea table assembly task knowledge and (3) digitisation of manual composite layup task knowledge. In these case studies, the digitisation framework is used as proposed to extract, decode and reproduce the manufacturing knowledge embedded within the tasks. In the third case study, the knowledge digitised by the framework is also benchmarked against the knowledge extracted by other means from the same task. The selection of case studies is broadly based on the following factors:

i. The case studies must include human actions of varying difficulty during task execution to test the effectiveness of using gaming interface sensors to capture different types of human motions.

ii. The case studies must include different types of workpieces with varying complexity to test the effectiveness of using gaming interface sensors to recognise and track different workpieces in real-time.
iii. The case studies must have the requirements that need the use of the features and functions of the framework not tested before.
iv. The tasks chosen must be not be complex enough to warrant the use of sophisticated motion capture and image processing algorithms because each of these areas are formative research subjects in themselves and are outside the scope of this research.

The case studies are briefly tabulated in Table 11:

Table 11: Case study description

S. No.	Case Study	Key Feature	Key Challenge
1	Digitisation of pen assembly task knowledge.	Human-workpiece interactions are segmented using the <u>trajectory-change-sampling</u> method. Object recognition and tracking using the edge detection method.	Recognition and tracking of small workpiece components based on depth and colour image processing. Inferring human actions from workpiece change tracking.
2	Digitisation of Ikea table assembly task knowledge.	Human-workpiece interactions are segmented using the <u>time-sampling</u> method. Use of the second generation of the Kinect sensor (Kinect V2). Object recognition and	Use of microphones to capture sound data and isolation of problem solving sessions from the human-workpiece interaction data. Use of two Kinect sensors to capture the task, especially the first use of

		tracking using the depth and brightness values of its pixels. Comparison of assembly skills between two users.	the Kinect V2 sensor. Comparison between Kinect V1 and V2 for evaluation of the Kinect V2 capability for this research. Using the framework to capture and digitise the assembly task performed by another user with no prior training or experience in the task.
3	Digitisation of manual composite layup task knowledge	The task is a <u>real world manufacturing</u> example which is at the interface between assembly and machining and therefore is completely different from other tasks in this research. The workpiece in this case is a deformable composite prepreg ply that is formed into a complex shape. Object recognition and tracking is fairly complex. Human actions are complex and continuous in nature with specific	Selection between the 3 proposed data segmentation techniques is difficult because the task displays characteristics that suit all 3 techniques. Workpiece recognition and tracking needs a completely different approach. Complex and precise human actions involved in the task require the motion tracking method to be much more accurate and reliable than the ones used in the tasks earlier.

		ply manipulation techniques embedded within them, which is a new constituent of task knowledge. The knowledge from this task is extracted before using other means and therefore will provide a benchmark for this research.	The task is performed at a different location to the one in which this research is carried out. Therefore, the framework must deal with uncertainties of setup and task environment.

7.2 Case Study 1: Pen assembly task

In this study, the task chosen is the manual assembly of a pen. The human-workpiece interactions involved during the assembly are captured and recorded using the Kinect, the continuous interaction data is segmented into discrete human action and workpiece observation states, the states are modelled using HMM to generate human-workpiece interaction models and these models are analysed to extract and decode the manufacturing knowledge associated with the assembly task. This case study differs from the others used in this research in 4 main areas:

i. Workpiece recognition and tracking method.
ii. Using workpiece tracking data to compensate for loss of human action data.
iii. Segmentation method.
iv. Representation of human action and workpiece states.

The pen assembly task is performed in the laboratory under controlled task environment conditions. Therefore, only the human and the workpiece were tracked during the task and the knowledge of the task environment was not digitised.

7.2.1 Choice of task

The pen assembly task is selected in accordance with the factors listed in section 7.1 and for the following additional reasons.

i. In this assembly task, the changes in workpiece states are gradual and continuous in nature and not in prominent chunks as seen in the Lego blocks assembly task. Therefore, a different technique is needed to segment the workpiece states.
ii. Workpiece tracking is more complex due to the small size and grayscale of the components. This requires the use of different object recognition and tracking method that does not rely on colour differentiation.
iii. The duration of the task is small and therefore the number of discrete states generated from the captured data will also be small. This ensures that more complexity is not added to the modelling and analysis phases of the digitisation process for this task.

7.2.2 Task description

In this task, the human uses his left and right hands to manipulate the two components of the pen in order to assemble it. The workpiece here refers to the two components of the pen, namely the left component and the right component.

Sequence of steps in the assembly task

i. Find the two components of the pen within the pre-defined virtual box on the table.
ii. Grasp the two components, one in each hand and move them to the front of the sternum.
iii. Move the two components towards each other until they mate.
iv. Upon mating, rotate the right component with respect to the left, the left being stationary, in order to complete the threaded fit assembly of the pen.

Task rules

The following rules are followed while performing the task and these rules are programmed into the task capture step of the digitisation process to minimise its complexity:

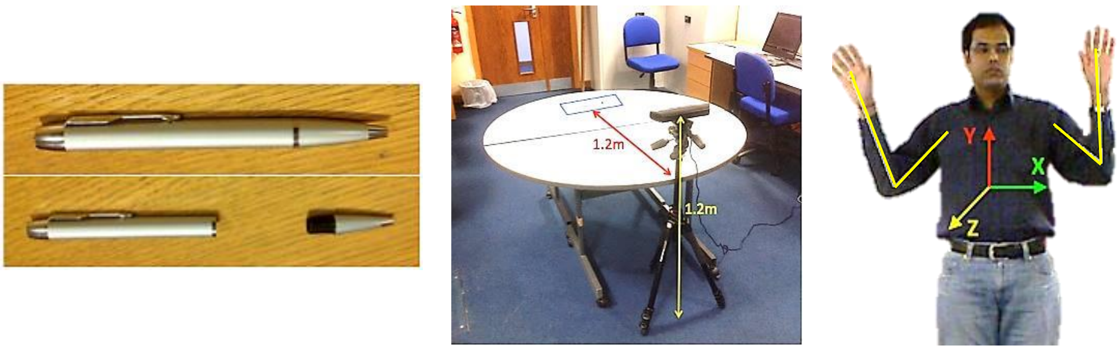
i. The assembly is executed within the pre-defined virtual zone in front of the human sternum. This rule cuts down the time taken by the task capture function by focussing the workpiece recognition and tracking to a small area rather than the entire 3D space in the visible view of the Kinect sensor.

ii. The task is performed at a normal speed so that real-life-like conditions can be simulated for capture.

iii. The same pen is used every time the task is run to capture data. Though depth-imaging processing is predominantly used for object recognition and tracking, in some instances when depth data is not reliable, RGB data is used. Therefore, using the same pen does not add to the complexity of workpiece tracking.

iv. The left and right components of the pen are held in left and right hands respectively without making a switch midway through the task. Again this rule does not add to the complexity of workpiece tracking during the task.

7.2.3 Task setup



(a) (b) (c)

Figure 60: (a) Workpiece components and assembled workpiece (b) Experiment Setup (c) Human arm tracking

The table is the assembly workstation with a pre-defined virtual box where the workpiece components are initially placed. A Kinect sensor is used to capture and record the entire assembly task. It is mounted on a tripod was placed at a distance of 1.2m from the human operator and at a height of 1.2m from the floor (Figure 60). The distance of 1.2m is chosen because it was the maximum distance at which the object recognition algorithm could track the workpiece components reliably. Using the standard human skeleton tracking functions of the library, the 3D motion of the human's left and right arms (shoulder, elbow and hand) are recorded throughout the assembly task. At the same time, the workpiece components are also tracked using a moving object recognition algorithm. A Java based software development platform called 'Processing' version 2.1 is used to write the image processing and skeletal tracing code and 'SimpleOpenNI' version 1.96; an open source library is used to interface with the Kinect sensor.

7.2.4 Implementation and testing of the framework

Step 1: Capture

The Kinect sensor captures the entire task performed by the human and the data produced is processed using standard and bespoke software functions to digitise and record human actions and the effects of those actions on the workpiece components. Sound is not captured in this implementation.

Workpiece component identification

The workpiece components must be placed within the pre-defined virtual area, marked in red outline in Figure 61, on the table in preparation for assembly. The workpiece identification algorithm scans this virtual area and reports the presence or absence of the two pen components of the workpiece using the blue and red outlines respectively. This step simulates the presence or absence of assembly components in their storage racks in an actual assembly workstation.

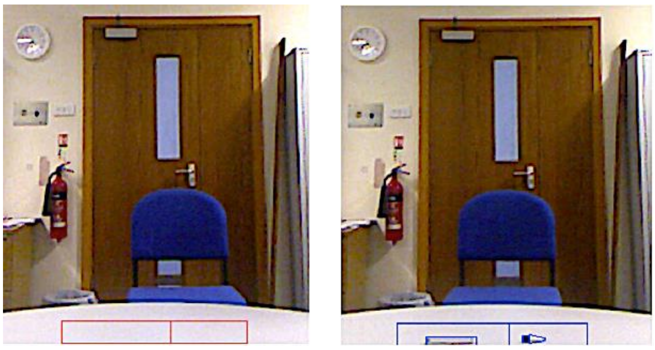
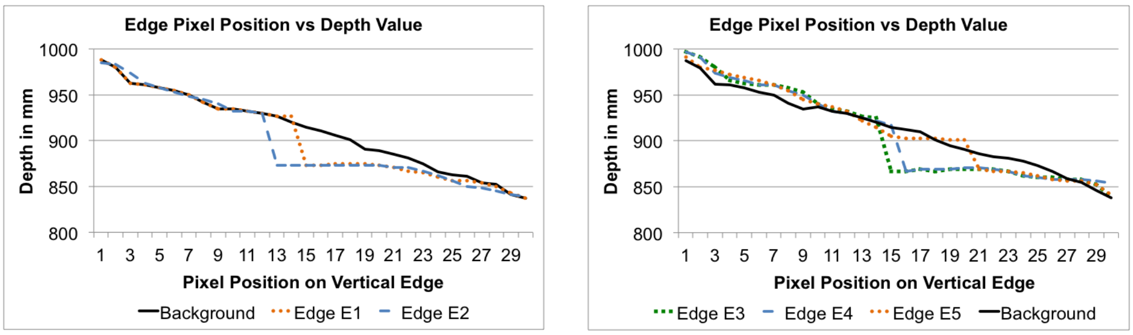

(a) (b)

Figure 61: The virtual area (red outline) for placement of workpiece components (b) Identification of components

The pre-defined area is divided into one pixel wide columns. For each column, the depth and RGB values of each pixel in the column are compared with those of the next pixel in the same column. Any abrupt change in pixel values against those of the background pixels is marked and this change pattern is compared against known pixel patterns for workpiece component edges E1 to E5 as shown in Figure 62. By recognising these edges, the workpiece components are identified.

Figure 62: Key workpiece component edges E1 to E5

The abrupt changes in the depth and RGB values of pixels corresponding to the workpiece component edges are shown in the charts in Figure 63.



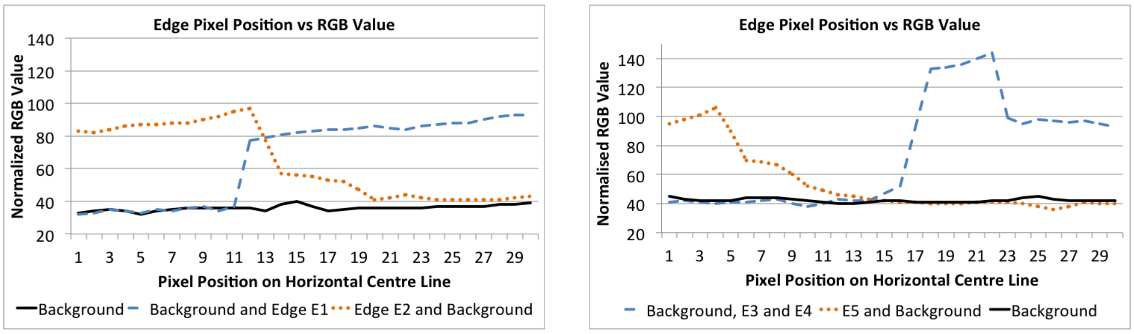


Figure 63: Depth pixel patterns of background and component edges E1, E2, E3, E4 and E5

Tracking human-workpiece interactions

The assembly process begins when the human picks up the workpiece components, left and right, using his left and right hand respectively. The human then brings them together until the two components are mated and cannot come any closer as shown in Figure 64.

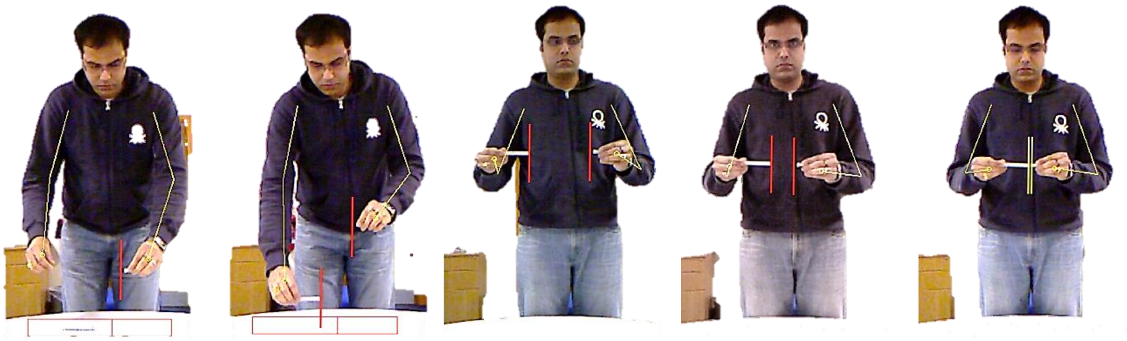
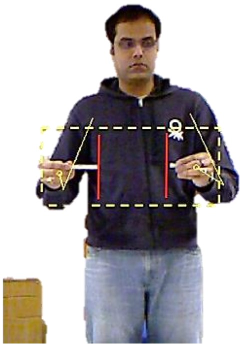
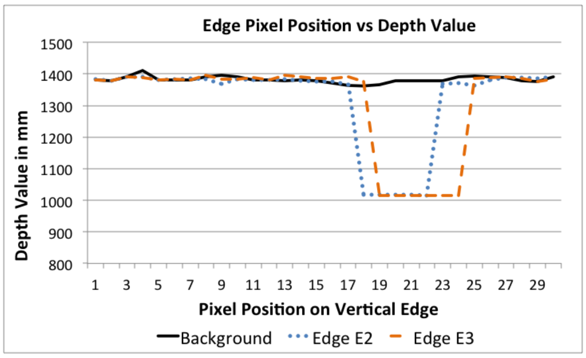


Figure 64: Human motion tracking (arms) and workpiece component tracking

During this process, the left and right human arm joints (shoulder, elbow and hand) as well as the key component edges are continuously identified and tracked until mating of the components occurs. The key component edges E2 and E3 (see Figure 62), which converge in the X direction during the assembly are detected by using an edge detection method, similar to the edge detection process of Figure 63, that distinguishes the pixel depth pattern of the edge against those of the background pixels (Figure 65b).



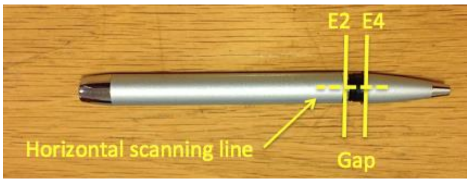


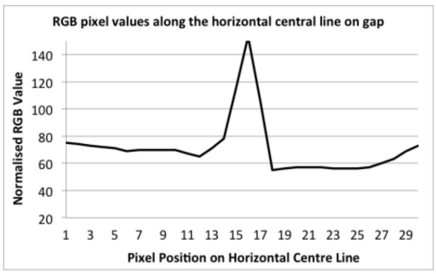
(a) (b)

Figure 65: (a) Component edge recognition (b) depth pixel pattern at the edges

The edge detection and tracking is confined to a virtual box, spanning the two human hands in X direction, 80 pixels in Y direction and a range of 900mm to 1150mm in Z direction in front of the human, to reduce the computational load of the algorithm. Since the depth change in pixels belonging to the edges as compared to the background is significant higher (greater than 200mm), there is no need to examine the RGB values of the pixels for edge detection.

In the final assembly step, the human performs the threaded fit process by rotating the right component with respect to the left until no further rotation is possible. Since the Kinect cannot detect rotation motion, the edges E2 and E4 are tracked continuously and the narrowing gap between them is computed. At this stage, only the RGB values of the pixels belonging to edges E2 and E4 are compared with the rest of the pixels on the horizontal scanning line because the Kinect is not able to reliably resolve the depth values of pixels in this small area. Because the pixels belonging to the gap are darker than others on the scanning line, the edges E2 and E4 are identified (Figure 66).





(a) (b)

Figure 66: Gap identification (b) RGB value of pixels along the scanning line

The capture system then approximates the amount and direction of rotation applied. The final threaded fit process is shown in Figure 67.

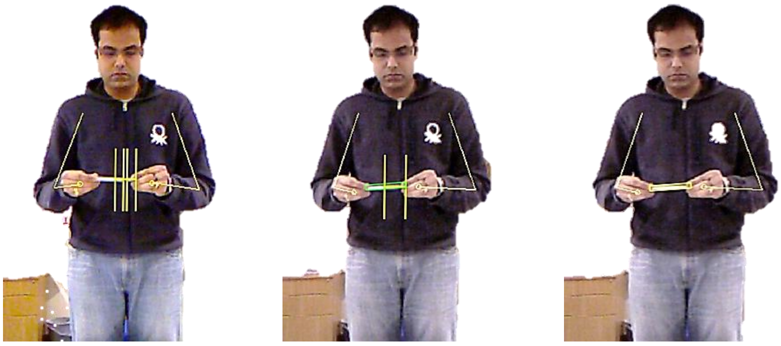


Figure 67: Tracking edges E2 and E4 for threaded fit action and identifying the completed pen assembly

During the assembly process, x, y and z coordinates of the human's arm joints (shoulder, elbow and hand) as well as the x, y and z coordinates of the key workpiece component edges E2, E3 and E4 are captured simultaneously in real-time along with the timestamps. All this captured data is recorded in a spreadsheet (Figure 68).

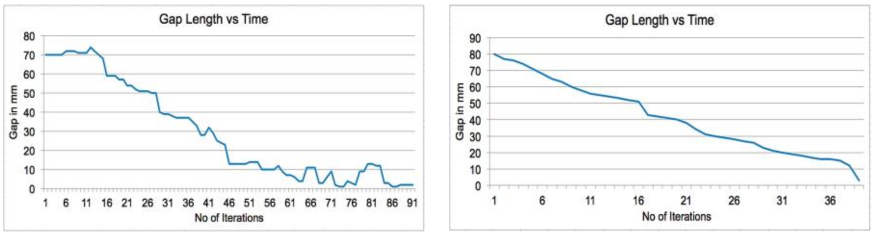
No.	LH_Hand	LH_Hand_Y	LH_Hand_Z	LS_X	LS_Y	LS_Depth	LH_Elbow_X	LH_Elbow_Y	LH_Elbow_Z	LH_Shoulder_X	LH_Shoulder_Y	LH_Shoulder_Z	RH_Hand_X
1	95	243	1065	250	442	1280	168	249	1359	233	194	1647	395
2	96	243	1053	250	442	1280	167	248	1349	233	193	1636	395
3	92	243	1050	250	442	1280	165	249	1345	233	193	1628	399
4	92	245	1074	250	442	1280	165	257	1362	234	192	1627	399
5	71	257	1142	250	442	1280	167	273	1415	237	193	1641	399
6	71	290	1287	250	442	1280	177	282	1514	244	196	1672	381
7	99	335	1471	250	442	1280	192	299	1616	254	199	1715	381
8	149	382	1610	250	442	1280	214	296	1702	262	202	1751	381
9	190	387	1635	250	442	1280	229	295	1729	266	204	1764	381
10	211	390	1576	250	442	1280	237	295	1706	265	209	1746	381
11	225	391	1470	250	442	1280	239	301	1639	262	220	1709	381
12	227	401	1392	250	442	1280	238	305	1582	259	233	1675	381
13	230	406	1332	250	442	1280	235	314	1531	256	239	1645	381
14	233	418	1295	250	442	1280	233	321	1498	253	245	1627	381
15	234	428	1273	250	442	1280	232	327	1478	252	247	1616	375

Figure 68: Snapshot of the spreadsheet with human action and workpiece data

Filtering

The coordinates of the human operator's hand joints and the workpiece component edges are extracted from the depth image stream sent by the Kinect at 30fps. The human skeletal tracking data as well as the workpiece motion data is noisy due to the low resolution of the Kinect sensor. Therefore, to reduce the noise and to provide more time for filtering and smoothing of the depth image stream in real-time, the image acquisition rate from the Kinect was slowed down to 6fps, which is still considerably fast as compared to the normal speed of the assembly task.

Filtering was needed to keep out the inherent noise in the motion data and was achieved by ignoring the data points that lie outside of the known interaction boundaries of the assembly operation. For example, hand motion diverging for a few milliseconds within the continuous converging motion is considered noise and is ignored. Smoothing was achieved by averaging out the variations in the filtered data points within a pre-defined deviation range to minimise the effect of high frequency noise that still crept in. The effect of filtering and smoothing of motion data on computation of gap width between the two components during assembly is shown in Figure 69.



(a) (b)

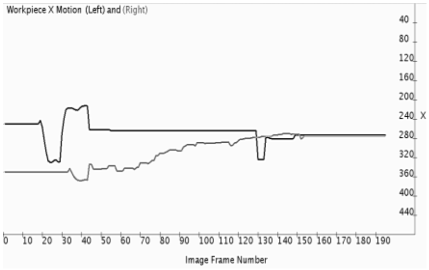
Figure 69: Gap tracking (a) before filtering and (b) after filtering

<u>Step 2 – Segment</u>

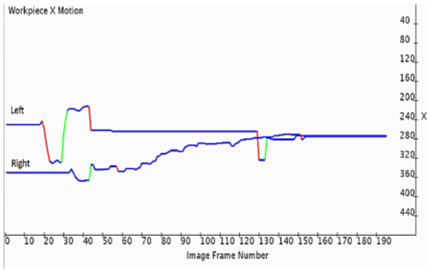
The continuous human and workpiece data is segmented into human action states and workpiece states using the trajectory-sampling method. The spreadsheet containing continuous raw human motion and workpiece tracking data is parsed and the data is segmented according to the following 6 steps.

i. *Segregation of workpiece motion into workpiece motion primitives*

   Workpiece motion data is segmented at places where abrupt change in motion direction is detected and the resulting segments are workpiece motion primitives. In Figure 70b, red segments indicate a sharp drop in direction of motion, green segments indicate a sharp rise in direction of motion and blue segments indicate gradual or no change in direction of motion.



(a)



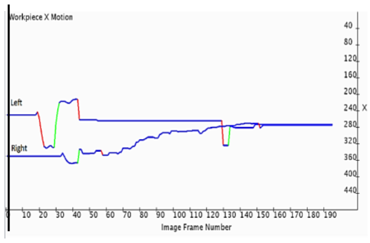
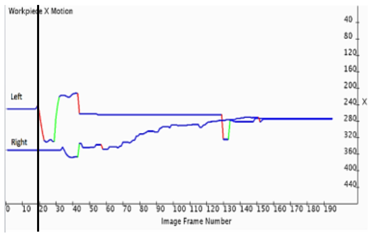
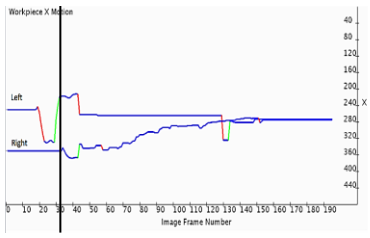
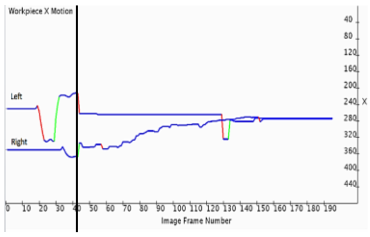
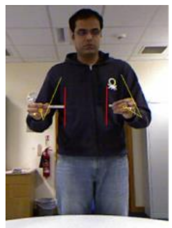
(b)

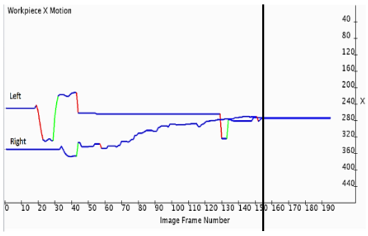
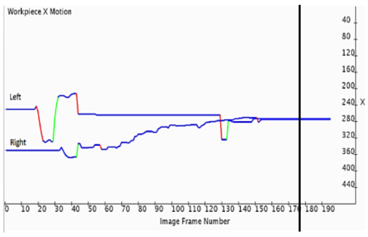
Figure 70: workpiece x-motion (b) Workpiece x-motion primitives

ii. *Generation of workpiece states*

The workpiece motion primitives are mapped to the corresponding steps of the assembly sequence (Table 12).

Table 12: Workpiece motion primitives mapped to assembly sequence

Frame No.	Workpiece Motion Primitive	Correspoding Assembly Step
1		
18		
33		
42		

154		
195		

The image frame numbers at which each of these primitives terminates are designated as workpiece states. The resulting workpiece states are listed in Table 13 and illustrated in Figure 71. Each state is annotated with its own data such as its image frame number, spatial position and whether the rotation action is applied to the component or not.

Table 13: Workpiece states for left and right workpiece components

State	Frame no.	Left Component		Right Component	
		X, Y, Z position	Rotation applied	X, Y, Z position	Rotation applied
0	1	250, 250, 1280	0	350, 252, 1271	0
1	18	250, 250, 1280	0	350, 252, 1271	0
2	33	217, 278, 1477	0	350, 252, 1271	0
3	42	212, 264, 1101	0	366, 255, 1126	0
4	154	273, 237, 1113	0	275, 236, 1116	0
5	195	273, 237, 1113	0	275, 236, 1116	1

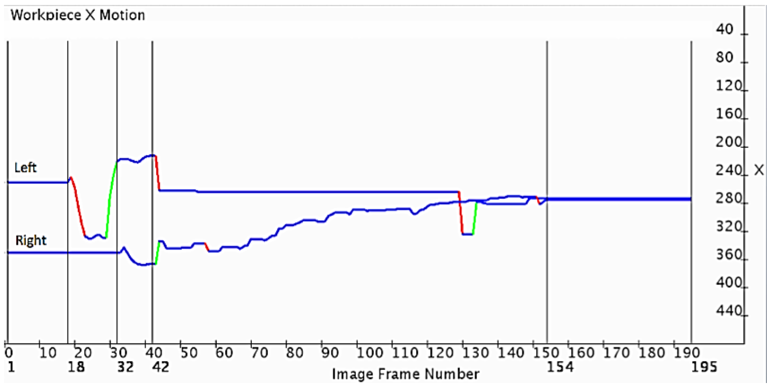


Figure 71: Workpiece states identified on the workpiece x-motion chart

iii. *Generation of human action states*

Human motion is segmented into human action states according to their spatio-temporal dependencies on the corresponding workpiece states. An example of human motion segmentation based on the workpiece states is shown in Figure 72 and the resulting human action states in Table 14.

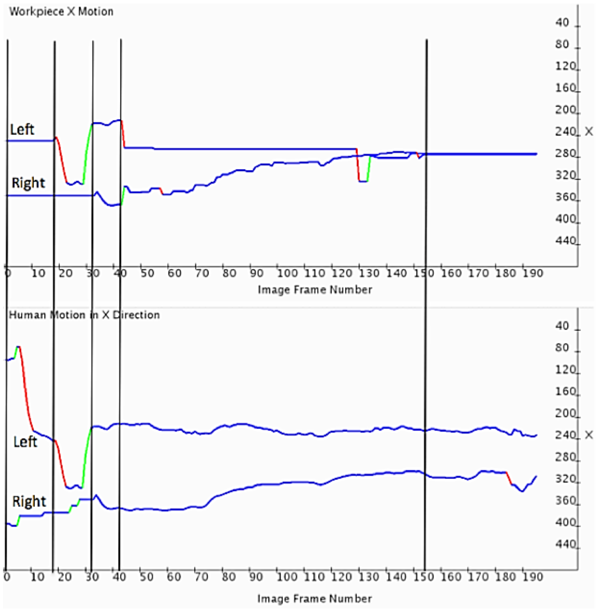
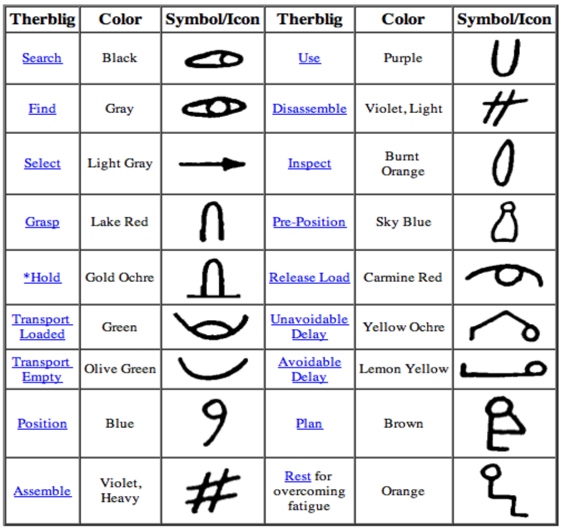


Figure 72: Segmentation of human motion according to the workpiece states

Table 14: Human action states (for left and right hand)

State	Left Hand				Right Hand		
	Frame no.	From position (X, Y, Z)	To position (X, Y, Z)	Rotation applied	From position (X, Y, Z)	To position (X, Y, Z)	Rotation applied
0	1	95, 243, 1065	-	0	395, 243, 1103	-	0
1	18	95, 243, 1065	243, 441, 1286	0	395, 243, 1103	375, 325, 1147	0
2	18	243, 441, 1286	-	0	375, 325, 1147	-	0
3	33	243, 441, 1286	217, 278, 1477	0	375, 325, 1147	351, 325, 1197	0
4	33	217, 278, 1477	-	0	351, 325, 1197	-	0
5	42	217, 278, 1477	212, 264, 1101	0	351, 325, 1197	366, 255, 1138	0
6	42	212, 264, 1101	-	0	366, 255, 1138	-	0
7	154	212, 264, 1101	225, 244, 1115	0	366, 255, 1138	303, 235, 1118	0
8	154	225, 244, 1115	-	0	303, 235, 1118	-	0
9	195	225, 244, 1115	233, 238, 1118	0	303, 235, 1118	308, 238, 1119	1
10	195	233, 238, 1118	-	0	308, 238, 1119	-	0

Therbligs: Therbligs are used in this study to represent human action states on the basis of what the states are intended for (Table 15). By doing this, every human state is associated with a work function within the overall assembly task. Therbligs were first proposed by the industrial psychologists Frank and Lillian Gilbreth for as a way of classifying human motions in a work task. The Gilbreths claimed that complex human tasks consist of 18 basic motions such as searching, gripping, moving and positioning (Figure 73) which could be used to study and improve human motion in work environments, such as an assembly line or a production workshop, and improve work efficiencies (Ferguson, 2000).



Therblig	Color	Symbol/Icon	Therblig	Color	Symbol/Icon
Search	Black		Use	Purple	
Find	Gray		Disassemble	Violet, Light	
Select	Light Gray		Inspect	Burnt Orange	
Grasp	Lake Red		Pre-Position	Sky Blue	
*Hold	Gold Ochre		Release Load	Carmine Red	
Transport Loaded	Green		Unavoidable Delay	Yellow Ochre	
Transport Empty	Olive Green		Avoidable Delay	Lemon Yellow	
Position	Blue		Plan	Brown	
Assemble	Violet, Heavy		Rest for overcoming fatigue	Orange	

Figure 73: The 18 Therbligs

Table 15: Therbligs associated with human action states

State	Frame no.	Left and Right Hand	
		Therblig	**Remarks**
0	1	Find (F)	Human finds the workpiece components on the table
1	18	Transport Empty (TE_L)	Left hand travels to the left component
2	18	Grasp (G_L)	Left hand grasps the left component
3	33	Transport Empty (TE_R)	Right hand travels to the right component
4	33	Grasp (G_R)	Right hand grasps the right component
5	42	Transport Loaded (T_L)	Both hands travel to the beginning of the prepositioning stage (stage 7)
6	42		
7	154	Pre-Position (PP)	Both hands pre-position the components by bringing them together and aligning them
8	154	Position (P)	Both the components have mated for the final threaded fit assembly stage
9	195	Assemble (A)	Both hands thread fit the two components with the right hand doing the rotation motion to complete the assembly operation
10	195		

iv. *Addressing unexpected motion primitives*

Unexpected workpiece motion primitives that are not associated with any assembly step but have corresponding human actions can be regarded as human response to a problem. Otherwise as in this example, these stray primitives are just noise and can be ignored (Figure 74).

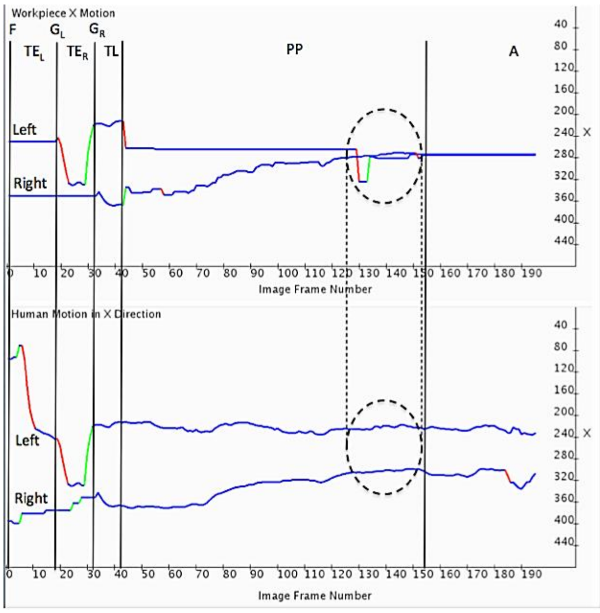


Figure 74: Unexpected workpiece motion primitives with no human reaction

Step 3: Model

The discrete human action and workpiece states generated in the previous step are now ready to be modelled using HMM. The human states of ‘Find (F)’ and ‘Position (P)’ are dropped during modelling because those are just snapshots of the human pose and not human actions that manipulate the workpiece. The nomenclature of the states is based on the Therblig they individually represent.

HMM generation

Two different task scenarios are captured for modelling. In scenario A, the human uses both hands simultaneously to position the two components before thread fitting them together whereas in scenario B, the human positions one component first and then the other.

The workpiece o sequence observed for scenario A is

$$O_A = \{WTL,\ WTR,\ WGL,\ WGR,\ WPPL,\ WPPR,\ WPL,\ WPR,\ WPA\}$$

The workpiece state sequence observed for scenario B is

$O_B = \{WTL,\ WTR,\ WGL,\ WPPL,\ WPL,\ WGR,\ WPPR,\ WPR,\ WPA\}$

HMM models λ_A and λ_B are generated for the two unique workpiece state sequences O_A and O_B (Figure 75).

P	TEL	TER	GL	GR	TLL	TLR	PPL	PPR	A
	0.47	0.47	0.01	0.01	0.01	0.01	0.01	0.01	0.01

A	TEL	TER	GL	GR	TLL	TLR	PPL	PPR	A
TEL	0.10	0.30	0.46	0.10	0.01	0.01	0.01	0.01	0.01
TER	0.05	0.10	0.35	0.46	0.01	0.01	0.01	0.01	0.01
GL	0.01	0.02	0.20	0.05	0.35	0.35	0.01	0.01	0.01
GR	0.01	0.01	0.05	0.30	0.21	0.40	0.01	0.01	0.01
TLL	0.01	0.01	0.01	0.05	0.20	0.31	0.30	0.10	0.01
TLR	0.01	0.01	0.01	0.05	0.21	0.30	0.10	0.30	0.01
PPL	0.01	0.01	0.01	0.01	0.08	0.08	0.10	0.10	0.60
PPR	0.01	0.01	0.01	0.01	0.08	0.08	0.10	0.10	0.60
A	0.01	0.01	0.01	0.01	0.01	0.05	0.10	0.10	0.70

B	WTL	WTR	WGL	WGR	WPPL	WPPR	WPL	WPR	WPA
TEL	0.70	0.22	0.02	0.01	0.01	0.01	0.01	0.01	0.01
TER	0.22	0.70	0.01	0.02	0.01	0.01	0.01	0.01	0.01
GL	0.01	0.12	0.80	0.02	0.01	0.01	0.01	0.01	0.01
GR	0.01	0.01	0.12	0.80	0.02	0.01	0.01	0.01	0.01
TLL	0.01	0.01	0.01	0.26	0.66	0.02	0.01	0.01	0.01
TLR	0.01	0.01	0.01	0.01	0.26	0.66	0.02	0.01	0.01
PPL	0.01	0.01	0.01	0.01	0.11	0.12	0.56	0.16	0.01
PPR	0.01	0.01	0.01	0.01	0.11	0.12	0.16	0.56	0.01
A	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.92

HMM model λ_A

P	TEL	TER	GL	GR	TLL	TLR	PPL	PPR	A
	0.47	0.47	0.01	0.01	0.01	0.01	0.01	0.01	0.01

A	TEL	TER	GL	GR	TLL	TLR	PPL	PPR	A
TEL	0.1	0.3	0.46	0.1	0.01	0.01	0.01	0.01	0.01
TER	0.05	0.1	0.15	0.65	0.01	0.01	0.01	0.01	0.01
GL	0.01	0.01	0.1	0.04	0.6	0.21	0.01	0.01	0.01
GR	0.01	0.01	0.05	0.1	0.21	0.6	0.01	0.01	0.01
TLL	0.01	0.01	0.01	0.05	0.2	0.31	0.3	0.1	0.01
TLR	0.01	0.01	0.01	0.05	0.21	0.3	0.1	0.3	0.01
PPL	0.01	0.01	0.01	0.5	0.08	0.07	0.01	0.01	0.3
PPR	0.01	0.01	0.01	0.01	0.08	0.08	0.1	0.3	0.4
A	0.01	0.01	0.01	0.01	0.01	0.05	0.2	0.2	0.5

B	WTL	WTR	WGL	WGR	WPPL	WPPR	WPL	WPR	WPA
TEL	0.7	0.22	0.02	0.01	0.01	0.01	0.01	0.01	0.01
TER	0.22	0.7	0.01	0.02	0.01	0.01	0.01	0.01	0.01
GL	0.01	0.12	0.8	0.02	0.01	0.01	0.01	0.01	0.01
GR	0.01	0.01	0.12	0.8	0.02	0.01	0.01	0.01	0.01
TLL	0.01	0.01	0.01	0.26	0.66	0.02	0.01	0.01	0.01
TLR	0.01	0.01	0.01	0.01	0.26	0.66	0.02	0.01	0.01
PPL	0.01	0.01	0.01	0.01	0.11	0.12	0.56	0.16	0.01
PPR	0.01	0.01	0.01	0.01	0.11	0.12	0.16	0.56	0.01
A	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.92

HMM model λ_B

Figure 75: HMM models for the pen assembly task

Optimisation of HMM parameters: In this example, the two HMM models λ_A and λ_B are put through the Baum Welch algorithm to optimise their parameters. For both models the algorithm did not go past the first run, which means that the initial estimation of parameters were good enough for the models to closely represent their task scenarios.

HMM evaluation – picking the right model

Stochastic evaluation is used to determine the most likely HMM model that closely represents a given task scenario. The task scenario is given in the form of a workpiece observation sequence.

Consider a task scenario represented by the given workpiece observation sequence $O_Q = \{WTL,\ WTR,\ WGL,\ WGR,\ WPPR,\ WPPL,\ WPR,\ WPL,\ WPA\}$, which is different from O_A and O_B.

Therefore, by stochastic evaluation $P(\lambda_A \mid O_Q)$ is compared with $P(\lambda_B \mid O_Q)$ and the model with the highest probability is picked. Therefore, by using the 'Forward' algorithm:

$P\ (\lambda_A \mid O_Q)\ =\ 3.21e-10$

$P\ (\lambda_B \mid O_Q)\ =\ 6.11e-10$

Since $P\ (\lambda_B \mid O_Q) > P\ (\lambda_A \mid O_Q)$, HMM model λ_B is picked for the given observation sequence O_Q.

Step 4: Extract

Once the most probable HMM model is identified for the given task scenario, the sequence of human action states responsible for that scenario can be extracted using the Viterbi algorithm. Once the human action sequence is extracted, detailed analysis can be performed on the individual states that make up the sequence to decode the knowledge embedded within them.

In the previous step, for observation sequence O_Q, HMM model λ_B is identified that most likely embodies the task scenario represented by O_Q. Using the 'Viterbi' algorithm, the most likely sequence of human actions H_Q that could produce O_Q is identified.

Therefore, $H_Q\ =\ \{TEL,\ TER,\ GL,\ GR,\ TLR,\ TLL,\ PPR,\ PPL, A\}$

Step 5: Decode

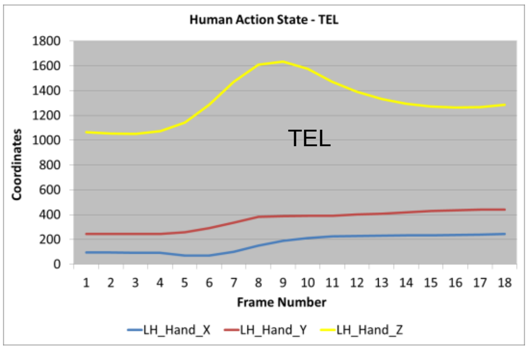
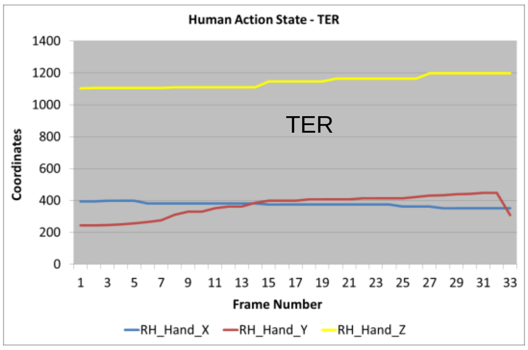
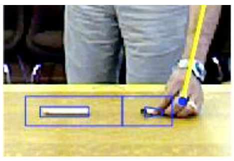
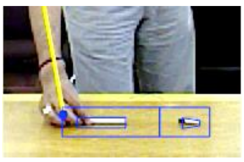
Given a task scenario, steps 4 and 5 have shown that the right HMM model that best represents that scenario can be picked and the human action sequence responsible for that scenario can be extracted. This sequence is made up of human action states that correspond directly to the changes the workpiece has undergone in the given task scenario. In this step, these human-workpiece interactions are further analysed to decode the manufacturing knowledge embedded within them.

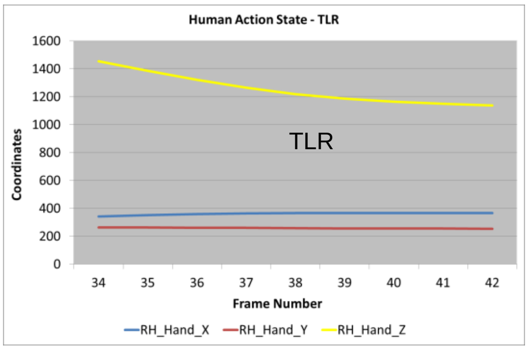
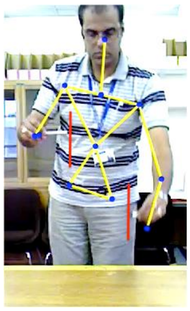
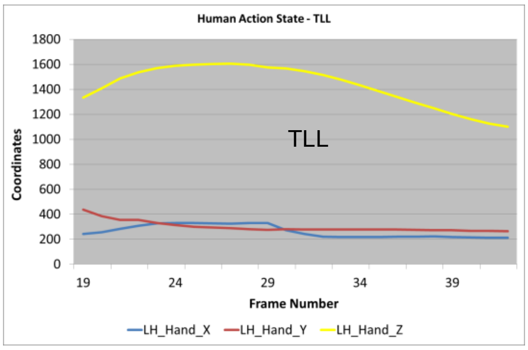
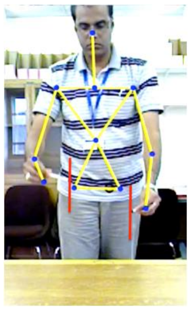
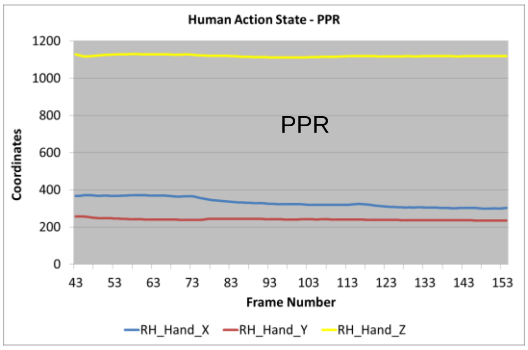
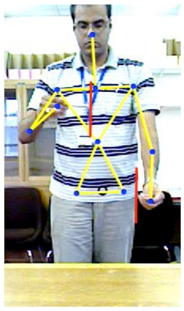
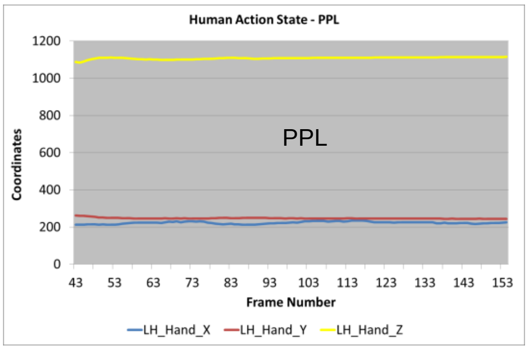
There are several constituents of manufacturing knowledge such as the task strategy adopted, nature and spatial characteristics of gestures made, mechanics of motion performed during the gestures, and workpiece manipulation techniques. These constituents can be decoded from the raw human action data and workpiece tracking data stored within the states that are extracted.

Extraction of states from the human action sequence

Consider the output sequence $H_Q = \{TEL, TER, GL, GR, TLR, TLL, PPR, PPL, A\}$ extracted from HMM model λ_B for the given observation sequence O_Q. The human action data within the states of H_Q is pulled out from the raw human motion data and mapped to the corresponding workpiece states from the observation sequence (Table 16).

Table 16: Human action states mapped to corresponding workpiece states

<table>
<tr><th>Human action state</th><th>Workpiece state in observed task status</th></tr>
<tr><td>
</td><td></td></tr>
<tr><td>
</td><td></td></tr>
<tr><td>There is no chart for this action state because the action is to ‘grasp’ the workpiece component, which cannot be captured by the Kinect.
GL</td><td></td></tr>
<tr><td>There is no chart for this action state because the action is to ‘grasp’ the workpiece component, which cannot be captured by the Kinect.
GR</td><td></td></tr>
</table>

	
	
	
	
There is no chart for thread fitting action because the Kinect cannot track hand rotation movements. A	

Decoding of manufacturing knowledge

Multiple constituents of manufacturing knowledge can be decoded from the extracted states and human action data. These constituents include the following but are not limited to:

- Task execution strategy.
- Nature and spatial characteristics of human gestures while working on the workpiece.
- Workpiece manipulation techniques.
- Mechanics of human movements such as body bending angle, angles between upper and lower arms, body orientation with respect to the workpiece, etc.

<u>Task execution strategy</u>

The task execution strategy can be decoded by determining the sequence of human actions responsible for the given sequence of workpiece states from the most likely HMM model. From the human action sequence, the following knowledge can be decoded:

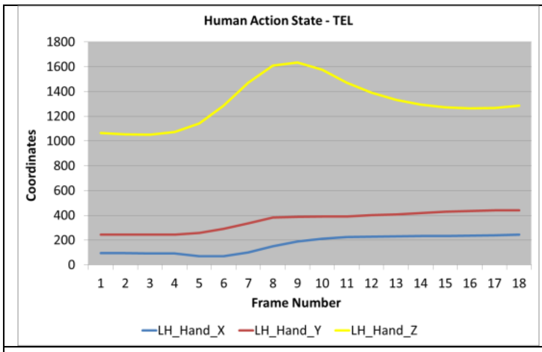
- Plan and approach of task execution by breaking it down into sub-tasks i.e. the action states
- Sequence of execution of the sub-tasks to achieve the main task. This sequence depends on the task scenario.
- Selections made during the task to choose specific actions from a repertoire of actions available to successfully complete the task. In this example, the human made a choice of positioning one component first and then the other component for thread fitting assembly. Problem-solving scenarios were not captured in this task therefore the expert's approach to an unforeseen problem could not be extracted.

<u>Nature and spatial characteristics of human gestures</u>

By visualising the human motion data within each extracted action state, the nature (trajectories and patterns) and spatial characteristics (3D coordinates) of human gestures with respect to the changes in the workpiece can be obtained.

The human motion data and corresponding workpiece change data for the extracted state 'TEL' is shown as an example (Table 17).

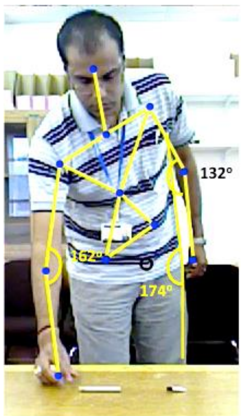
Table 17: Human motion data and gestures during the human action state TEL

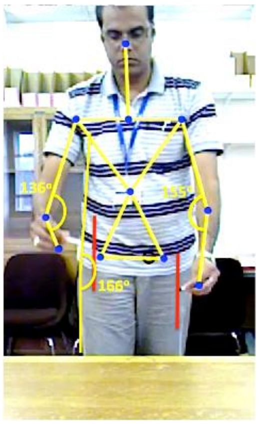
				
Human state 'TEL'	1	2	3	4

<u>Mechanics of human motion</u>

Motion mechanics such as body bending angle, angles between the upper and lower arms, and body orientation can be obtained using vector computing. This information is vital to understand the physical nuances of human skills exhibited during task execution. Table 18 illustrates how body bending angle and the angles between upper and lower arms can be annotated on image frames extracted from the action states.

Table 18: Mechanics of human motion annotated over the actual task images

	
• Angle between upper & lower arm (Left): 157^o • Angle between upper & lower arm (Right): 168^o • Body bending angle: 173^o	• Angle between upper & lower arm (Left): 132^o • Angle between upper & lower arm (Right): 162^o • Body bending angle: 174^o

	
• Angle between upper & lower arm (Left): 155^{o} • Angle between upper & lower arm (Right): 136^{o} • Body bending angle: 166^{o}	• Angle between upper & lower arm (Left): 41^{o} • Angle between upper & lower arm (Right): 28^{o} • Body bending angle: 176^{o}

In this case study, not only the interaction between the human and the workpiece components but also the interaction between the left and right hands of the human during the assembly task can be decoded. With further analysis of these interactions, it is possible to visualise and extract a particular human skill involved in the assembly task that allows the human to sub-consciously align the two pen components in y and z axis simultaneously as the two components are brought together for mating. During this pre-positioning stage of assembly, the components provide a visual feedback on their spatial position to the human who analyses it and adjusts the motion of his hands continuously to ensure alignment (Figure 76). Without this alignment, it wouldn’t be possible to mate the components, resulting in a failed assembly.

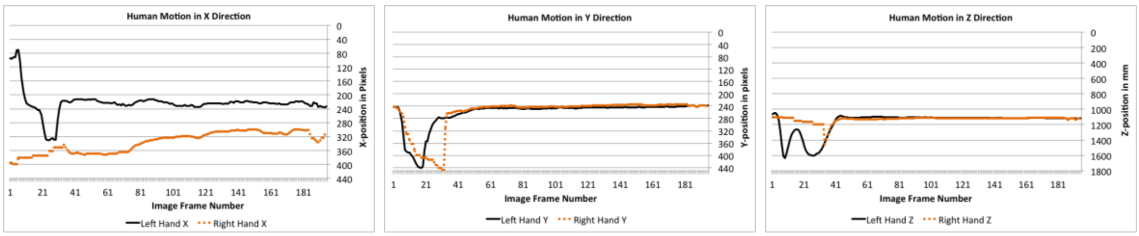


Figure 76: The pre-positioning stage for component alignment

<u>Step 5: Reproduce</u>

2D animation is produced as a medium to reproduce the human-workpiece interactions and augment this animation with manufacturing knowledge constituents extracted and decoded in this framework. An example of such an animation is shown in the form of static snapshots of the animation at various times of the task in Figure 77.

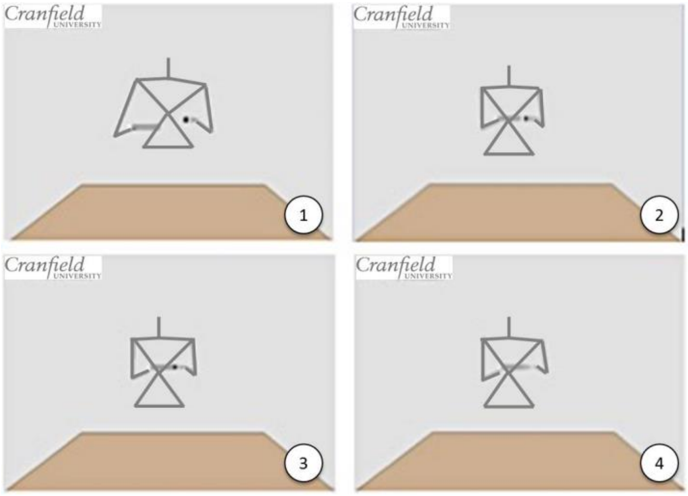


Figure 77: Human-workpiece interactions animated for a specific task scenario

Therefore, the 6-step digitisation framework has been successfully tested and demonstrated for digitisation of task knowledge embedded within the manual pen assembly task. Most of the work in this case study is also presented in a journal paper and a conference paper (See 'List of Publications').

7.3 Case Study 2: Ikea table assembly task

In this study, the task chosen is the manual assembly of an Ikea table. The human-workpiece interactions involved during the assembly are captured, segmented, modelled and the manufacturing knowledge associated with the assembly task is extracted and decoded. This case study differs from the others in 5 main areas:

1. Workpiece recognition and tracking method.
2. Segmentation method used to generate human action and workpiece states.
3. Representation of human action and workpiece states.
4. Capture of verbal inputs from the human during the task.
5. Digitisation of the assembly task performed separately by two humans for skill comparison using the framework.

The Ikea table assembly task was performed in the laboratory under controlled conditions. Therefore, only the human and the workpiece were tracked during the task and the knowledge of the task environment was not digitised.

This case study was implemented to test all the 6 steps of the framework on a real-life-like assembly task example. As this study is the fourth task to be digitised, task capture and segmentation steps of the framework had already matured into reliable processes. Therefore, more focus is given to modelling and knowledge extraction and decoding in this study.

7.3.1 Choice of task

The Ikea table assembly task is selected in accordance with the factors listed in section 7.1 and for the following additional reasons.

1. In this assembly task, the human actions are cyclical in nature with regular intervals between two prominent actions. Therefore, the time-sampling segmentation method, which has not been tested so far, is used in this study.
2. The workpiece tracking in this case is a real-time tracking of changes in workpiece configurations rather than its position, dimension or shape. A predominantly depth-based object recognition method that relies on the known geometry of the workpiece, which has not been tested so far, is used in this study.
3. A clear assembly sequence can be built into the task so that any workpiece components assembled out of turn can be flagged as a problem. The resolution of the problem can then be analysed as a separate task scenario.

7.3.2 Task description

In this task, the human assembles the different components of an Ikea table together to form the finished workpiece consisting of the table base and the four legs. The human, considered the task expert, manipulates the workpiece components and assembles the table on an assembly workstation. An assembly sequence is built into the task which if not followed is flagged as an error that needs resolving by the task expert. To shorten and simplify the assembly task, the legs of the table are not thread fitted into the base but simply placed on it.

Assembly sequence

i. Grasp and pick the base of the table from its storage area and place it on the workstation within the defined virtual box.
ii. Grasp and pick a table leg from the storage area and place it on the base at position number 1.
iii. Grasp and pick a table leg from the storage area and place it on the base at position number 2.
iv. Grasp and pick a table leg from the storage area and place it on the base at position number 3.
v. Grasp and pick the remaining leg from the storage area and place it on the base at position number 4 to complete the assembly task.

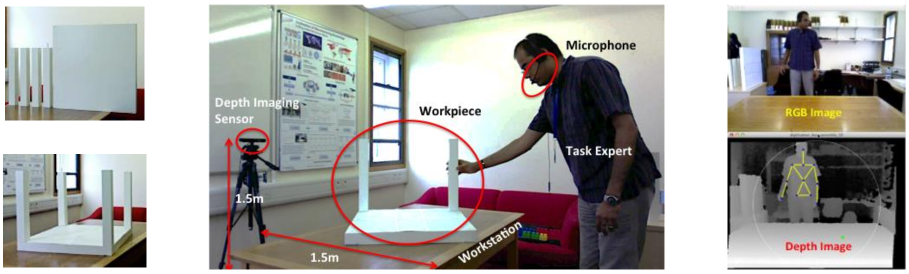
Task rules

Four rules are followed while executing the task and these rules are programmed into the task capture step of the digitisation framework to minimise its complexity.

i. The assembly is executed within the pre-defined virtual box on the workstation. This rule cuts down the time taken by the task capture function by focussing the workpiece tracking to a small area rather than the entire 3D space in the visible view of the Kinect sensor.
ii. The task is performed at a normal speed so that real-life like conditions can be simulated for capture.
iii. All the components of the table have the same colour (white) to test the ability of the workpiece tracking to recognise workpiece configuration using only the depth information.
iv. No other sound is generated during the task except for verbal instructions from the task expert. This rules ensures that the instructions are captured clearly and immediately upon being spoken.

7.3.3 Task setup

The task is performed on the assembly workstation with a pre-defined virtual area where the workpiece components are placed. A Kinect sensor is used to capture and record the entire assembly task. It is mounted on a tripod at a height of 1.5m from the floor and at a distance of 1.5m from the area where the human will perform the assembly. The distance of 1.5m was chosen because it was the optimum distance at which the object recognition algorithm could track the relatively large workpiece components reliably (Figure 78).



(a) (b) (c)

Figure 78: (a) Workpiece components and assembled workpiece (b) experiment setup (c) RGB and depth images with tracking of upper human body motion

The software development platform and the human motion tracking method are the same as the previous tasks. An online speech processing library from Google is introduced in this study to process and transcribe the task expert's verbal inputs during the task, which are captured by a microphone headset.

7.3.4 Implementation and testing of the framework

<u>Step 1: Capture</u>

The Kinect sensor captures the entire task performed by the human and the data produced is processed using standard and bespoke software functions to digitise and record human actions and the effects of those actions on the workpiece components. Sound is also captured in this implementation.

Workpiece recognition

The workpiece components are assembled inside the pre-defined virtual are on the assembly workstation. This virtual area is marked in a white circle outline as shown in Figure 78c. The workpiece tracking algorithm scans this virtual area for presence of surfaces not belonging to the workstation, based the depth patterns and identifies them using blue dots.

To identify the object placed on the workstation as the Ikea table and to track its configuration as it is assembled, 8 specific points within the virtual area are continuously scanned for presence or absence of the components belonging to the table. The location of these points is determined from the known geometry of the assembled workpiece as shown in Figure 79 (a). The four base points (BP1 to BP4) are used to identify the presence of the base on the workstation and leg points (LP1 and LP4) are used to identify the individual legs when they are assembled on the base (Figure 79 (b)).

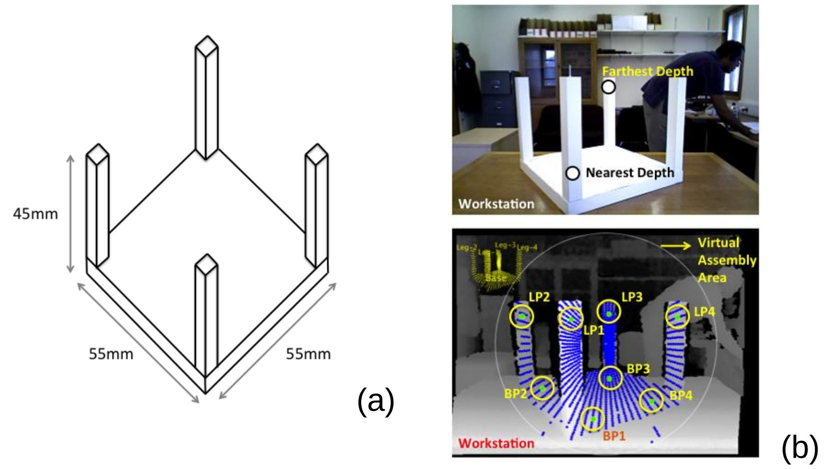


Figure 79: (a) Known geometry of the assembled workpiece (b) Workpiece feature recognition points

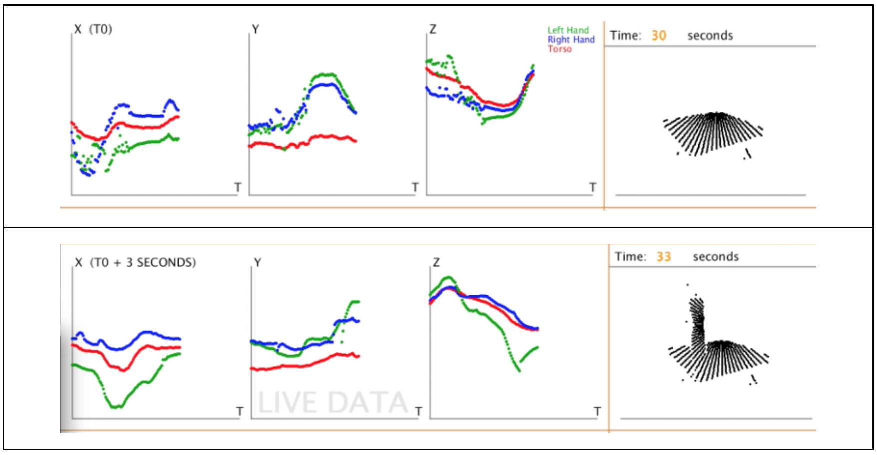
Tracking human-workpiece interactions

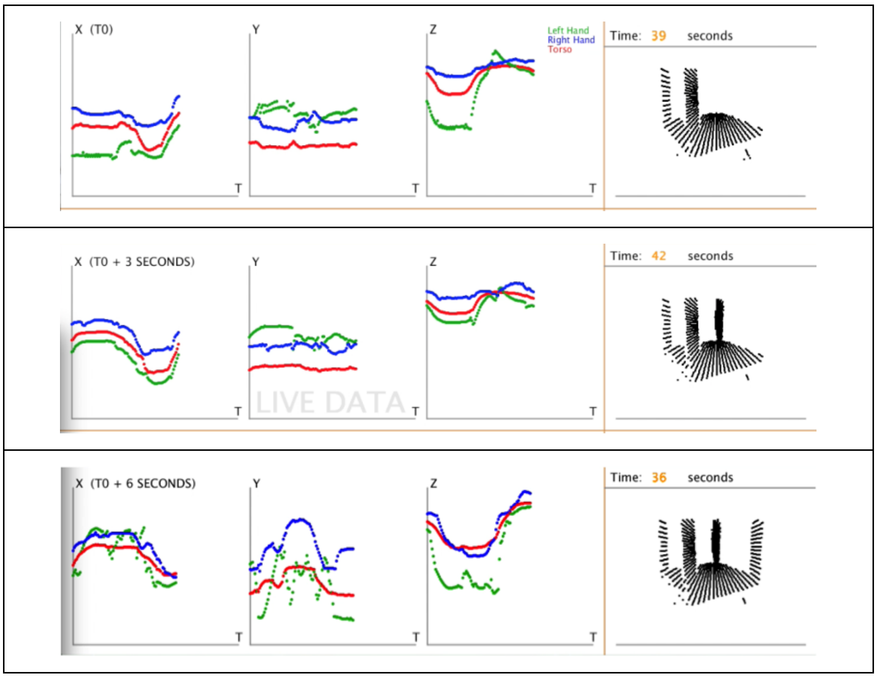
The assembly process begins when the human picks up the workpiece components one by one and places them on the workstation at specific locations in a specific sequence for assembly. The base is placed first followed by the legs in the assembly order of 1 to 4, as marked on the base. Two task scenarios are captured. In the first scenario (scenario A), the assembly sequence is followed and in the second scenario (scenario B), the assembly sequence is not followed resulting in a problem-solving session within the task. Figure 80 shows the captured assembly sequence in scenario A.

Figure 80: The captured assembly sequence in scenario A

This task scenario is continuously tracked by the capture step and the live status of the task is graphically rendered as shown (Table 19). The human motion (of left hand, right hand and torso) is continuously plotted and a workpiece configuration snapshot is taken and displayed every 3 seconds. This way, the human action within every 3-second period is responsible for the workpiece configuration at the end of the 3-second period.

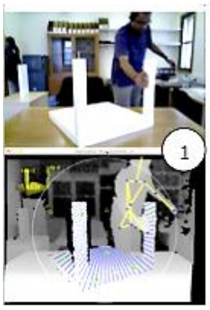
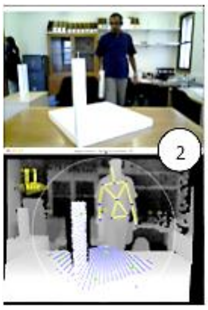
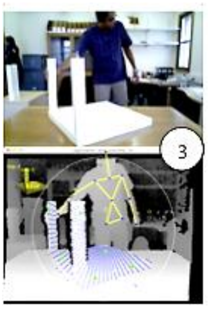
Table 19: Live human action and workpiece progress (scenario A)





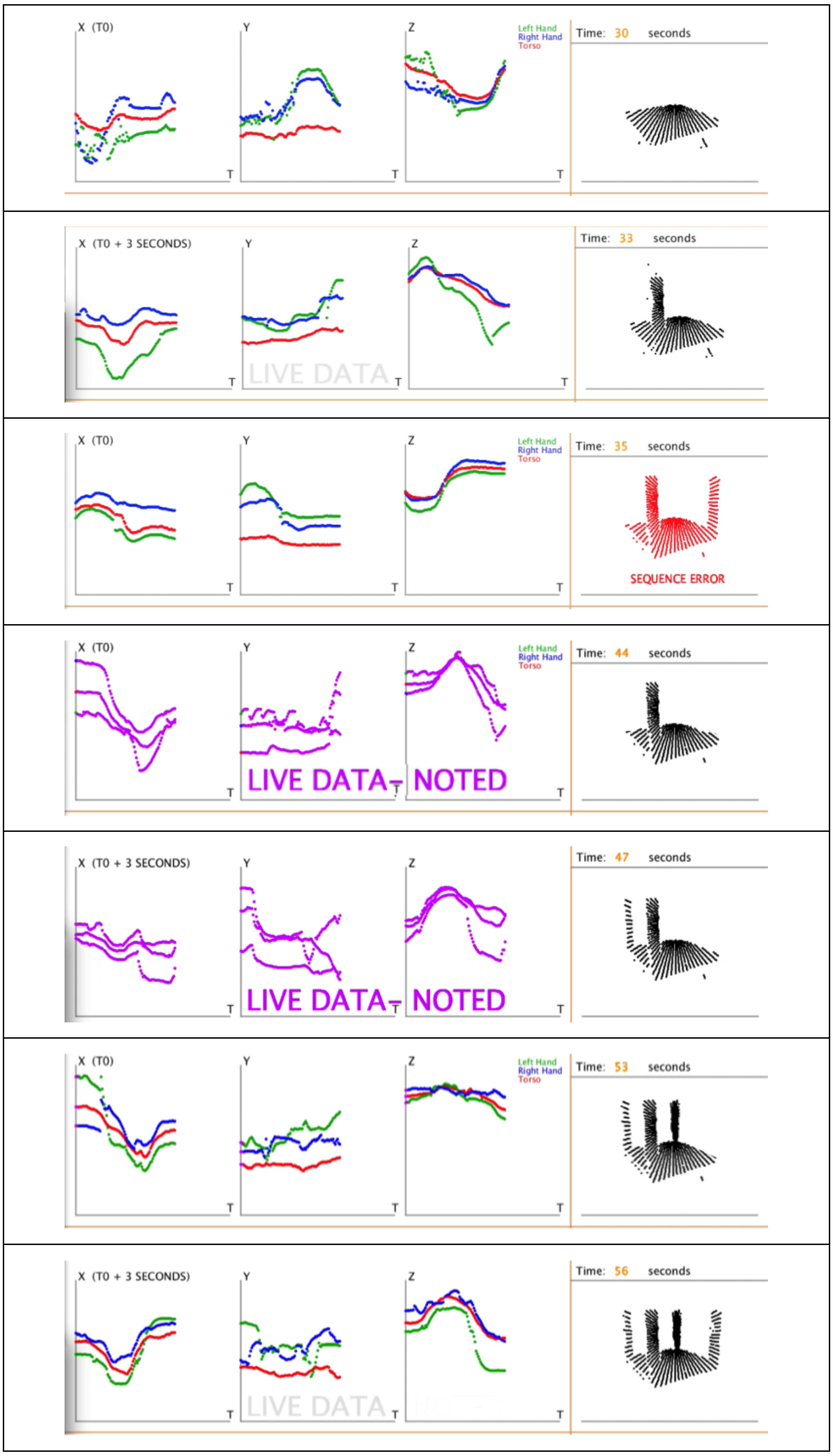
In scenario B, the expert places leg 4 before legs 2 and 3 and therefore triggers a problem alert. Upon activation of this alert, the expert issues a verbal instruction for the capture system to label his next set of actions as problem-solving actions until he issues an instruction again to stop labelling (Table 20).

Table 20: The problem-solving sequence for scenario B

				
The expert places the table leg on location 4 before placing the table legs in locations 2 and 3 triggering a problem alert.	The expert issues a verbal instruction to the capture system to label his next set of actions as problem-solving actions	The notices and expert corrects the problem by removing the table leg from location 4.	The expert now places the table leg in location 2 thereby solving the problem and subsequently following the sequence.	The expert issues a verbal instruction to the capture system to stop labelling his actions as problem-solving actions.

This problem-solving scenario is rendered graphically in Table 21. A ‘sequence error’ alert is raised by changing the colour of the workpiece snapshot from black to red and displaying a ‘sequence error’ message on the capture system screen. Once the verbal instructions to label the expert’s action is given, the human motion plot changes its colour to violet indicating that the motion data is labelled and recorded as a problem-solving session in the spreadsheet.

Table 21: Live human action and workpiece progress (scenario B)



This human-workpiece interaction data is filtered and smoothened before it is stored in a spreadsheet. Snapshots of the spreadsheet for scenarios A and B are presented in Figure 81 and Figure 82 respectively.

Prob_Solv	No.	Timestamp	LH_Hand_X	LH_Hand_Y	LH_Hand_Z	LH_Elbow_X	LH_Elbow_Y	LH_Elbow_Z	LH_Shoulder_X	LH_Shoulder_Y	LH_Shoulder_Z	RH_Hand_X	RH_Hand_Y
N	1	559	372.5598	213.7	2467.306	397.55756	142.43707	2452.2112	440.79703	93.059906	2269.4006	470.80902	145.56625
N	2	560	371.9292	213.7929	2464.585	396.7448	142.5008	2451.525	439.12775	91.851715	2272.7673	483.37848	135.06975
N	3	561	370.8063	215.4547	2468.137	394.70746	143.71832	2448.3477	438.96603	91.17984	2272.2	498.64984	91.77692
N	4	562	370.3335	218.2235	2444.038	390.6541	148.01959	2446.8384	426.8841	91.48331	2299.0376	496.59497	104.10576
N	5	563	371.2278	221.159	2439.386	388.14624	151.32602	2455.879	422.4062	91.55324	2309.657	496.96307	86.74747
N	6	564	365.8514	225.3613	2433.628	383.51187	155.83035	2447.4946	419.47107	91.00511	2312.5588	545.59753	225.41154
N	7	565	361.6148	226.232	2424.891	379.119	156.50618	2447.2075	409.74374	93.55795	2327.7603	523.42163	221.32263
N	8	566	358.1276	228.5142	2428.23	373.98395	156.28381	2448.561	405.3458	93.245026	2330.5225	514.2904	233.73824
N	9	567	357.5886	230.7806	2414.257	370.79413	159.72305	2448.826	400.89926	94.52829	2330.8748	502.26785	231.21411
N	10	568	347.7605	233.0878	2403.531	355.61188	161.59723	2448.2622	377.14413	94.651825	2359.0244	483.4088	228.28275
N	11	569	345.1841	231.7304	2413.876	353.88394	158.32745	2445.1057	377.461	93.477295	2352.6265	474.8857	222.28493
N	12	570	339.2243	224.2387	2407.458	343.45825	156.14836	2464.617	371.73688	95.709305	2354.3958	443.83978	209.70898
N	13	571	336.4395	222.7629	2403.154	337.69003	152.75377	2466.081	367.9642	94.47171	2350.1094	447.36407	216.27971
N	14	572	331.1358	219.7753	2411.147	328.10608	148.67136	2463.0488	363.40384	93.09636	2352.2534	444.8157	215.0115
N	15	573	325.2428	217.4437	2442.26	321.2109	147.88312	2471.1853	353.84583	90.62204	2370.6055	431.02127	209.31804

Figure 81: Snapshot of the spreadsheet for task scenario A

N	15	1233	376.5945	262.4607	2531.851	392.75522	183.5045	2415.284	414.77658	92.58418	2306.9487	594.1868	238.87299
N	16	1234	378.6616	263.4113	2527.831	387.9424	183.6932	2422.139	414.77435	92.547	2310.9365	594.6947	238.30357
N	17	1235	382.9579	258.5673	2523.751	387.59726	178.6729	2420.285	414.27438	92.73833	2310.7173	596.1108	237.05696
N	18	1236	382.4816	260.3348	2524.617	384.14545	180.5099	2426.6372	417.54355	93.53447	2303.893	597.125	236.87053
N	19	1237	380.3824	265.7982	2532.003	388.57086	186.1162	2424.31	418.33435	93.404144	2305.5706	598.5809	239.02843
N	20	1238	374.1679	277.3712	2556.83	384.10904	195.9277	2439.6553	418.8257	93.2744	2301.068	599.6382	239.21614
N	21	1239	369.8818	285.3182	2575.027	387.37018	204.8073	2443.3237	419.51538	92.958374	2297.2666	600.2307	239.0218
Y	1	1241	365.5863	291.6942	2591.438	394.4408	213.8449	2441.3442	419.10464	93.17879	2301.816	599.8428	240.16629
Y	2	1242	364.3796	292.857	2593.358	393.76556	215.0087	2441.6714	418.94647	93.38132	2308.7302	598.6921	239.34991
Y	3	1243	363.9761	293.3684	2592.934	397.204	216.9836	2440.842	418.93912	93.396805	2313.8613	597.3666	239.51442
Y	4	1244	363.3514	293.471	2592.611	396.58832	216.9215	2440.425	419.11908	92.9718	2316.4614	595.4489	239.13338
Y	5	1245	376.7142	240.4955	2512.714	401.55154	171.9946	2395.1494	416.2784	93.117584	2313.5369	592.8807	238.84666
Y	6	1246	371.1643	268.6044	2543.068	396.0567	191.7558	2415.3877	414.71027	93.03665	2312.8743	589.9582	238.7348
Y	7	1247	376.7988	261.9709	2530.939	394.5853	184.0719	2408.5103	414.66345	93.10826	2312.7458	587.5026	238.12082
Y	8	1248	378.3504	262.7024	2523.968	388.61917	183.4172	2414.751	417.34717	93.682465	2306.6177	590.5585	242.09613
Y	9	1249	377.0327	269.5724	2531.853	387.90656	189.1727	2420.2166	416.8694	93.847595	2296.5588	588.5759	242.52513
Y	10	1250	373.7421	278.4126	2550.531	385.97958	197.3132	2429.6045	417.24283	93.86502	2291.7673	587.873	242.19156
Y	11	1251	369.8568	285.6187	2568.253	385.14383	204.2381	2438.8157	417.95712	93.44032	2292.1772	586.2388	242.5183
Y	12	1252	367.4842	289.7632	2579.313	389.00177	209.6256	2440.3953	417.26404	93.58258	2291.198	585.1554	242.55293
Y	13	1253	365.546	291.9122	2587.33	392.332	213.4864	2441.1802	416.85123	93.67616	2289.4846	588.9928	245.77925
Y	14	1254	364.4126	292.9321	2589.734	392.57062	214.6099	2442.5823	416.67584	93.60855	2289.5588	587.1063	245.24547
Y	15	1255	362.9408	292.3911	2588.3	396.24063	215.7385	2441.7993	416.92407	93.38756	2290.6409	584.8609	244.60187
Y	16	1256	361.3836	291.8361	2588.612	394.78958	215.1399	2441.905	417.02185	93.09964	2292.5242	583.9856	245.39366

Figure 82: Snapshot of the spreadsheet for task scenario B

Note that the problem solving ('Prob-Solv') field in the spreadsheet for all data points in scenario A is 'N' which denotes normal actions. For scenario B, the label changes from 'N' to 'Y' when the task expert issues an instruction to note his problem-solving actions. Such data points with can be isolated easily for further analysis.

Step 2 – Segment

The continuous human and workpiece data is segmented into human action states and workpiece states using the time-sampling method and the spatio-temporal dependencies between these states is identified. The spreadsheet containing continuous raw human motion and workpiece tracking data is parsed and the data is segmented every 3 seconds (Figure 83). This interval can be changed to change the granularity of the digitised knowledge desired.

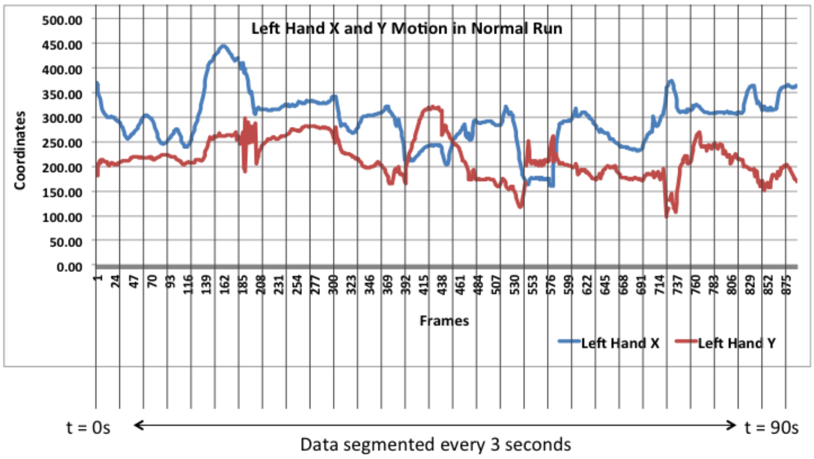


Figure 83: Segmentation of human motion data for task scenario A

In this case, the time unit used is 3 seconds. The segmentation process and resulting states are shown in and Table 22 respectively.

Nomenclature of states

The workpiece states are named according to the following format:

Assembly Started? (Y/N) – Base (B/O) – Leg 1 (L/O) – Leg 2 (L/O) – Leg 3 (L/O) – Leg 4 (L/O)

Note that in the above format, symbol 'O' means 'Not Present'. For example, if no workpiece is present, the workpiece state is $NOOOOO$ and if the base and first 2 legs are present, the workpiece state is $YBLLOO$.

The nomenclature of a human action state is similar but these states can also indicate a change effected in the workpiece state. For example, if only *YBOOOO* occurs during a human action state, that state is named H_YBOOOO and if two workpiece states, for example $YBOOOO$ and $YBLOOO$, occur within a human action state, then that state is named H_YBOOOO_YBLOOO indicating that the human has changed the workpiece configuration in that state.

The human action and workpiece states for task scenarios A and B are illustrated in Table 22 and Table 23 respectively.

Table 22: Human action and workpiece states for normal task scenario A

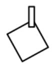
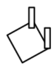
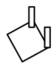
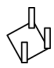
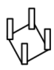
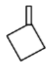
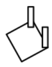
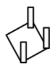
Human action state	Workpiece state
H_NOOOO	 NOOOO
H_NOOOO_YBOOOO	 YBOOOO
H_YBOOOO	 YBOOOO
H_YBOOOO_YBLOOO	 YBLOOO
H_YBLOOO_YBLLOO	 YBLLOO
H_YYBLLOO	 YBLLOO
H_YYBLLOO_YBLLLO	 YBLLLO
H_YYBLLOO_YBLLLL	 YBLLLL
H_YBLLLL	 YBLLLL

Table 23: Human action and workpiece states for the problem solving scenario B

Human action state	Workpiece state
H_NOOOO	 NOOOO
H_NOOOO_YBOOOO	 YBOOOO

H_YBOOOO	 YBOOOO
H_YBOOOO_YBLOOO	 YBLOOO
H_YBLOOO_YBLOOL (Sequence error)	 YBLOOL
H_YYBLOOL (Sequence error)	 YBLOOL
H_YYBLOOL_YBLOOO	 YBLOOO
H_YYBLOOO_YBLLOO	 YBLLOO
H_YYBLLOO_YBLLLO	 YBLLLO
H_YYBLLOO_YBLLLL	 YBLLLL
H_YBLLLL	 YBLLLL

Isolation of problem-solving session from within the task

Another method of isolating problem-solving sessions from the continuous task data is by comparing the human motion charts of the same task performed in multiple scenarios and identifying the segments where the charts differ (Figure 84). The problem solving sessions can be separately modelled and the knowledge embedded within them can be extracted and decoded.

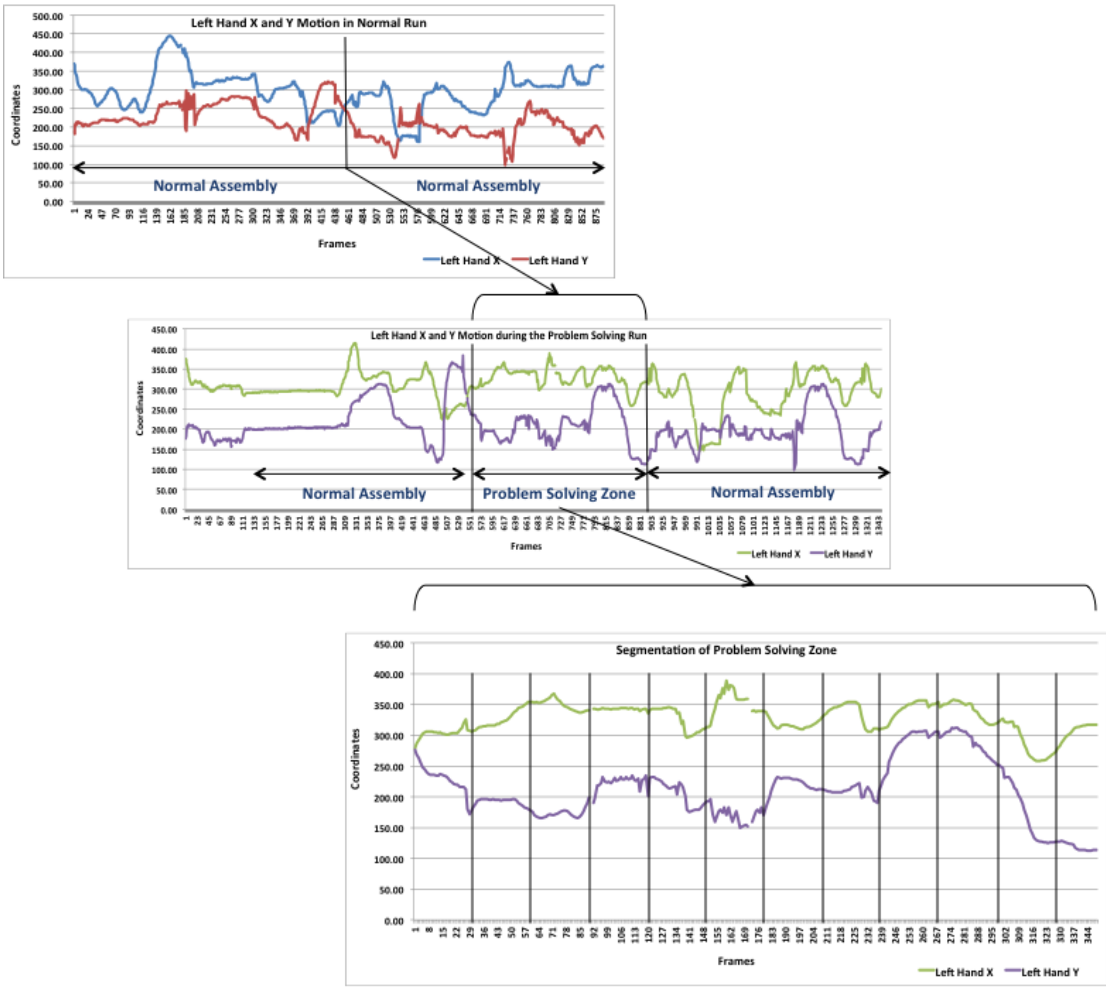


Figure 84: Identification, isolation and segmentation of problem-solving session from a captured task scenario

<u>Step 3: Model</u>

The discrete workpiece and human action states generated in the segmentation step are now ready to be modelled using HMM.

HMM generation

The topology of the HMM model depends on the number of human action and workpiece states generated and the parameters of the HMM model are estimated by the task expert. Since the time-sampling segmentation method is used, different set of human action states can be generated for the same task scenario. This is because the time taken for each action within the task scenario can vary from one capture run to another.

4 unique HMM models ($\lambda_A, \lambda_B, \lambda_C$ and λ_D) are generated, out of which 3 models ($\lambda_A, \lambda_B, \lambda_C$) belong to the normal task scenario from which 3 different human action states were captured and 1 model (λ_D) belongs to the unique problem-solving session that was captured once.

All the HMM models are optimised using the Baum Welch algorithm and the optimised models used for knowledge extraction and decoding.

HMM evaluation – picking the right model

Consider a task scenario represented by the workpiece observation sequence

$O_{Q1} = \{NOOOOO,\ YBOOOO,\ YBLOOO,\ YBLLOO,\ YBLLLO,\ YBLLLL\}$

Using the 'Forward' algorithm, the following model probabilities are computed:

$P(\lambda_A \mid O_{Q1}) = 2.68E-5$

$P(\lambda_B \mid O_{Q1}) = 7.68E-6$

$P(\lambda_C \mid O_{Q1}) = 3.28E-6$

$P(\lambda_D \mid O_{Q1}) = 4.70E-7$

Since $P\,(\lambda_A \mid O_{Q1})$ is the highest probability HMM model λ_A is picked for the given observation sequence O_{Q1}.

Similarly, consider the problem-solving scenario represented by the workpiece observation sequence:

$O_{Q2} = \{YBLOOL,\ YBLOOO,\ YBLLOO,\ YBLLLO,\ YBLLLO,\ YBLLLL\}$

Using the 'Forward' algorithm, the four probabilities are computed for the new observation sequence O_{Q2}

$P(\lambda_A \mid O_{Q2}) = 4.15E-10$

$P(\lambda_B \mid O_{Q2}) = 2.43E-10$

$P(\lambda_C \mid O_{Q2}) = 8.77E-9$

$P(\lambda_D \mid O_{Q2}) = 1.48E-7$

Since $P\ (\lambda_D \mid O_{Q2})$ is the highest probability HMM model λ_D is picked for the given observation sequence O_{Q2}.

Step 4: Extract

Once the most probable HMM model is identified for the given task scenario, the sequence of human action states responsible for that scenario can be extracted using the Viterbi algorithm. Once the human action sequence is extracted, detailed analysis can be performed on the individual states that make up the sequence to decode the knowledge embedded within them.

Consider the observation sequence O_{Q2} for which λ_D is identified as the most likely model that represents O_{Q2}. Using the 'Viterbi' algorithm, the most likely sequence of human actions H_{Q2} that could produce O_{Q2} is identified.

Therefore, $H_{Q2} = \{H\_YBLOOL\_YBLOOO,\ H\_YBLOOO,\ H\_YBLOOO\_YBLLOO,$ $H\_YBLLOO,\ H\_YBLLOO\_YBLLLO,\ H\_YBLLLO\_YBLLLL\}$

The manufacturing knowledge embedded within this action sequence H_{Q2} will be decoded in the next step.

Step 5: Decode

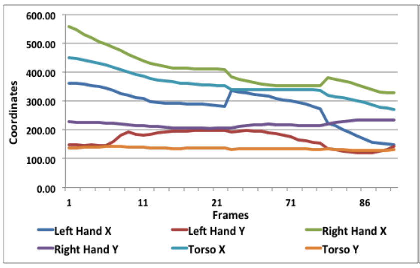
It has been demonstrated so far that human response in a given task scenario can be extracted from the HMM model that represents that scenario. The task scenario may or may not have been explicitly observed while capturing the task. From the human response, how exactly did the human manipulate the workpiece during the scenario can be obtained from the raw human-workpiece interaction data that is stored in spreadsheets. The link to the relevant data in the spreadsheet is provided by the model.

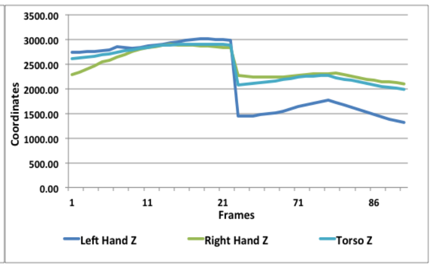
Consider the human action states H_{Q2} extracted earlier for the given task scenario represented by the workpiece observation sequence O_{Q2}. Before, any manufacturing knowledge constituents can be decoded from H_{Q2}, the human action states contained within it must be extracted along with the raw human-workpiece interaction data.

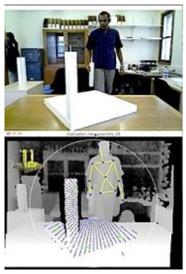
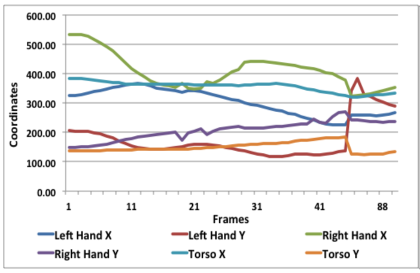
States in H_{Q2} are: ***1)*** *H_YBLOOL_YBLOOO*, ***2)*** *H_YBLOOO*, ***3)*** *H_YBLOOO_YBLLOO*, ***4)*** *H_YBLLOO*, ***5)*** *H_YBLLOO_YBLLLO*, ***6)*** *H_YBLLLO_YBLLLL*

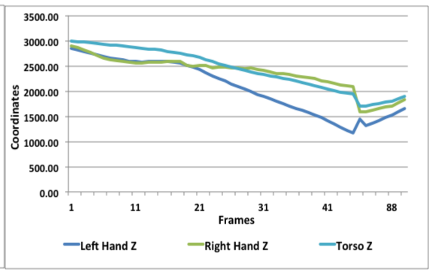
These human action states are mapped to the corresponding workpiece states from the observation sequence and the human-workpiece interaction data associated with the states are tabulated in Table 24.

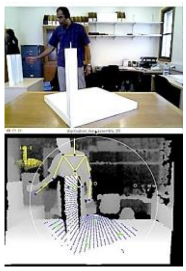
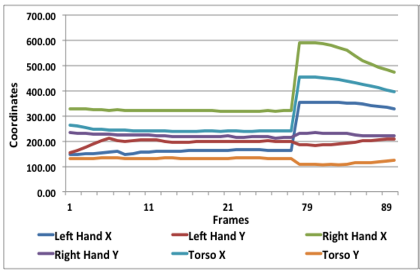
Table 24: Extracted human action mapped to corresponding workpiece states.

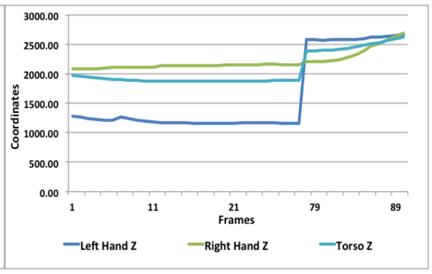
<table>
<tr><th>Human action state</th><th>Workpiece state and task status</th></tr>
<tr><td>


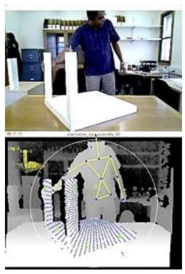
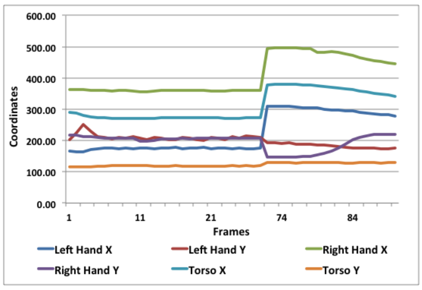


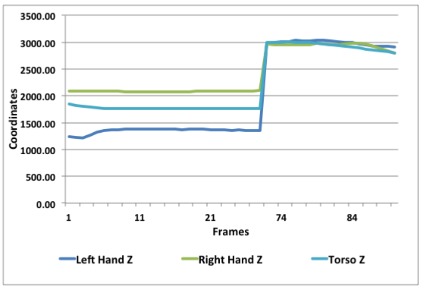
H_YBLOOL_YBLOOO</td><td></td></tr>
<tr><td>


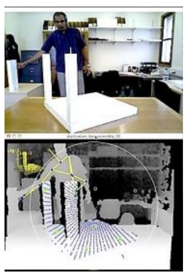


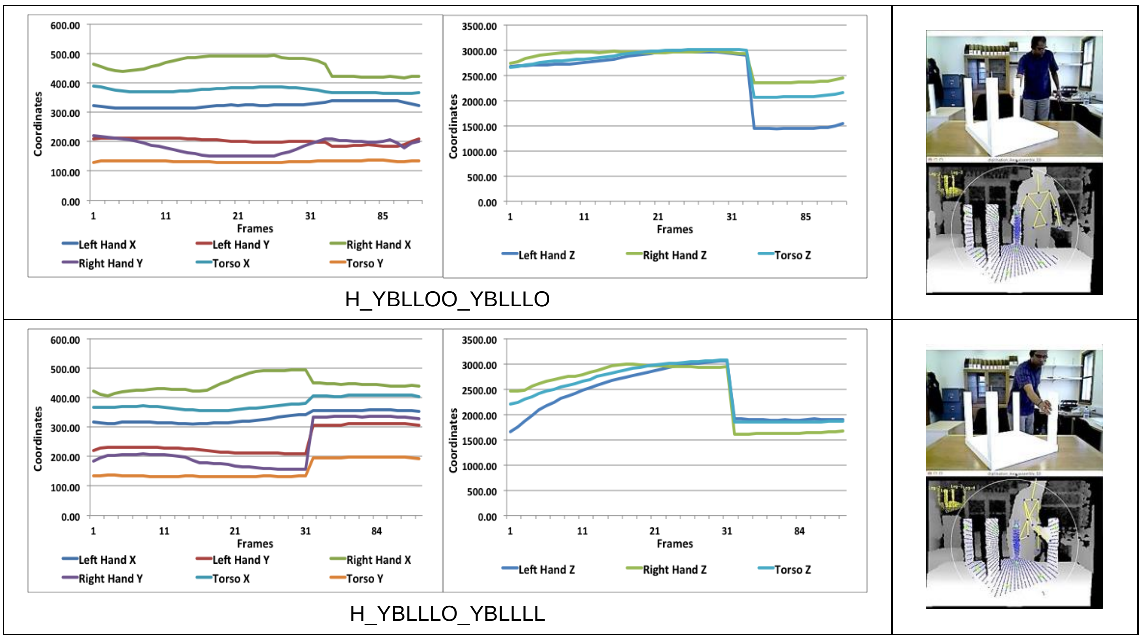
H_YBLOOO</td><td></td></tr>
<tr><td>




H_YBLOOO_YBLLOO</td><td></td></tr>
<tr><td>




H_YBLLOO</td><td></td></tr>
</table>



From the above human action data, several constituents of manufacturing knowledge such as the task strategy adopted, nature and spatial characteristics of gestures made, mechanics of motion performed during the gestures, and workpiece manipulation techniques can be decoded.

The knowledge constituents decoded in this study are:

- Task execution strategy
- Nature and spatial characteristics of human gestures during the task
- Workpiece manipulation techniques
- Mechanics of human motion such as head bending angle (glance angle), body bending angle, angles between upper and lower arms, gesture speeds, body orientation with respect to the workpiece, etc.

<u>Task execution strategy</u>

The task execution strategy can be decoded by determining the sequence of human actions responsible for the given sequence of workpiece states from the most likely HMM model. From the human action sequence, the following knowledge can be mined:

a. Planning and approach of task breakdown into sub-tasks as seen in Table 24 where each action can be considered a sub-task.

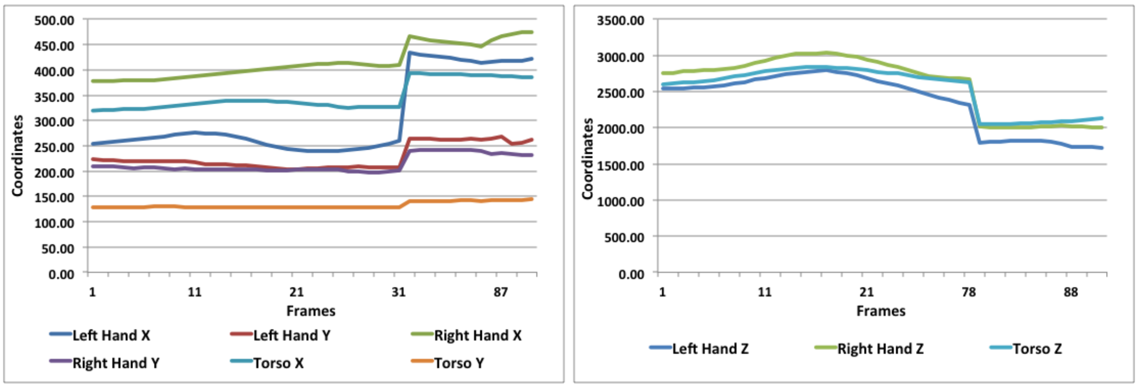
b. Execution sequence of sub-tasks to achieve the main task. This sequence changes if a different task scenario were to be considered.

c. Selections made during the task to choose specific actions from a choice of actions available to successfully complete the task. In this example, during the problem-solving scenario where a leg was wrongly placed, the human either had the choice to remove that leg and place it in the right position, or to place legs at the locations that he had missed before making the sequence error.

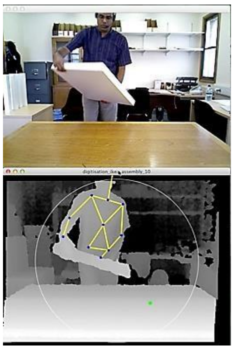
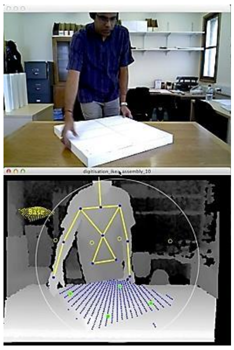
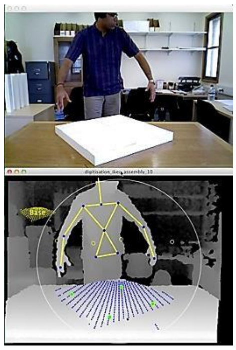
<u>Nature and spatial characteristics of human gestures</u>

By visualising the human motion data within each extracted action state, the nature (trajectories and patterns) and spatial characteristics (3D coordinates) of human gestures with respect to the changes in the workpiece can be obtained. The human motion data and corresponding workpiece change data for the extracted state *H_NOOOOO_YBOOOO* is shown in Table 25.

Table 25: Human gestures during an action state



H_NOOOOO_YBOOOO

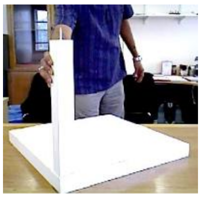
			
1	2	3	4

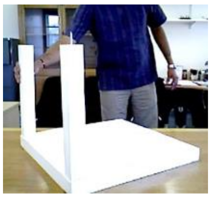
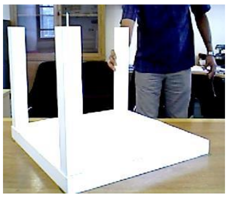
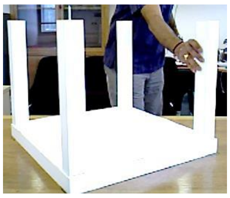
Workpiece manipulation techniques

Component grasping is an important constituent of manufacturing knowledge in assembly tasks because improper grasping can result in accidentally dropping the component or making additional manoeuvres to orient it correctly for successfully assembly. Grasping techniques used by skilled experts could be extracted and annotated with the corresponding human actions states in which component grasping occurs so that these techniques are also transferred when skill training is conducted.

In this study, the workpiece grasping techniques that occur in different human action states are extracted (Table 26).

Table 26: Workpiece component grasping techniques

Human action state	Grasping technique
H_NOOOOO_YBOOOO	
H_YBOOOO_YBLOOO	

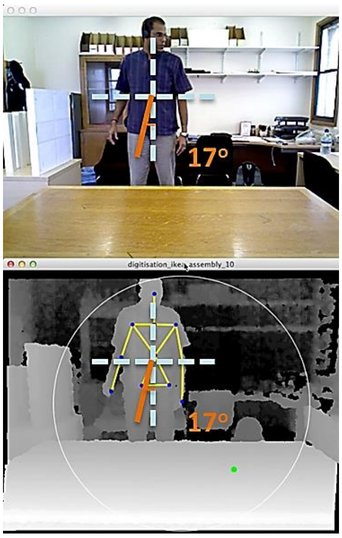
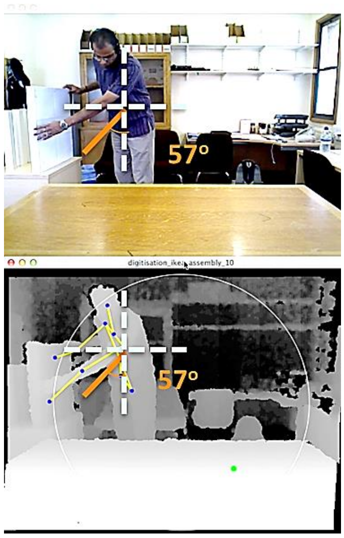
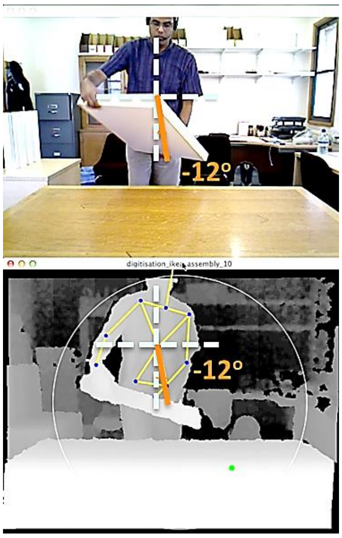
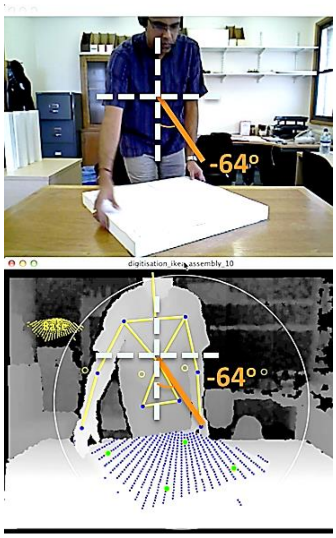
H_YBLOOO_YBLLOO	
H_YBLLOO_YBLLLO	
H_YBLLLO_YBLLLL	

Mechanics of human motion

Dexterity is an important trait that humans possess and can use this trait effectively to perform complex movements, such as those required in manual manufacturing tasks. Humans who are experienced at performing a task can routinely perform complex gestures because the movements involved within the gestures become embedded in muscle memory due to repetitive practice. Dexterity manifests itself in the mechanics of human motion such as in body postures, nature and spatial characteristics of gestures, workpiece manipulation techniques, gesture speeds, glancing angles, body orientations with respect to the workpiece, etc.

Table 27 shows the changes in body orientation with respect to the y-axis while manipulating the workpiece in human action state H_NOOOOO_YBOOOO.

Table 27: Body orientation computed and illustrated over actual task images

<u>Step 5: Reproduce</u>

2D animation is used as a medium to reproduce the human-workpiece interactions and augment this animation with manufacturing knowledge extracted and decoded in this framework. An example of such an animation (without the annotations) is shown in the form of static snapshots in Figure 85.

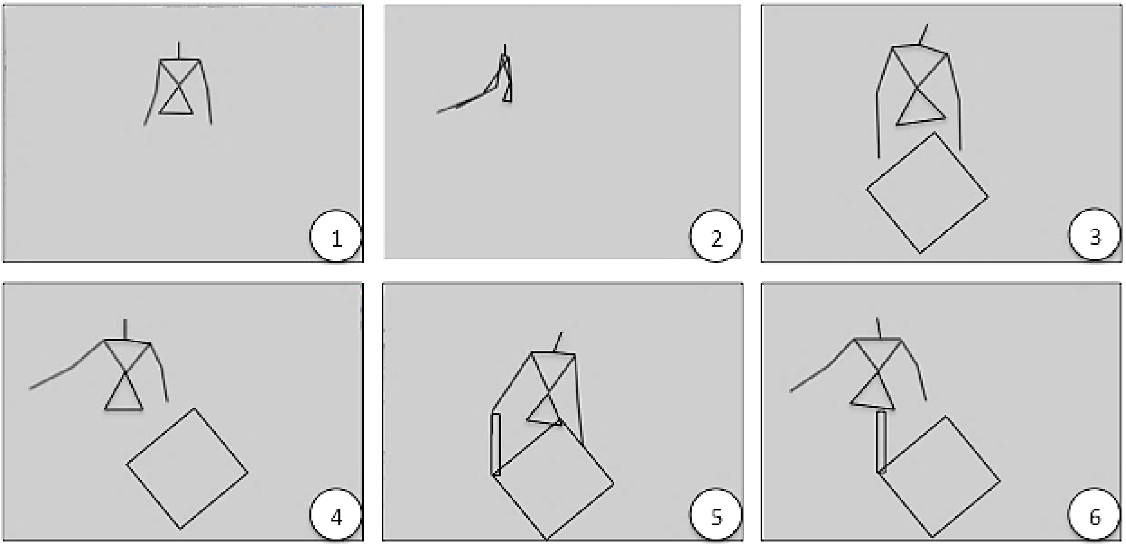


Figure 85: Animation of a specific task scenario

7.3.5 Task performed by another human

So far, the proposed framework was used to digitise the tasks performed by the same human, the author. In order to demonstrate that the framework is independent of the human performing the tasks, the Ikea table assembly task was performed by another human (henceforth called ‘Operator 2’) who is not involved in this research. The 6-step framework was used to digitise this task with no changes made to any of its components. The same task capture setup was used as before (Figure 78). Operator 2 was shown the individual components of the workpiece and the final assembled workpiece but he was not trained or briefed on how to perform the actual assembly. This was done so that the skills naturally used by operator 2 to perform the task could be digitised.

Operator 2 performed the assembly task 5 times and each time the task was captured using the methods prescribed in the framework. The human action and workpiece states were then generated using the time-sampling technique and then modelled using hidden Markov modelling to produce an HMM model that represents the second operator’s execution of the task. This model is used to extract the operator’s action states for any given task scenarios. The constituents of manufacturing knowledge extracted from the author’s execution of the task are extracted and decoded from the second operator’s task too.

Results

In this section, only the extraction and decoding of the manufacturing knowledge constituents are presented since the implementation of the framework is already described in the earlier sections of this study.

<u>Task execution strategy:</u> It is observed that the task execution strategy used by operator 2 is different from that of the author. The sequence in which operator 2 assembled the workpiece was extracted from the HMM model generated for operator 2 as

$$H_{O2} = \{H\_NOOOO\_YBOOOO,\ H\_YBOOOO\_YBOLOO,\ H\_YBOLOO,\ H\_YBOLOO\_YBOLLO,\ H\_YBOLLO\_YBOLLL,\ H\_YBOLLL\_YBLLLL\}$$

Note that the same nomenclature for human action and workpiece states is used as before. The assembly sequence check was disable for operator 2 so that a different task strategy could be observed. Operator 2 assembled the base of the table first followed by the table legs in the order 2 -> 3 -> 4 -> 1 whereas the author assembled the legs in the order 1 -> 2 -> 3 -> 4. Though this is fairly trivial knowledge which could also be visually observed, the framework enables automatic extraction of this knowledge which is very useful for complex tasks where visual observation and task segmentation is not as easy.

<u>Human gestures and grasping techniques:</u> The gestures performed by operator 2 during the task also varied from those of the author. This could be seen from the differing hand motion charts for the action states that are common for the two operators. An example of such an action state in which this difference is clearly demonstrated is shown in Figure 86.

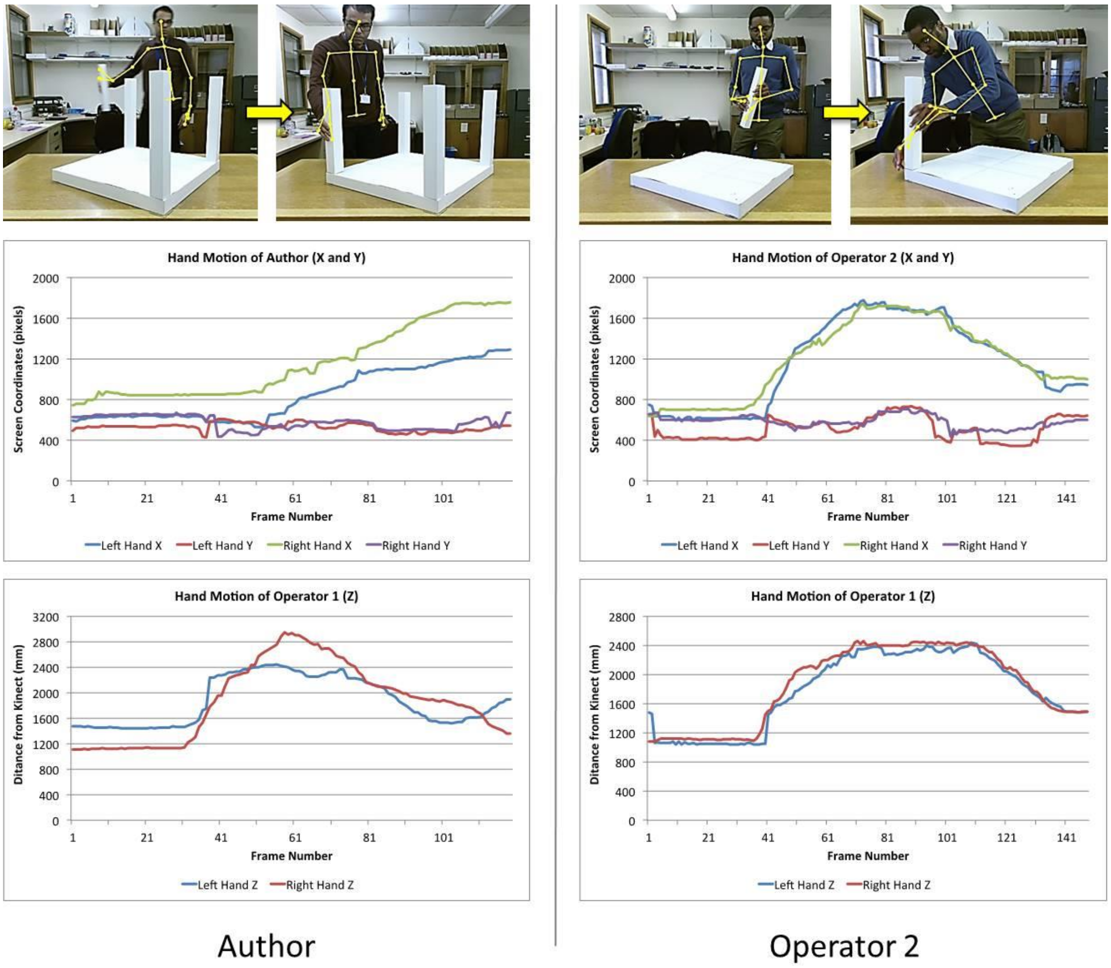


Figure 86: Differences in hand motion during the task

From these charts and images it can be seen that the author while assembling the leg component grasps it with his right hand and transfers it to his left hand before placing it on the table base, while always keeping his hands separated whereas operator 2 grasps the leg component with both his hands and places the leg on the base never separating the two hands during the action state.

<u>Mechanics of human motion:</u> The mechanics of movements of the two operators is also different and this can be compared because the digitisation framework is able to extract and decode it. An example of this is shown in Figure 87 where the mechanics of the two operators while placing the table base is computed and displayed. The difference in mechanics between the author and operator 2 represents their different skills levels used while

performing the assembly task. The images also show the different ways in which the two operators grasp the base for placement.

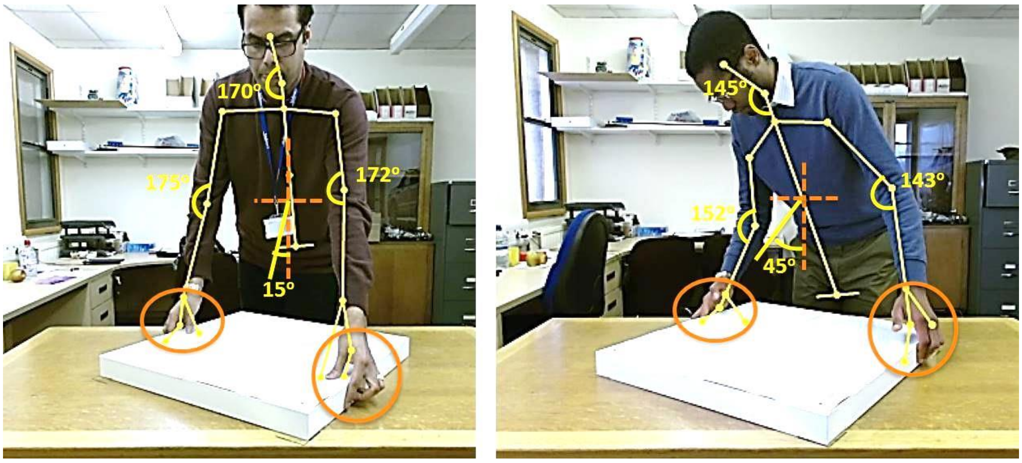


Figure 87: Difference in motion mechanics while performing assembly

It is therefore demonstrated that the 6-step digitisation framework is successfully implemented and tested to digitise the manual manufacturing task knowledge embedded within the Ikea table assembly task irrespective of who performs the task. It also demonstrates the ability of the framework to digitise the skills of two different humans which can be useful in applications where skill comparison is needed, such as when evaluating the competency of novice workers before and after skill training.

7.4 Case Study 3: Composite Layup Task

This case study concludes the validation of this research by implementing the proposed 6-step framework to digitise the manufacturing knowledge embedded within a real-world manufacturing task. The task chosen is the manual layup of pre-impregnated composite plies over a metal mould with complex surface geometry. Composite layup is an important process in the manufacturing industry and has not been fully automated because of the complexity involved and the difficult nature of the composite material itself. However, because of high production costs, low production speeds and inconsistent quality, manual layups are prime targets for automation. Also, because of the increasing scarcity of skilled layup technicians, effective methods of skill transfer/training are urgently sought. This research aims to capture, extract and decode the tacit constituents of manufacturing knowledge that essentially are a significant part of human layup skills to enable effective skill transfer from people to people and people to machines. A successful implementation of the framework to achieve this aim and a benchmarking of this research with similar work done by another researcher group are key steps needed to validate this research and are presented in this chapter.

The two main objectives of this case study are

i. To demonstrate the feasibility of the proposed digitisation framework to capture, extract and decode manufacturing knowledge, especially tacit knowledge such as human skill, from a real-world manufacturing task.
ii. To compare the proposed digitisation method and resulting outcomes with those of another reported method used to extract manufacturing knowledge from the same task.

Most of the work in this case study is presented in a journal paper (See 'List of Publications').

7.4.1 Choice of task

A real-life manufacturing task is chosen for the first time in this research at a stage when the proposed digitisation framework has been successfully implemented and tested for 4 tasks of varying complexity so far. Therefore, all the 6-steps of the framework have developed enough to be tested with a real-life case study.

The manual composite layup task is chosen primarily because it exemplifies a vast set of manual tasks used in the manufacturing industry which are more art than science. Their dependency on human skill is so heavy that despite the market-driven push to lower costs, increase production speeds and provide consistent quality, it hasn't been possible to fully automate them. Secondly, composite layup is growing in significance and use in the manufacturing industry across different sectors due to high performance potential of the composite materials. An example of this is the growing percentage of composite material in the aircraft structures used in commercial airplanes of today.

Therefore, there is a need to standardise the manual layup process to reduce costs and the discrepancies between parts caused by human variation. Alternatively, Automated Fibre Placement (AFP) techniques have been developed and used by the composite manufacturing industry. However, these techniques have not advanced enough to be able to completely replace human skills that are required to manipulate the hard-to-work-with pre-impregnated composite plies into complex mould shapes leaving no air pockets, wrinkles or distorted surfaces.

7.4.2 Related work in literature

Given the growing importance and value of manual layup processes in composite manufacturing, it is surprising that there is little documentation about the best practices of manual layup of pre-impregnated woven materials in literature. Researchers have suggested that this may be because of the dearth of holistic methods to observe, capture, extract and decode the human skills involved in manual layup tasks, which according to them is a necessary precursor for documenting the best practices, standardising the processes and eventually automating them. There are a few attempts being reported in literature to understand the manual layup process. Bekey et al. (1993) have used heuristics from human motor skills observations to develop a knowledge-based control tool for robot hand grasping actions, Buckingham and Newell (1996) have used the explicit understanding of laminators actions while picking and placing large plies of pre-impregnated material to design automated pick and place systems, Skordos et al. (2005) have proposed a technique to optimise the layup process of woven composite material but with no evidence of direct observation of manual layup being performed. Kikuchi et al. (2013) come close to reporting a full technique to observe a manual layup process to extract relationships between human skill and material properties involved in the process. Researchers at the University of Bristol’s Advanced Composite Centre for Innovation and Science (ACCIS) believe that for standardising and improving manual composite layup processes, a full understanding of the processes facilitated by direct visual observation and analysis is necessary.

According to them, there is a big gap in knowledge to clearly understand the skills used by experienced layup technicians to manipulate pre-impregnated composite material by hand into the complex shapes of the moulds. This deduction agrees with the literature review presented in this thesis by the author that also reports the lack of holistic methods to capture and digitise tacit manufacturing knowledge from manual manufacturing.

Studies at the ACCIS have attempted to bridge this gap by visually observing and examining each step of the layup process performed by skilled and experienced technicians on different moulds. From these examinations, a list of hand layup techniques used by the technicians were extracted and documented to understand how, where and why these techniques were used in relation to the mould geometry. The 3 main objectives of that study were to improve the current layup process, improve the methods used to transfer layup skills from one technician to another and lay the foundations for automating complex layup tasks in the future.

7.4.3 Final validation problem

In this final case study, the proposed digitisation framework is used to capture, extract, decode and reproduce the tacit manufacturing knowledge embedded within a manual composite layup task. Each of the 6 steps of the framework are implemented and the outcomes of these steps are reported.

The validation study was conducted at the premises of the ACCIS at the University of Bristol. The task involved hand layup of the pre-impregnated carbon and glass woven material (prepreg) onto a metallic mould. The mould consisted of surfaces with varying orientations and recesses in order to introduce layup complexities in the task that only technicians skilled in the art could handle successfully. Two researchers (known in the context of this study as technicians T1 and T2 to maintain anonymity) who are actively involved in composite layup research at the ACCIS and who are sufficiently trained and skilled in the manual layup process performed the task 3 times each. These performances were captured and analysed using the proposed digitisation framework and the human skills brought into the task by the two technicians were extracted, decoded, compared and reproduced as per the framework. Finally, in order to benchmark this research, the methods and the resulting outcomes of the proposed digitisation framework and those of the research conducted by the ACCIS researchers themselves in understanding the manual layup process are compared to discuss the similarities, differences and identify the complementary aspects of the two studies.

7.4.4 Experimental method

The experiment was setup to implement the step 1 (the Capture step) of the proposed digitisation framework. The setup (Figure 88) consisted of a worktable on which the metallic mould is placed and secured. 2 technicians, trained and skilled in the composite layup task for the chosen mould, performed the layup process for 3 times each thereby resulting in 6 different observations. 2 Kinect sensors were used in this study, one to track the technician's actions and the second one to track the real-time changes to the workpiece, which in this case is the mould with the pre-impregnated ply laid on top of it. The reason why two Kinect sensors were used instead of just one is to take the advantage of superior skeletal motion tracking of the second generation Kinect (Kinect V2) for human action tracking and the easier correlation and processing of the depth and colour image streams of the first generation Kinect (Kinect V1) for workpiece progress tracking. More details on comparison between the Kinect V1 and V2 sensors is provided later in section 8.6.3.

Figure 88: Experiment setup for the validation study

The Kinect sensors were mounted on a tripod at a height of 1.2m from the floor and at a distance of 0.7m from the workpiece. While the Kinect V1 was tilted to accommodate the workpiece in its field of view, the Kinect V2 was not. Two different resin pre-impregnated composite materials were tested in the reconnaissance stage, namely the carbon fibre woven and plain glass woven prepregs. The plain glass woven prepreg was eventually chosen because its surface rendered better than the carbon fibre woven prepreg on the infrared camera image of the Kinect V1. The experiments were conducted in accordance to the University of Bristol's policy for experiments involving human participants. All the six task observations were conducted at 20^{o}C in clean room conditions. The technicians used a tool called the 'dibber' to assist during layup.

Human action tracking

The human skeletal joint tracking functions within the Kinect V2 library developed by Lengeling (2014) were used in the software code that was written in the java-based 'Processing' development platform. The 12 skeletal joints belonging to the upper half of the technician's body are tracked (Figure 89) at the rate of up to 20 frames per second and recorded in a spreadsheet along with the tracking timestamps in seconds.

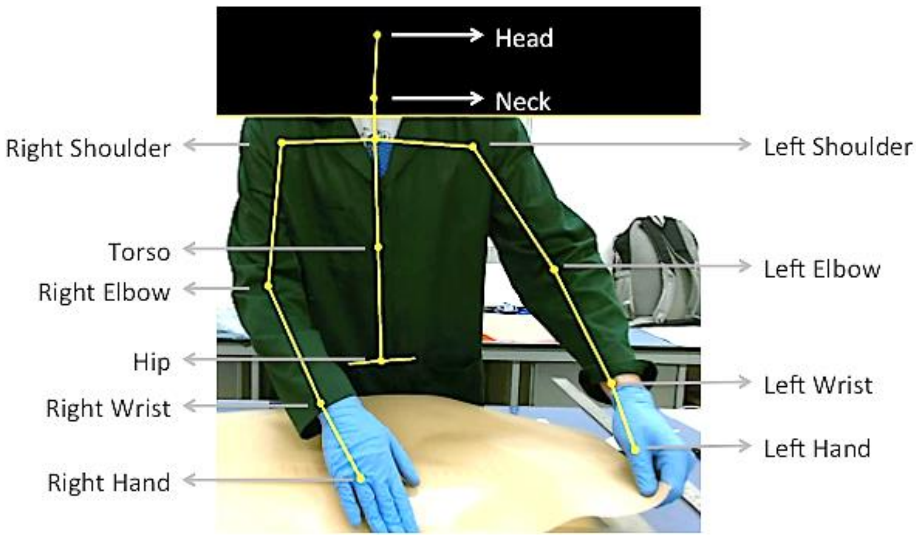


Figure 89: Upper body skeletal tracking joints

Workpiece progress tracking

A different method to track workpiece progress is used in this validation study because of the nature of the task. The layup task starts with the workpiece being in its initial state in which the prepreg is placed over the mould with the ply not conforming to the mould surfaces below. The technician then presses the ply all over the mould in a specific pattern in order to make it conform to the surface of the mould underneath thus forming the final shape of the composite part. Therefore, the surface of the ply changes from non-conforming to conforming to the surfaces of the mould as the layup progresses (Figure 90).

Figure 90: Workpiece progress from blank mould to fully laid prepreg

The surface area of the ply is divided into 7 distinct sectors (Figure 91). The technician lays up the ply on the mould one sector at a time and covers the entire mould in a sequence of sectors according to a technician-specific task strategy.

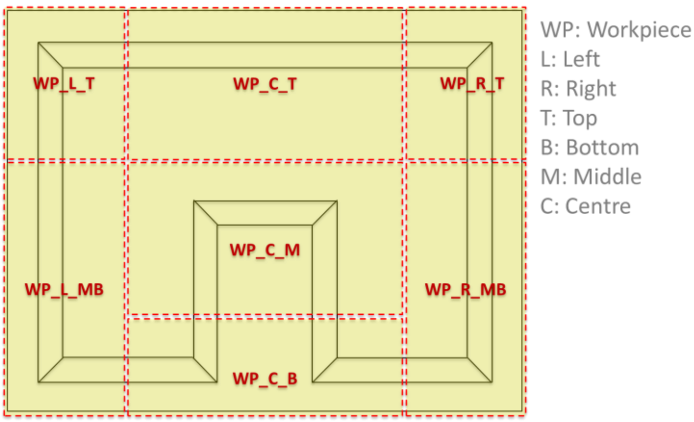


Figure 91: Workpiece divided into 7 layup sectors

The conversion of the ply from non-conforming to conforming can be captured by dividing its surface into finite elements and tracking the orientation of these elements in real-time. The surfaces of the ply whose elements have the same orientation as that of the contours of the mould surface underneath are considered to be fully conformed and laid up. The areas tracked are grouped into the seven sectors as mentioned above.

3-Step method to determine surface orientation of the ply

Step 1: The surface of the ply is divided into a finite number of triangular planes (elements). The number of elements can be increased or decreased by varying the resolution factor. Higher the resolution, higher is the granularity and better is the accuracy of determining surface orientation.

Step 2: For each element, the cross product of the vectors that represent any of the two sides of the triangle is computed and the resulting vector is the surface normal of the element (Figure 92).

Step 3: The surface normals are grouped and displayed in different colours depending on their orientation with respect to the unit vectors along x, y and z axis. This way, the surface contour of the ply can be visualised as either being conformed to the mould surfaces underneath or not.

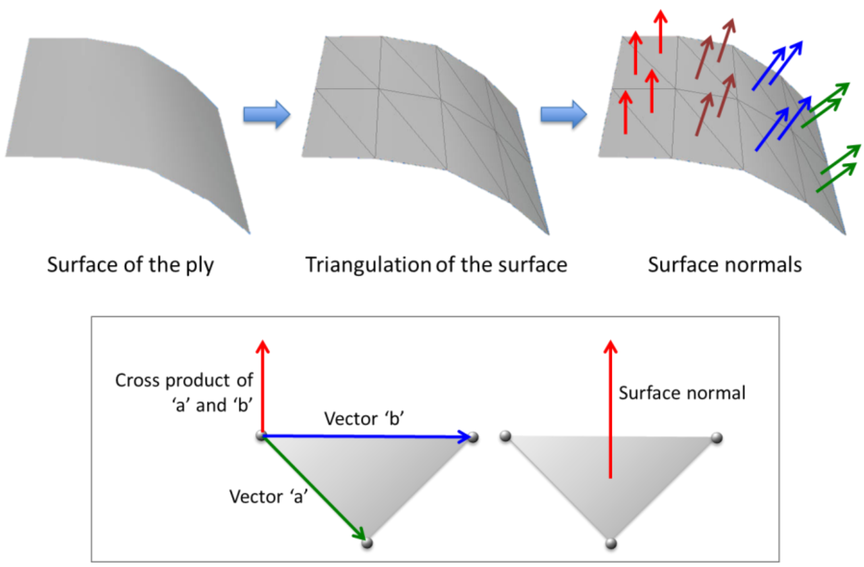


Figure 92: Method to determine ply surface contours

In this study, the surface normals are grouped into 4 different colours, each representing a differently orientated surface as shown in Figure 93.

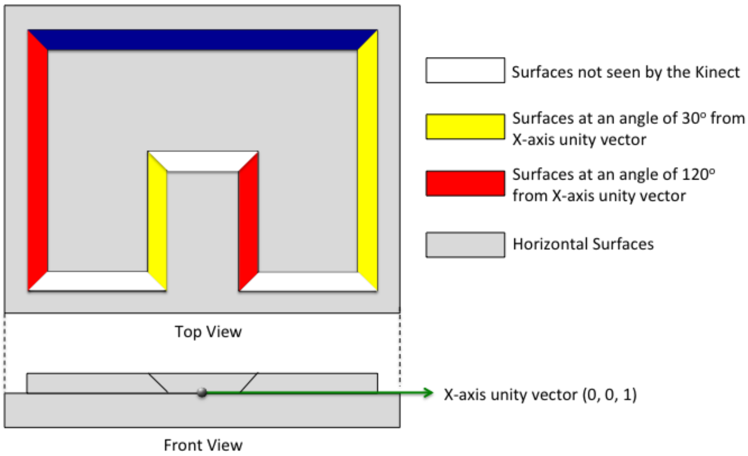


Figure 93: Different surface orientations of the mould

By applying the 3-step method as described above on the prepreg ply continuously as the layup task progresses, the changes happening to the workpiece as the technician manipulates the ply can be tracked. In Figure 94, the surface normals of the bare mould, the prepreg ply surface before layup and the same surface after layup is shown. It can be noted that the surface orientations of the finished workpiece should be the same as those of the bare mould implying that the ply has fully conformed to the mould.

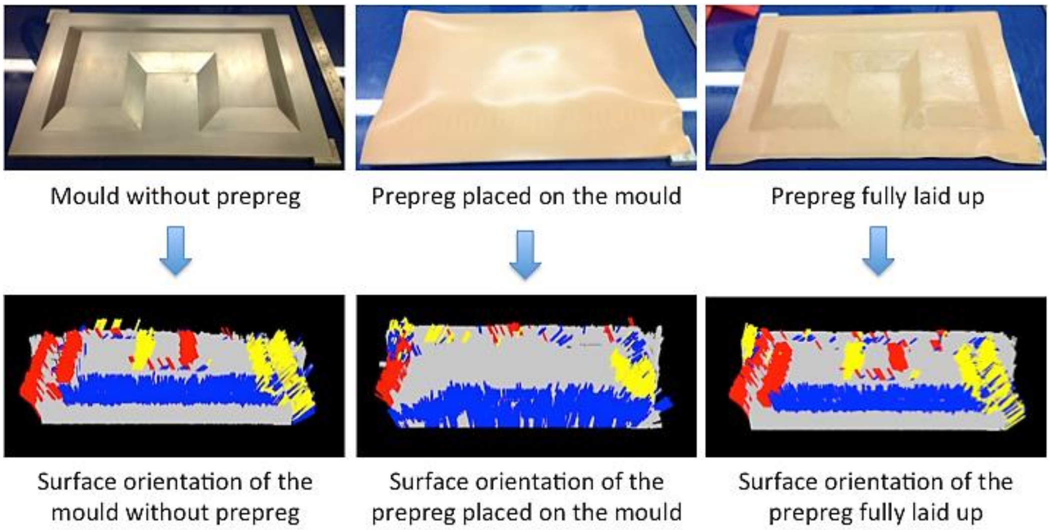


Figure 94: Surface orientation of the workpiece being tracked

It should be noted that the above method is useful to visually track the progress of composite laying on the mould and is not accurate enough to measure the actual orientation in absolute angles of the ply surface.

7.4.5 Implementation and testing of the digitisation framework

The digitisation of the manufacturing knowledge involved in the composite layup task is achieved by implementing the 6-step framework as described below.

Step 1: Capture

Human action capture and workpiece progress tracking are performed simultaneously so that both have common timestamps. Hence, each human action can be associated with the corresponding changes to the workpiece at any given time during the task. The interface windows in Figure 95 show the human action and workpiece progress being recorded by the capture step and the resulting spreadsheet in which the task data is stored is shown in Figure 96. 6 task observations are captured: 3 each by the 2 technicians.

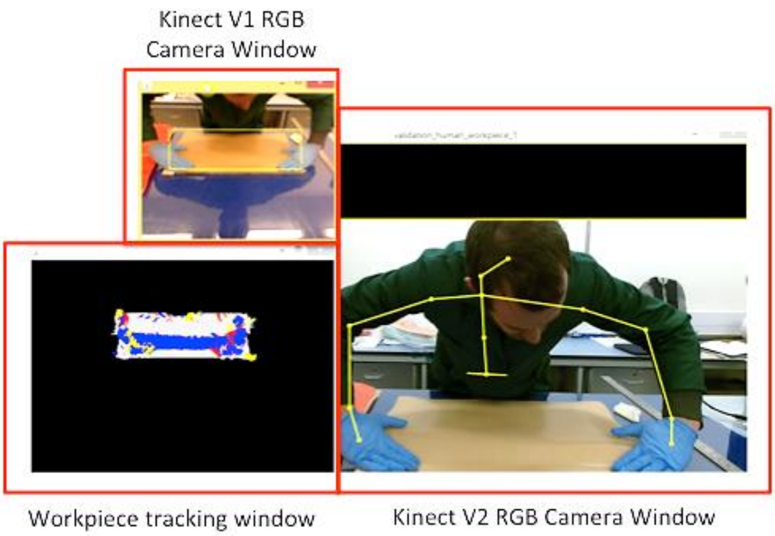


Figure 95: Task capture application windows

Time (s)	Timestamp (s)	Frame_No	WP_Progress_State	Head_X	Head_Y	Head_Z	LH_Hand_X	LH_Hand_Y	LH_Hand_Z	RH_Hand_X	RH_Hand_Y	RH_Hand_Z
56.8	56.8	438	0	1093	322	702	901	869	781	1693	843	662
56.9	56.9	439	0	1103	313	705	900	869	781	1702	896	652
57.1	57.1	440	0	1129	284	714	897	868	780	1699	923	733
57.2	57.2	441	0	1166	254	729	894	873	778	1625	910	807
57.3	57.3	442	0	1208	201	754	891	848	792	1532	803	916
57.5	57.5	443	0	1242	141	799	901	782	815	1536	816	901
57.6	57.6	444	0	1236	119	873	910	741	826	1525	808	881
57.7	57.7	445	0	1194	106	968	905	686	913	1436	760	933
57.9	57.8	446	0	1133	109	1083	913	678	1041	1311	718	1100
58.0	58.0	447	0	1071	126	1218	912	664	1197	1235	713	1299
58.1	58.1	448	0	1026	141	1350	877	719	1365	1192	700	1487
				Workpiece state change								
58.4	58.4	449	1	1008	128	1433	827	735	1574	1167	680	1586
58.5	58.5	450	1	1006	130	1503	821	717	1707	1167	698	1617
58.6	58.6	451	1	1010	133	1569	834	709	1750	1169	695	1678
58.8	58.8	452	1	1019	136	1638	866	704	1738	1168	689	1714
58.9	58.9	453	1	1029	143	1675	894	687	1742	1165	686	1735
59.0	59.0	454	1	1038	148	1707	896	683	1744	1164	683	1746
59.2	59.1	455	1	1043	151	1725	895	689	1745	1165	683	1746
59.3	59.3	456	1	1046	152	1731	889	676	1745	1166	685	1747
59.4	59.4	457	1	1051	152	1728	891	696	1745	1168	683	1745
59.5	59.5	458	1	1060	148	1707	893	693	1726	1180	666	1743

Figure 96: Spreadsheet recording the technician's motion and workpiece state

Step 2: Segment

As explained in section 7.4.4, the workpiece progress is tracked by monitoring the surface orientations of the prepreg ply in the pre-defined area where the mould is located. When the technician manipulates the ply during the task, the workpiece tracking is disturbed because of the presence of the technician's hands in the area (Figure 97).

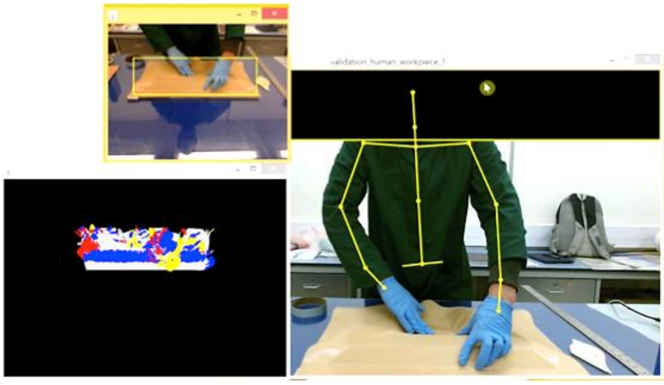


Figure 97: Technician's hands interfering with workpiece tracking during layup

Therefore, instead of recording the workpiece progress at regular time intervals, it is recorded whenever the technician completes laying up a sector of the mould (Figure 91). The technician is therefore instructed to move back from the workpiece by at least 200mm after he has completed working on a sector and this movement is tracked by the Kinect sensor to record the state of the workpiece. The workpiece state is also recorded as a running number in the spreadsheet in which the technician's skeletal motion is recorded. This way the workpiece state is recorded at the beginning of the task and after every sector is completed in order to track its progress throughout the layup task.

The resulting seven workpiece states that correspond to the seven workpiece sectors and the associated human action states are shown in Figure 98. It must be noted that in this case study, the segmentation of human-workpiece interaction data is done at the time of its capture rather than as a separate step.

Figure 98: Human action and resulting workpiece states for technician T1

Step 3: Model

The nomenclature of the human and workpiece states is based on the workpiece sector that those states belong to. The workpiece sectors are illustrated in Figure 91. The human action and workpiece states generated and their sequence observed during the layup tasks performed by the two technicians are shown in Table 28.

Table 28: Human action and workpiece states for technicians T1 and T2

State Sequence	Technician T1		Technician T2	
	Human Action State	**Workpiece State**	**Human Action State**	**Workpiece State**
1	H_C_T	WP_C_T	H_C_T	WP_C_T
2	H_C_M	WP_C_M	H_C_M	WP_C_M
3	H_R_T	WP_R_T	H_R_T	WP_R_T
4	H_L_T	WP_L_T	H_R_MB	WP_R_MB
5	H_C_B	WP_C_B	H_L_T	WP_L_T
6	H_R_MB	WP_R_MB	H_L_MB	WP_L_MB
7	H_L_MB	WP_L_MB	H_C_B	WP_C_B

HMM generation

The human-workpiece interactions involved in the composite layup task are represented using HMM models. The concept of HMM and its adoption in the digitisation framework is described in section 5.3.4. The two technicians use different approaches to perform the layup task and therefore are represented by two different HMM models λ_{T1} and λ_{T2}.

Two more models λ_{T3} and λ_{T4} are generated for the problem-solving scenarios that the two technicians faced during their third task run. T1 solves a problem (wrinkled ply) that occurs in the state 2 (H_C_M now named $H\_C\_M\_P$) in 3 steps, namely, H_M_PS1, H_M_PS2, and H_M_PS3 to restore the task back to the correct state H_C_M. T2 also solves a wrinkled play problem that occurs in state 4 (H_R_MB now named $H\_R\_MB\_P$) in 3 steps, namely, $H\_R\_MB\_PS1$, $H\_R\_MB\_PS1$, and $H\_R\_MB\_PS3$ to restore the task back to H_R_MB.

The transition probabilities in matrix A and the emission probabilities in matrix B for all the four HMM models are generated after consulting the two technicians. These models now represent the strategies used by the two technicians during their individual layup tasks. Initial matrix π is the same for both the technicians since they both start the lay up process at WP_C_T sector. The four models are shown in Figure 99 to Figure 102.

π	H_C_T	H_C_M	H_R_T	H_L_T	H_C_B	H_R_MB	H_L_MB
	0.5	0.05	0.15	0.15	0.05	0.05	0.05

A	H_C_T	H_C_M	H_R_T	H_L_T	H_C_B	H_R_MB	H_L_MB
H_C_T	0.03	0.60	0.15	0.15	0.03	0.03	0.03
H_C_M	0.05	0.05	0.40	0.20	0.10	0.10	0.10
H_R_T	0.05	0.05	0.05	0.50	0.10	0.20	0.05
H_L_T	0.05	0.05	0.05	0.05	0.50	0.15	0.15
H_C_B	0.04	0.04	0.04	0.04	0.04	0.50	0.30
H_R_MB	0.05	0.05	0.05	0.05	0.05	0.05	0.70
H_L_MB	0.05	0.20	0.10	0.10	0.40	0.10	0.05

B	WP_C_T	WP_C_M	WP_R_T	WP_L_T	WP_C_ B	WP_R_MB	WP_L_MB
H_C_T	0.90	0.02	0.02	0.02	0.02	0.02	0.02
H_C_M	0.02	0.90	0.02	0.02	0.02	0.02	0.02
H_R_T	0.02	0.02	0.90	0.02	0.02	0.02	0.02
H_L_T	0.02	0.02	0.02	0.90	0.02	0.02	0.02
H_C_B	0.02	0.02	0.02	0.02	0.90	0.02	0.02
H_R_MB	0.02	0.02	0.02	0.02	0.02	0.90	0.02
H_L_MB	0.02	0.02	0.02	0.02	0.02	0.02	0.90

Figure 99: λ_{T1} (normal scenario) for technician T1

π	H_C_T	H_C_M	H_R_T	H_R_MB	H_L_T	H_L_MB	H_C_B
	0.5	0.05	0.15	0.05	0.15	0.05	0.05

A	H_C_T	H_C_M	H_R_T	H_R_MB	H_L_T	H_L_MB	H_C_B
H_C_T	0.025	0.6	0.15	0.025	0.15	0.025	0.025
H_C_M	0.05	0.05	0.4	0.1	0.2	0.1	0.1
H_R_T	0.05	0.05	0.05	0.5	0.2	0.05	0.1
H_R_MB	0.05	0.05	0.05	0.15	0.5	0.15	0.2
H_L_T	0.04	0.04	0.04	0.04	0.04	0.5	0.3
H_L_MB	0.05	0.05	0.05	0.05	0.05	0.05	0.7
H_C_B	0.05	0.4	0.1	0.1	0.1	0.05	0.2

B	WP_C_T	WP_C_M	WP_R_T	WP_R_MB	WP_L_T	WP_L_MB	WP_C_B
H_C_T	0.9	0.02	0.02	0.02	0.02	0.02	0.02
H_C_M	0.02	0.9	0.02	0.02	0.02	0.02	0.02
H_R_T	0.02	0.02	0.9	0.02	0.02	0.02	0.02
H_R_MB	0.02	0.02	0.02	0.9	0.02	0.02	0.02
H_L_T	0.02	0.02	0.02	0.02	0.9	0.02	0.02
H_L_MB	0.02	0.02	0.02	0.02	0.02	0.9	0.02
H_C_B	0.02	0.02	0.02	0.02	0.02	0.02	0.9

Figure 100: λ_{T2} (normal scenario) for technician T2

π	H_C_T	H_C_M_P	H_C_M_P_S1	H_C_M_P_S2	H_C_M_P_S3	H_C_M	H_R_T	H_L_T	H_C_B	H_R_MB	H_L_MB
	0.525	0.2	0.025	0.025	0.025	0.025	0.05	0.05	0.025	0.025	0.025

A	H_C_T	H_C_M_P	H_C_M_P_S1	H_C_M_P_S2	H_C_M_P_S3	H_C_M	H_R_T	H_L_T	H_C_B	H_R_MB	H_L_MB
H_C_T	0.0125	0.6	0.0125	0.0125	0.0125	0.0125	0.15	0.15	0.0125	0.0125	0.0125
H_C_M_P	0.025	0.025	0.5	0.06	0.05	0.05	0.1	0.1	0.03	0.03	0.03
H_C_M_P_S1	0.025	0.025	0.05	0.5	0.06	0.05	0.1	0.1	0.03	0.03	0.03
H_C_M_P_S2	0.025	0.025	0.05	0.05	0.5	0.06	0.1	0.1	0.03	0.03	0.03
H_C_M_P_S3	0.025	0.025	0.04	0.05	0.05	0.5	0.15	0.1	0.02	0.02	0.02
H_C_M	0.025	0.025	0.025	0.025	0.025	0.025	0.5	0.1	0.2	0.025	0.025
H_R_T	0.025	0.025	0.025	0.025	0.025	0.025	0.025	0.5	0.1	0.2	0.025
H_L_T	0.025	0.025	0.025	0.025	0.025	0.025	0.025	0.025	0.7	0.05	0.05
H_C_B	0.025	0.025	0.025	0.025	0.025	0.025	0.025	0.025	0.025	0.5	0.275
H_R_MB	0.025	0.025	0.025	0.025	0.025	0.025	0.025	0.025	0.025	0.025	0.75
H_L_MB	0.025	0.2	0.025	0.025	0.025	0.025	0.05	0.05	0.35	0.2	0.025

B	WP_C_T	WP_C_M_P	WP_C_M_P_S1	WP_C_M_P_S2	WP_C_M_P_S3	WP_C_M	WP_R_T	WP_L_T	WP_C_B	WP_R_MB	WP_L_MB
H_C_T	0.9	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
H_C_M_P	0.01	0.9	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
H_C_M_P_S1	0.01	0.01	0.9	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
H_C_M_P_S2	0.01	0.01	0.01	0.9	0.01	0.01	0.01	0.01	0.01	0.01	0.01
H_C_M_P_S3	0.01	0.01	0.01	0.01	0.9	0.01	0.01	0.01	0.01	0.01	0.01
H_C_M	0.01	0.01	0.01	0.01	0.01	0.9	0.01	0.01	0.01	0.01	0.01
H_R_T	0.01	0.01	0.01	0.01	0.01	0.01	0.9	0.01	0.01	0.01	0.01
H_L_T	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.9	0.01	0.01	0.01
H_C_B	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.9	0.01	0.01
H_R_MB	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.9	0.01
H_L_MB	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.9

Figure 101: λ_{T3} (problem-solving scenario) for technician T1

π	H_C_T	H_C_M	H_R_T	H_R_MB_P	H_R_MB_PS1	H_R_MB_PS2	H_R_MB_PS3	H_R_MB	H_L_T	H_L_MB	H_C_B
	0.525	0.025	0.05	0.2	0.025	0.025	0.025	0.025	0.05	0.025	0.025

A	H_C_T	H_C_M	H_R_T	H_R_MB_P	H_R_MB_PS1	H_R_MB_PS2	H_R_MB_PS3	H_R_MB	H_L_T	H_L_MB	H_C_B
H_C_T	0.0125	0.6	0.15	0.0125	0.0125	0.0125	0.0125	0.0125	0.15	0.0125	0.0125
H_C_M	0.025	0.025	0.5	0.025	0.025	0.025	0.025	0.05	0.2	0.05	0.05
H_R_T	0.025	0.025	0.025	0.6	0.025	0.025	0.025	0.11	0.11	0.015	0.015
H_R_MB_P	0.025	0.025	0.025	0.025	0.5	0.025	0.025	0.15	0.15	0.025	0.025
H_R_MB_PS1	0.025	0.025	0.025	0.025	0.025	0.5	0.025	0.15	0.15	0.025	0.025
H_R_MB_PS2	0.025	0.025	0.025	0.025	0.025	0.025	0.5	0.15	0.15	0.025	0.025
H_R_MB_PS3	0.025	0.025	0.025	0.025	0.025	0.025	0.025	0.5	0.1	0.2	0.025
H_R_MB	0.025	0.025	0.025	0.025	0.025	0.025	0.025	0.025	0.7	0.05	0.05
H_L_T	0.025	0.025	0.025	0.025	0.025	0.025	0.025	0.025	0.025	0.5	0.275
H_L_MB	0.025	0.025	0.025	0.025	0.025	0.025	0.025	0.025	0.025	0.025	0.75
H_C_B	0.025	0.2	0.025	0.025	0.025	0.025	0.05	0.05	0.35	0.2	0.025

B	WP_C_T	WP_C_M	WP_R_T	WP_R_MB_P	WP_R_MB_PS1	WP_R_MB_PS2	WP_R_MB_PS3	WP_R_MB	WP_L_T	WP_L_MB	WP_C_B
H_C_T	0.9	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
H_C_M	0.01	0.9	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
H_R_T	0.01	0.01	0.9	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
H_R_MB_P	0.01	0.01	0.01	0.9	0.01	0.01	0.01	0.01	0.01	0.01	0.01
H_R_MB_PS1	0.01	0.01	0.01	0.01	0.9	0.01	0.01	0.01	0.01	0.01	0.01
H_R_MB_PS2	0.01	0.01	0.01	0.01	0.01	0.9	0.01	0.01	0.01	0.01	0.01
H_R_MB_PS3	0.01	0.01	0.01	0.01	0.01	0.01	0.9	0.01	0.01	0.01	0.01
H_R_MB	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.9	0.01	0.01	0.01
H_L_T	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.9	0.01	0.01
H_L_MB	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.9	0.01
H_C_B	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.9

Figure 102: λ_{T4} (problem-solving scenario) for technician T2

None of the HMM models required any optimisation because the Baum-Welch algorithm did not go beyond the first iteration and diverged immediately indicating that the models appropriately represented the tasks. Though this does not imply that the models are at their most optimum, it does imply that the probability inputs from the two technicians were aptly representative of the workpiece state sequence observed and captured during the tasks.

Step 4: Extract

The main aim of the digitisation framework in the study of manual composite layup is to extract manufacturing knowledge embedded within the layup process. This knowledge consists of the following constituents that may be unique to the technician performing the task depending on his/her skills and experience:

1. Approach taken or strategy adopted by the technicians to successfully lay the prepreg ply over the mould taking into account the shape of the mould and its surface orientations.
2. Ply manipulation techniques used to lay the ply over critical areas of the mould and the time taken for each technique.
3. Motion mechanics of the technician’s body when applying the layup techniques and
4. Problem-solving techniques used to correct layup errors such as folds and wrinkles in the laid up ply.

According to this research, the above constituents of manufacturing knowledge are embedded within the HMM models that represent the layup tasks and associate the tasks with the technician. Therefore, the HMM models are queried with a task scenario and the most likely human action sequence responsible for that task scenario is obtained. The above knowledge constituents are then decoded from the human action sequence in the next step of the framework.

The task scenario is nothing but a sequence of states that the workpiece goes through during the task. This workpiece sequence could also be one that is not previously observed but is queried to ascertain the most likely human response to unforeseen task scenarios. However, since there are multiple models representing the variations observed in the same task, the most appropriate model for a given task scenario must be first identified.

HMM evaluation – picking the right model for a given task scenario

Consider task scenarios represented by the workpiece state sequences $O_{Q1} = \{WP\_C\_T,\ WP\_C\_M,\ WP\_R\_T,\ WP\_L\_T,\ WP\_C\_B,\ WP\_R\_MB,\ WP\_L\_M\}$ and $O_{Q2} = \{WP\_C\_T,\ WP\_C\_M,\ WP\_R\_T,\ WP\_R\_MB,\ WP\_L\_T,\ WP\_L\_M,\ WP\_C\_B\}$

Since these are previously observed and captured workpiece state sequences, a simple comparison with the workpiece state sequences of all the four models results in identification of HMM λ_{T1} and λ_{T2} as the most likely models that represent $O_{Q1}$and O_{Q2} respectively.

<u>For normal task scenarios</u>

Consider another task scenario $O_{Q3} = \{WP\_C\_T,\ WP\_C\_M,\ WP\_C\_B,\ WP\_R\_T,\ WP\_R\_MB,\ WP\_L\_T,\ WP\_L\_MB\}$. This scenario has not been observed previously therefore the 'Forward' algorithm is used to identify the most probable model that represents it. The probabilities of the four HMM models given the workpiece state sequence O_{Q3} are:

$$P(\lambda_{T1} \mid O_{Q3}) = 8.34e-7$$

$$P(\lambda_{T2} \mid O_{Q3}) = 3.12e-7$$

$$P(\lambda_{T3} \mid O_{Q3}) = 1.90e-9$$

$$P(\lambda_{T4} \mid O_{Q3}) = 1.17e-8$$

Since $P(\lambda_{T1} \mid O_{Q3})$ is the highest probability, λ_{T1} is picked as the most likely model to represent the task scenario of O_{Q3}.

<u>For problem-solving task scenarios</u>

Consider the following problem solving scenarios

$O_{Q4} = \{WP\_C\_M\_P,\ WP\_C\_M\_PS1,\ WP\_C\_M\_PS2,\ WP\_C\_M\_PS3,\ WP\_C\_M\}$ and $O_{Q5} = \{WP\_R\_MB\_P,\ WP\_R\_MB\_PS1,\ WP\_R\_MB\_PS2,\ WP\_R\_MB\_PS3,$

$WP\_R\_MB\}$

The 'Forward' algorithm picks HMM models λ_{T3} and λ_{T4} as the most likely models that closely represent O_{Q4} and O_{Q5}.

Extraction of human action sequence from the HMM model

In this case of the previously observed task scenarios O_{Q1} and O_{Q2}, the human action sequences can be directly extracted from the identified models λ_{T1} and λ_{T2} without the need to use the Viterbi algorithm and the sequences are

$H_{Q1} = \{H\_C\_T,\ H\_C\_M,\ H\_R\_T,\ H\_L\_T,\ H\_C\_B,\ H\_R\_MB,\ H\_L\_MB\}$

and

$H_{Q2} = \{H\_C\_T,\ H\_C\_M,\ H\_R\_T,\ H\_R\_MB,\ H\_L\_T,\ H\_L\_M,\ H\_C\_B\}$

However for task scenario O_{Q3}, which is not previously observed, the Viterbi algorithm is needed to find the most likely human action sequence from the identified HMM λ_{T1}. The algorithm yields H_{Q3} as the most likely human response to task scenario O_{Q3}.

$H_{Q3} = \{H\_C\_T,\ H\_C\_M,\ H\_C\_B,\ H\_R\_T,\ H\_R\_MB,\ H\_L\_T,\ H\_L\_MB\}$

The Viterbi algorithm is also used to find the human action sequence during problem solving scenarios O_{Q4} and O_{Q5} from the identified models λ_{T3} and λ_{T4} respectively. The human action sequences identified are

$H_{Q4} = \{H\_C\_M\_P,\ H\_C\_M\_PS1,\ H\_C\_M\_PS2,\ H\_C\_M\_PS3,\ H\_C\_M\}$

and

$H_{Q5} = \{H\_R\_MB\_P,\ H\_R\_MB\_PS1,\ H\_R\_MB\_PS2,\ H\_R\_MB\_PS3,\ H\_R\_MB\}$

It can be observed that the human action sequences match the workpiece state sequences perfectly. This is because in this case, there is a close association between human actions and the resulting workpiece states according to the emission matrices of the four HMM models. The probability of one human action state resulting in a workpiece state other than which it is associated with in the emission matrix is extremely low. However, it must be noted that this phenomenon is not necessarily true for other tasks where one human action can result in more than one workpiece state as can be seen in other case studies.

Step 5: Decode

In the previous step, human actions that are most likely responsible for a task scenario are obtained by querying the appropriate HMM model with the workpiece state sequence representing the scenario. From the extracted human actions, the four constituents of manufacturing knowledge that are embedded within the actions can be decoded as follows:

Technician's task strategy

This is the highest level of knowledge embedded within a manufacturing task. The approach taken by the technician to lay the ply on the mould depends on the geometry of the mould and the knowledge of how the ply that is already laid up in one sector can affect the lay up on the neighbouring sectors. There are also sector dependencies where one sector must be laid before another to avoid layup errors such as folds or wrinkles. The approach taken by one technician may vary from the other based on individual skills and experience. In this case, the task strategy can be observed from the sequence of actions taken by the technician to perform the task. This sequence is obtained automatically in the previous step where human action state sequences are obtained for any given task scenarios.

For example, for the task scenario O_{Q1}, the human action sequence obtained was H_{Q1}. This sequence came from model λ_{T1} that belongs to technician T1. Therefore, it can be deduced that in order to get the workpiece through the task scenario of O_{Q1}, the skills and experience of technician T1 and the human action sequence of H_{Q1} would be most suitable. The human action sequence in H_{Q2} is similarly obtained for O_{Q2} from model λ_{T2} and hence reveals the task strategy used by technician T2 in the task. The difference in task strategies adopted by T1 and T2 for laying up the same mould with the same ply is illustrated in Figure 103.

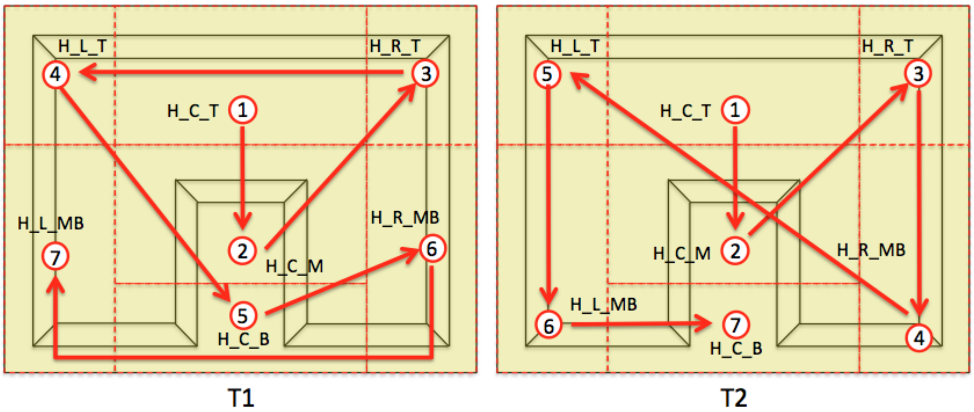


Figure 103: Difference in task strategy between technicians T1 and T2

Ply manipulation techniques

The actual action data from within each human action state from the extracted sequence is obtained from the spreadsheet that contains the skeletal motion data of the technician. The x, y and z motion of the technician's left and right hands are plotted against time so that the motion patterns can be visualised for each state thereby revealing the techniques used by the technician in each state. As an example, the technician's hand actions during state 5 (WP_C_B) from the extracted human action state sequence H_{Q1} is shown below. Similarly, motion plots of the technician's rest of the upper body joints, such as elbows, shoulders, head and torso can also be obtained and visualised.

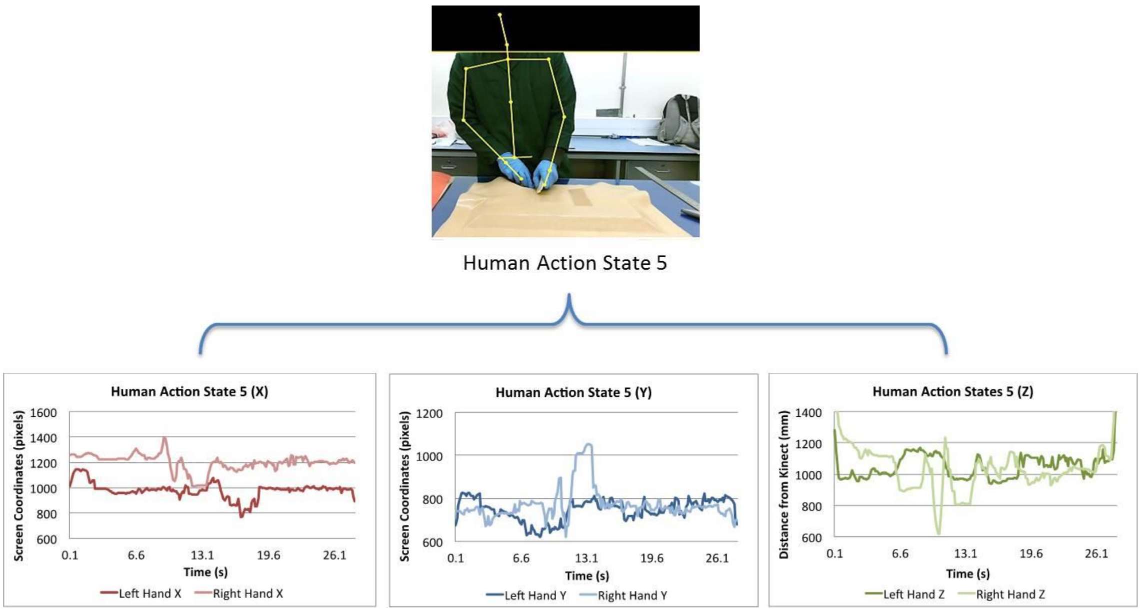
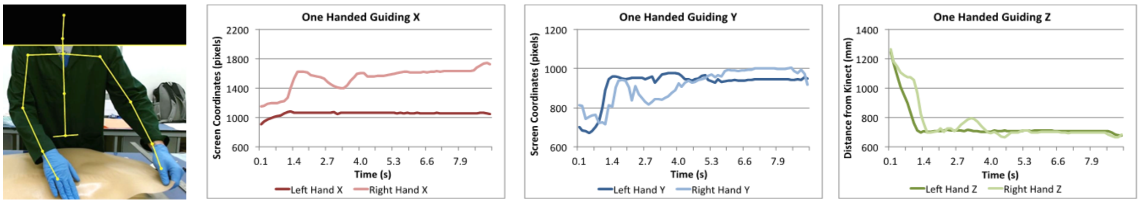
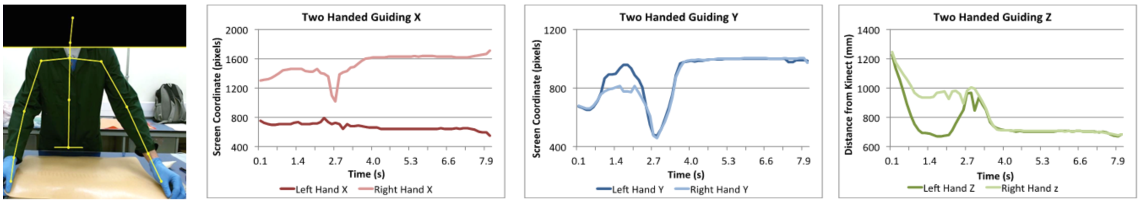


Figure 104: Technician's left and right hand motion plots (x, y, z)

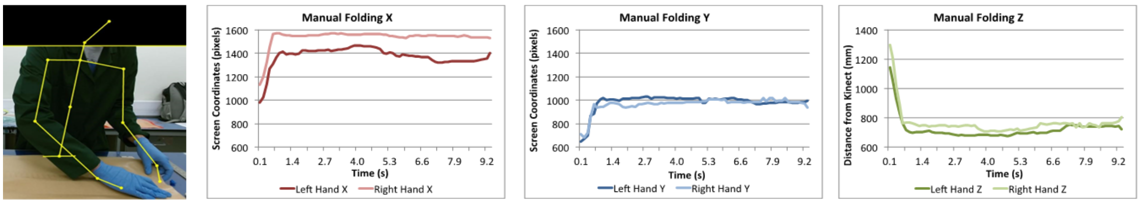
According to the Elkington et al., 2015, there are 7 standard hand techniques used by the technicians to manipulate the ply. The techniques are (1) one handed guiding, (2) two handed guiding, (3) manual folding, (4) mould interaction shearing, (5) double tension shearing, (6) tension secured shearing, and (7) smoothing and tensioning. One or more of these techniques are used within each of the human action states and therefore can be isolated and revealed as an important constituent of manufacturing knowledge embedded within the manual layup task. The seven techniques captured are listed in Figure 105.



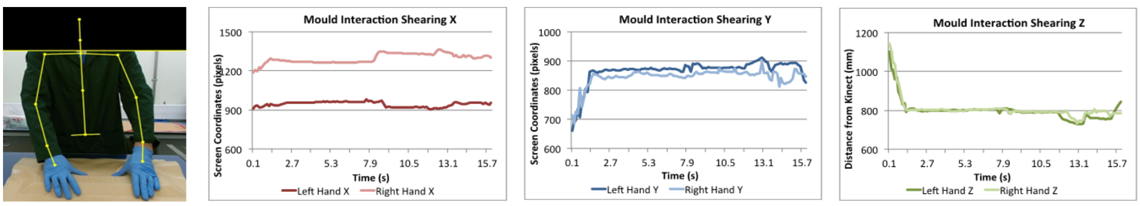
(1) One Handed Guiding



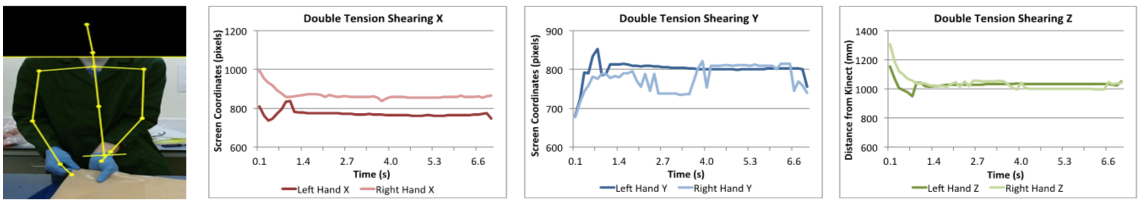
(2) Two Handed Guiding



(3) Manual Folding



(4) Mould Interaction Shearing



(5) Double Tension Shearing

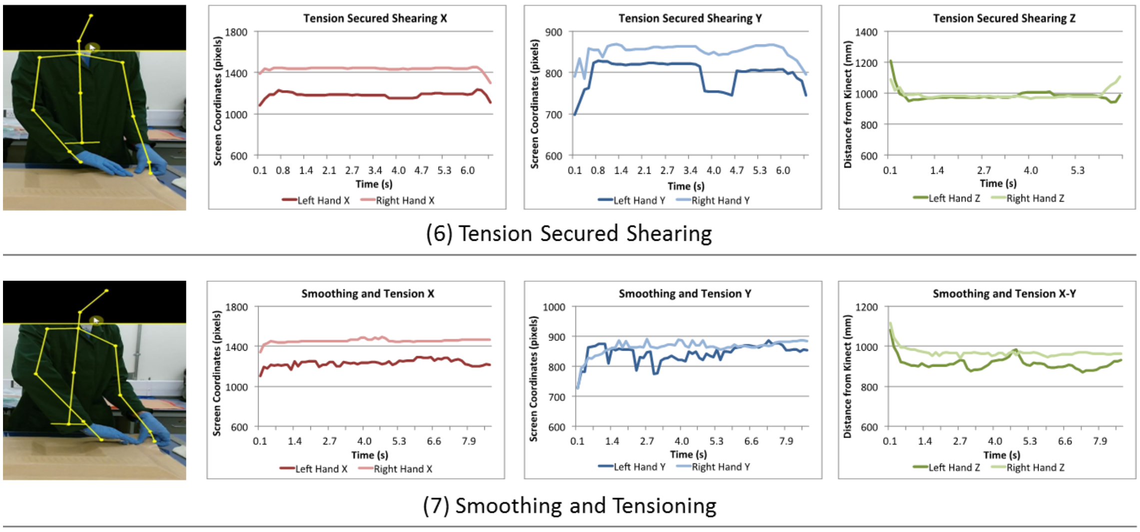


Figure 105: The hand techniques of ply manipulation

Time taken per workpiece sector

The time taken by the technician for each human action state can be easily obtained from the spreadsheet containing the task capture data because it also contains the timestamp of each Kinect frame from which the human skeletal joint coordinates are captured. With this information, the workpiece areas that take longer to layup than others can be identified which might indicate higher mould shape complexity in those areas. This also allows comparison of time taken by the two technicians to layup the 7 sectors of the workpiece (**Figure 106**) as a measure to compare skill levels.

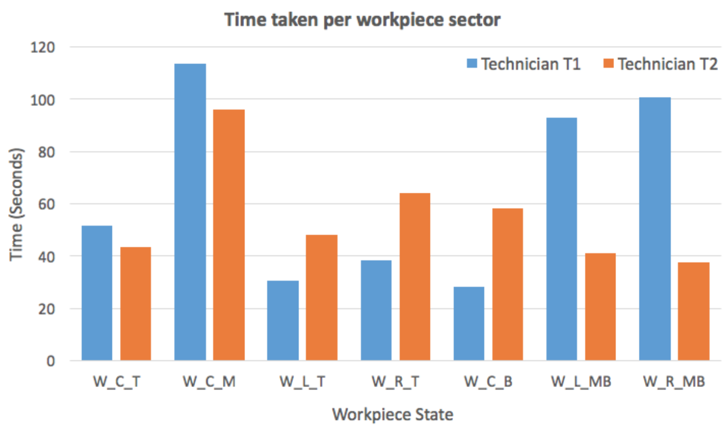


Figure 106: Difference in time taken by the two technicians per workpiece sector

Motion mechanics of the technician's body

The skeletal joint coordinates belonging to the upper body of the technician are recorded in the spreadsheet in the capture step of the framework. From these joint coordinates, several motion parameters can be obtained using vector computing. Examples of four different motion mechanics computed using skeletal coordinate data is shown in Figure 107. This data helps in finding the technician's body posture and orientation, glance angle and the positions of his hands while performing critical hand layup techniques.

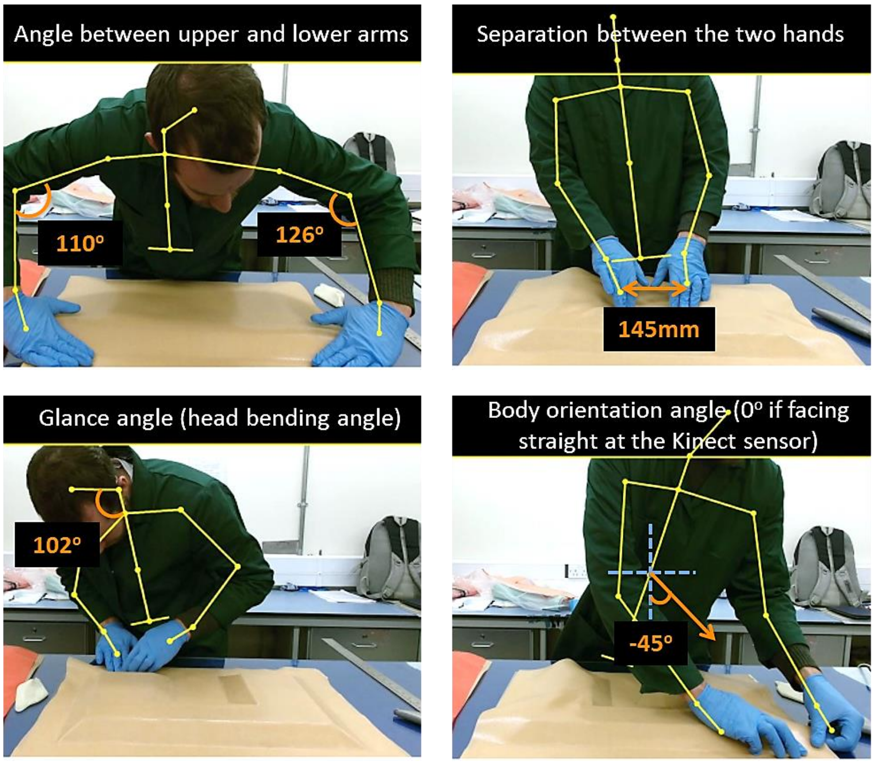


Figure 107: Body motion mechanics computed from skeletal coordinate data

Another critical knowledge constituent that can be obtained is the gesture speed. In the layup process, the hand speed while performing ply manipulation is critical to the success of the process, especially in certain critical areas of the mould and therefore is also indicative of the technician's skill. A difference in the hand speeds between two technicians also implies a difference in skill levels. Higher hand speeds however are not necessarily a sign of superior skills in a layup task.

A small portion of hand motion data is plotted in Figure 108 and the hand speed in two zones A and B is computed from skeletal coordinate data. In this example, screen coordinates are converted to Kinect sensor coordinates so that speed values are obtained in mm/s rather than in pixels/s.

In zone A:

Right hand X speed = 228.6mm/s

Left Hand Y speed = 277.5mm/s

Left Hand Z speed = 440mm/s

In zone B:

Right hand X speed = 364.6mm/s

Left Hand Y speed = 377.7mm/s

Left Hand Z speed = 211mm/s

Figure 108: Hand motion speeds in zones A and B

Problem-solving techniques

In this study, the two technicians deliberately introduced a problem while laying up a particular area of the workpiece. An error was made in the layup resulting in a wrinkle on the surface of the ply. This problem was solved by the technicians using their individual techniques in 3 steps resulting in the wrinkle being removed from the surface of the ply. The problem and the problem solving steps captured for technician T1 are shown in Figure 109.

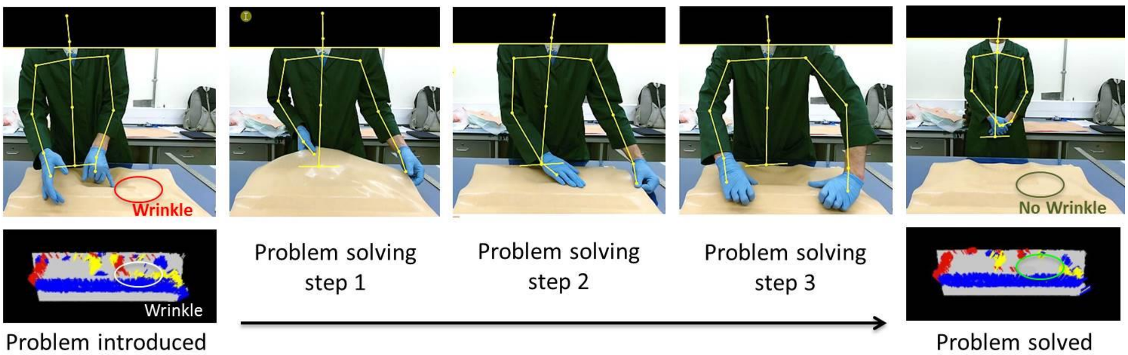


Figure 109: Problem solving scenario within the layup task

The human action state sequence used by technician T1 is already extracted in step 2, which is H_{Q4}.

$$H_{Q4} = \{H\_C\_M\_P,\ H\_C\_M\_PS1,\ H\_C\_M\_PS2,\ H\_C\_M\_PS3,\ H\_C\_M\}$$

For each state in this sequence, the hand techniques used as well as the motion mechanics during these techniques can be obtained.

Step 6: Reproduce

The spreadsheet that contains a stream of skeletal joint coordinates of the technician's upper body and the workpiece states is an accurate digital representation of the task. This way a task can be digitally captured and stored in a text file less than 1 megabyte in size instead of the usual practice of capturing and storing tasks in video files in sizes of the order of a few gigabytes. The skeletal coordinates stored in the spreadsheet can be rendered graphically to produce a stickman animation of the captured layup task. However, when the need to refer to greater level of detail, such as finger positions, is required then the animation does not suffice and the actual colour images belonging to the concerned part of the task are required.

A few snapshots of such an animation and the corresponding workpiece states are shown in Figure 110. Visualising the task animation as well as studying the decoded constituents of manufacturing knowledge enables the transfer of skills from an experienced technician to a novice technician. Since the various task scenarios are stored in digital models, access to the knowledge within these models is possible on-demand.

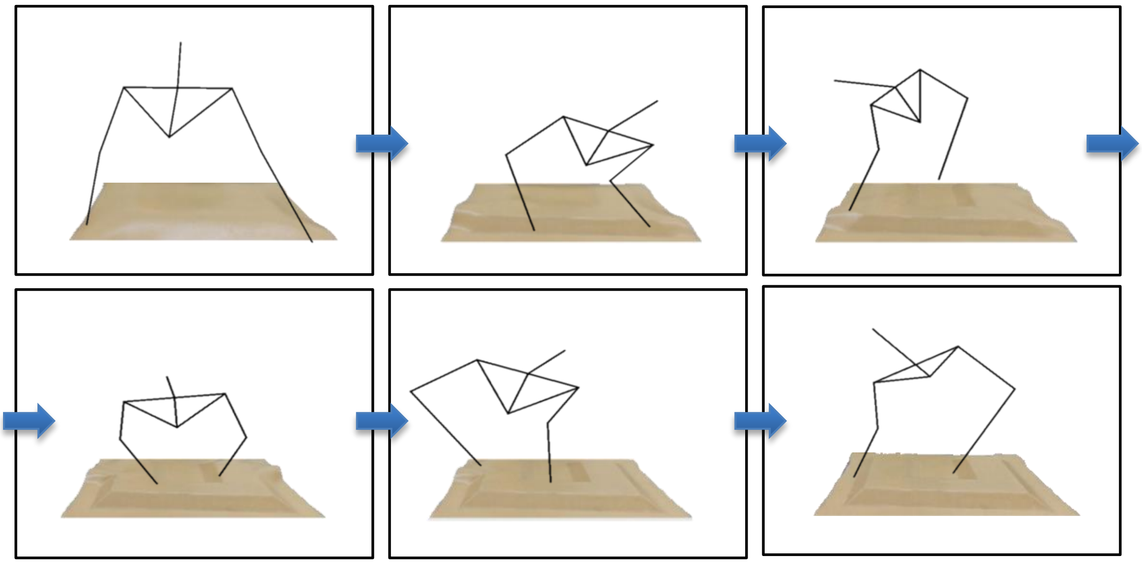


Figure 110: Layup task animation

7.4.6 Research benchmarking

The final case study has demonstrated that the 6-step framework is successfully able to digitise, the manufacturing knowledge embedded within a real-world composite layup task, considered as an important manual task in the high value manufacturing industry. It is also reports how layup skills, represented by the constituents of manufacturing knowledge, can be acquired from skilled technicians so that they could be transferred to novice technicians.

An approach to benchmark the framework is by comparing its methods and outcomes to those reported by similar research done by Elkington et al. (2015) at the ACCIS (University of Bristol), in which the knowledge associated with manual composite layup tasks were extracted on the basis of visual observation. The comparison is presented in Table 29 with comments on whether the compared aspects between the two researches are similar, different or complementary.

Table 29: Comparison between the proposed research and the ACCIS research (Elkington et al., 2015)

Aspect	This work	ACCIS work	Comments
Aim	To capture, extract and decode the manufacturing knowledge embedded within a manual layup task in digitised form.	To fully understand the manual layup task in detail in order to further develop the task, improve training methods for the task and support automation of the task.	The aims are similar in nature but this work also aims to digitise the manufacturing knowledge in a form that can be easily transferred.
Concept	The human-workpiece interactions involved in the layup task are captured and modelled into digital data and the knowledge constituents are automatically extracted and decoded from the digital data.	The layup task is observed and recorded as video footages which are then revisited several times to manually extract the manufacturing knowledge constituents embedded within the task.	This work uses a structured 6-step framework to digitise manufacturing knowledge whereas the ACCIS work uses a 2-step method to analyse and extract knowledge from the task recorded in video files.
Task	The layup task was performed by two technicians for 3 times each to provide 6 task datasets. Both technicians were moderately experienced.	The layup task was performed by four technicians for 3 times each to obtain 12 task datasets. Out of the four technicians, two were vastly experienced and two were moderately experienced.	The ACCIS work has much richer and diverse task datasets as compared to the proposed work due to the higher number of technicians and the varying skill levels of the technicians involved. The skills levels of differently skilled technicians can be compared in the ACCIS work, which is not possible

			in this work where both technicians are equally skilled and experienced.
Task Observation Setup	Gaming interface sensors such as the Microsoft Kinect are used to capture the task and record it using low definition numerical data.	A video camera is used to record the task in a high fidelity video file.	The ACCIS work uses HD camera as against the low-resolution Kinect sensor used by this work. However, the Kinect sensor produces both RGB and depth images thereby providing the capability to obtain human motion tracking and object recognition from the image frames at the same time.
Task Capture Method	The technician's actions are automatically captured as skeletal joint motion of the upper body. The workpiece progress is also captured simultaneously by tracking the manipulation of the ply over the mould.	The only source of task capture is to visually observe the technician's actions and effects of these actions on the ply throughout the task. Any significant actions are noted manually for future revisits and analysis.	This work uses innovative software methods that access the RGB and depth image streams coming from the Kinect sensor to track human actions and their effects on the workpiece automatically. Since all this data is digital, it is stored in a spreadsheet for subsequent modelling and knowledge extraction.
Outcome	The digitised task scenarios are modelled into hidden Markov models from which the task execution strategy, ply manipulation techniques, mechanics of	By revisiting the video footage several times, the hand techniques used by the technicians to manipulate the ply are observed, noted and grouped into categories for effective documentation.	This work produces a rich knowledge base from the captured task right from the highest levels of task execution strategy down to the mechanics of technician's actions (upper body motion) when manipulating the ply during

	gestures as well as problem-solving actions during the task can be extracted and decoded.	Screenshots of the video footage showing these techniques are captured and stored. The emphasis is to understand the skill used in each of these techniques as well as the purpose of each technique based on when and where it is used during the layup task.	the task. The ACCIS work can only extract ply manipulation techniques at the hand level. The use of modelling allows this work to predict human response to unobserved task scenarios, which is not possible in the ACCIS work. In order to get down to finer task details such as finger positions during ply manipulation, both works have access to video data from which these details can be retrieved.
Knowledge Reproduction	Since the task and its manufacturing knowledge constituents are captured in digital form, those can be easily reproduced in using multiple media, the most basic of which is 2D animation, providing an effective and easy medium for skill transfer.	Since the extraction of manufacturing knowledge constituents is manual in nature, their reproduction is only possible by documenting them in static documents and revisiting the video footage of the tasks. This however does not provide any easier means of skill transfer.	The digitised task knowledge produced in this work can be consumed in multiple forms ranging from viewing the various task scenarios in 2D or 3D animation with the manufacturing knowledge constituents augmented within the animation to using virtual reality to visualise the task scenarios in an immersive environment. This potential is not available with the ACCIS work.

The ACCIS research is among the first known to successfully attempt to understand complex manual layup tasks in detail to provide a clear documentation of the hand techniques used by technicians of varying skill levels during manipulation of a pre-impregnated ply. The proposed research due to its advanced task capture, digitisation and modelling methods and its capability to extract, decode and reproduce manufacturing knowledge constituents embedded within not just the observed task scenarios but also from the unobserved ones, makes it a potential candidate to advance the understanding of not just manual composite layup tasks but also a broader spectrum of manual manufacturing tasks from several sectors of the manufacturing industry based on the other case studies reported earlier.

7.5 Validation summary

The framework has been successfully applied to digitise manual manufacturing task knowledge as demonstrated using the 5 chosen manual tasks including one real-life manufacturing task. Through these 5 tasks, the framework was extensively tested across variations in tasks, workpieces, humans, task strategies, nature of human actions, and ICT methods and tools used.

The framework is built on a strong foundation of the advanced human-workpiece interaction theory. The strength of the theory was essential to hold the framework up against the extensive testing. The theory itself was based on the research hypothesis that arose out of the research problem. The literature review contributed to the design of the framework structure as well as to the identification of seminal theories in human behaviour and object analysis for advancing the human-workpiece interaction theory.

Table 30 helps to clearly visualise the diversity and depth of investigation that this research has undergone from the initial to the concluding period. The text in vertical signifies the underpinning structure on which the digitisation framework was built and the horizontal text signifies the framework steps each with its own methods and tools used.

The case studies provided a platform for implementation, testing and validation of the framework. It can be observed that the framework was tested extensively to cover a big variety of tasks and technical requirements on the basis on which it can be said that the framework is successfully validated.

Table 30: Breadth and depth of research

Breadth of investigation					Depth of Investigation								
									Case Studies				
									1. Lego	2. Wheel	3. Pen	4. Table	5. Layup
	Research Aim and Objectives	Human-Workpiece Interaction Theory	Gibson's Theory of Object Affordances	Digitisation Framework	Capture	Method	Gaming Sensor	Kinect V1	√	√	√	√	
								Kinect V2				√	√
							Setup	Single sensor	√		√	√	
								Multiple sensors		√		√	√
						Task	Type	Toy	√				
								Simulated		√	√	√	
								Real-world					√
							Length	Short	√	√	√	√	
								Long					√
							Environment	Controlled	√		√	√	
								Uncontrolled		√			√
							Scenario	Normal	√	√	√	√	√
								Problem-solving	√			√	√
	Literature Review		Rasmussen's Decision Ladder Concept			Human	Action type	Smooth					√
								Jerky	√				
								Cyclical				√	√
								Irregular			√		√
							Gesture type	Small	√		√		√
								Big				√	
						Workpiece	Type	Non-deformable	√	√	√	√	
								Deformable					√
								Small	√		√		
								Large		√		√	√
								Coloured	√				
								Grayscale		√	√	√	√
	Research Hypothesis		Rasmussen's S-R-K Framework					Simple	√		√	√	
								Complex		√			√
							Nature of change	Dimension	√				
								Shape					√
								Position		√	√		
								Configuration				√	
							Tracking method	Edge		√	√		
								Colour & brightness	√				
								Contour				√	
								Surface orientation					√
					Segmentation	Method	Time-sampling					√	
	Research Problem		Fundamental Human-Workpiece Interaction Concept				Workpiece-sampling		√				√
							Trajectory-sampling				√		
							Problem-solving isolation					√	
						Outcome	States		√			√	√
							Therbligs				√		
					Modelling	Hidden Markov Modelling			√		√	√	√
					Knowledge extraction and decoding	Knowledge constituents	Task strategy		√		√	√	√
							Gestures & techniques		√		√	√	√
							Mechanics of motion		√		√	√	√
							Decision making		√			√	√
							Skill comparison between humans					√	√
					Reproduction	Method	Documentation		√		√	√	√
							Animation		√		√	√	√

7.6 Chapter summary

This chapter presents the validation of the proposed framework using 3 case studies. In these case studies, 2 real-life like and 1 real-life manual manufacturing tasks were performed and the manufacturing knowledge embedded within them was successfully digitised using the proposed digitisation framework. The knowledge extracted by the framework for the composite layup task was benchmarked with the knowledge extracted from the same task by researchers at ACCIS, University of Bristol. The variety of tasks and technical requirements that the framework was tested was also summarised to prove the generality and validity of the framework. The framework has its advantages and limitations, which will be covered in the next chapter.

CHAPTER 8

8 DISCUSSION AND CONCLUSIONS

This chapter discusses the key findings and outcomes of this research, maps key concepts in theory to their practical implementations and identifies the research limitations. It discusses the contributions to knowledge made by this research and identifies future work.

This chapter aims to achieve the following objectives:

- Present the contributions to knowledge.
- Discuss the contribution of the human-workpiece interaction theory to the development of the digitisation framework.
- Map the concepts of hierarchical task analysis and cognitive work analysis to the implementation of the digitisation framework.
- Discuss the advantages and limitations of this research.
- Identify future work.
- Present conclusions.

8.1 Overview

The aim of this research is to develop a framework for digitisation of manual manufacturing task knowledge. The motivation behind this research is the need for the industry to capture and digitise the tacit aspects of knowledge, such as human skill, embedded within the manufacturing tasks currently performed by skilled task experts. The captured knowledge being digital in nature can be transformed into appropriate media for fast and easy up-skilling of new people in the workforce and for informing the design of next generation automation solutions. The research is timely because the manufacturing industry especially in high wage economies is beginning to feel the ill effects of a global skill supply crunch and is under intense pressure to maintain its competitiveness compared to the industry from low-wage economies.

In this context, the framework for digitisation of manual manufacturing task knowledge has been developed. The framework is strongly underpinned by the human-workpiece interaction theory that is built around the research hypothesis and is reinforced by the Rasmussen's S-R-K framework and Rasmussen's decision ladder proposed to understand human behaviour during tasks and the seminal theory of object affordances proposed by Gibson. The framework is implemented using an example assembly task and all the 6 steps were successfully applied to digitise the manufacturing knowledge from the task. The framework is also successfully validated using 3 case studies including a real-life manufacturing task, each presenting a different challenge to the framework but collectively demonstrating the framework's potential to be implemented across most manufacturing tasks.

The chapter will discuss the contributions of this research to knowledge, the role of human-workpiece interaction theory in the development of the digitisation framework and the usefulness of the framework for Hierarchical Task Analysis (HTA) and Cognitive Work analysis (CWA). It will present the key findings of this research along with the outcomes vis-à-vis the research objectives and identify the pros and cons of the methods and tools proposed in the framework.

8.2 Contributions to knowledge [vis-à-vis research gaps]

This research provides the following 5 contributions to knowledge:

1. The theory of human-workpiece interactions to decipher human behaviour during manual manufacturing tasks is developed on the basis of the research hypothesis which states that a manual task can be broken down into a series of human-workpiece interactions and by observing and analysing these interactions, the knowledge embedded within the task can be captured and digitised. **[RG1]**
2. A cohesive and holistic framework to digitise manual manufacturing task knowledge making innovative use of consumer-grade hardware and ICT software. The framework is underpinned by the theory of human-workpiece interactions. **[RG2]**
3. For the first time, consumer-grade gaming interface sensors, such as the Kinect, were used to digitise the manufacturing knowledge embedded within manual tasks. Innovative methods that leveraged these sensors were developed to recognise and track human actions and their effect on the workpieces in real-time and were successfully demonstrated using a variety of tasks and scenarios within those tasks. **[RG3]**
4. For the first time, Hidden Markov Modelling (HMM) was used to co-relate human actions to workpiece progress during a manual manufacturing task. Therefore, human-workpiece interactions and the manufacturing knowledge embedded within them are represented using hidden Markov models, which could be used to extract and decode all the key constituents of knowledge on demand. Also for the first time, each task expert could be uniquely represented by his/her own model giving birth to the concept of digital skill models that could be queried to extract each task expert’s skills in a digital, transferable form. **[RG4]**
5. Extraction of manufacturing knowledge such as task strategies from the digital skill models and decoding of the constituents of this knowledge such as workpiece manipulation skills from the raw human-workpiece interaction data provide the means to perform human task analysis with a focus on cognitive work analysis. **[RG5]**

8.3 Mapping the human-workpiece interaction theory to the digitisation framework

The human-workpiece interaction (HWI) theory (section 4.1) is mainly derived from the research hypothesis that by observing human-workpiece interactions that occur during a manual task, the manufacturing knowledge that is embedded within that task can be captured and digitised.

The HWI theory is reinforced by (a) Rasmussen's Skill-Rules-Knowledge (S-R-K) framework, which categorises human behaviour during a task into skill-based, rule-based and knowledge-based behaviour, (b) Rasmussen's decision ladder concept that explains the human behaviour during problem-solving scenarios in the task and (c) Gibson's theory of object affordances that explains how the action chosen by the human to perform on the workpiece at any given instance during the task depends on the affordance presented by the workpiece at that particular instance. Based on the HWI theory, the framework for digitisation of manual manufacturing task knowledge is developed (section 5.2).

8.3.1 Mapping the basic HWI theory to the framework

The first step 'Capture' is designed to record the entire manufacturing task in which the human interacts with the workpiece. The idea is to get raw digital data of human actions and workpiece progress during the task. Therefore, the capture step digitises the fundamental human-workpiece interaction theory from the HWI theory as shown in Figure 111.

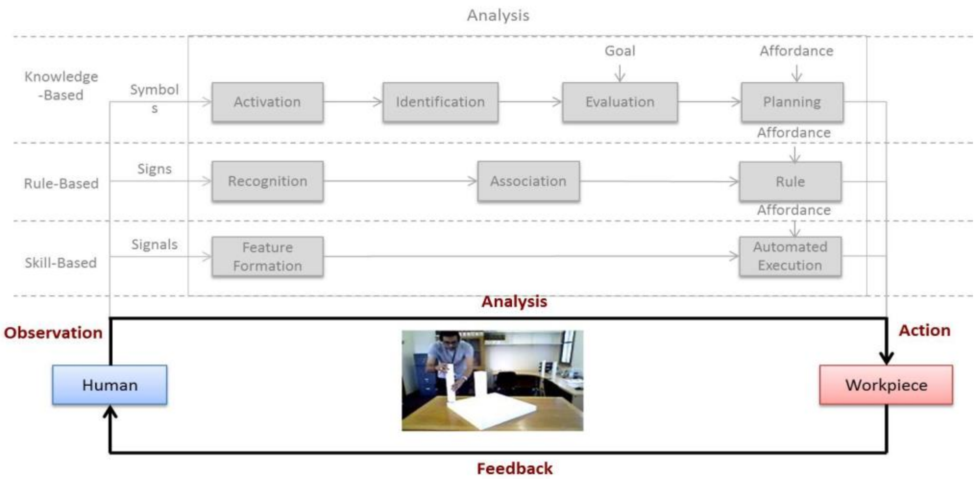


Figure 111: Digitisation of basic human-workpiece interactions

The human-workpiece interaction data can be visualised by using motion charts with augmented workpiece progress data as shown in Figure 112 where the data is picked from the second validation case study of the Ikea table assembly.

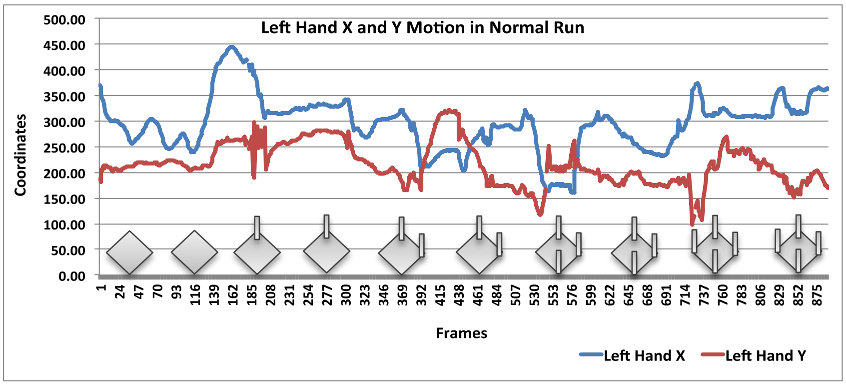


Figure 112: Motion chart of continuous human-workpiece interaction data

8.3.2 Mapping the skill-based behaviour to the framework

The above continuous data is broken down by the 'Segment' step into discrete human action states and workpiece progress states. The human skeletal motion during each action state is responsible for the following workpiece progress state. For example in (Figure 113: the Ikea table assembly task), the human action state 'H_NOOOOO_YBOOOO' is responsible for advancing the workpiece from state '$NOOOOO$' (sequence no. 1) to state '$YBOOOO$' (sequence no. 5), where the base of the table is placed on the workstation.

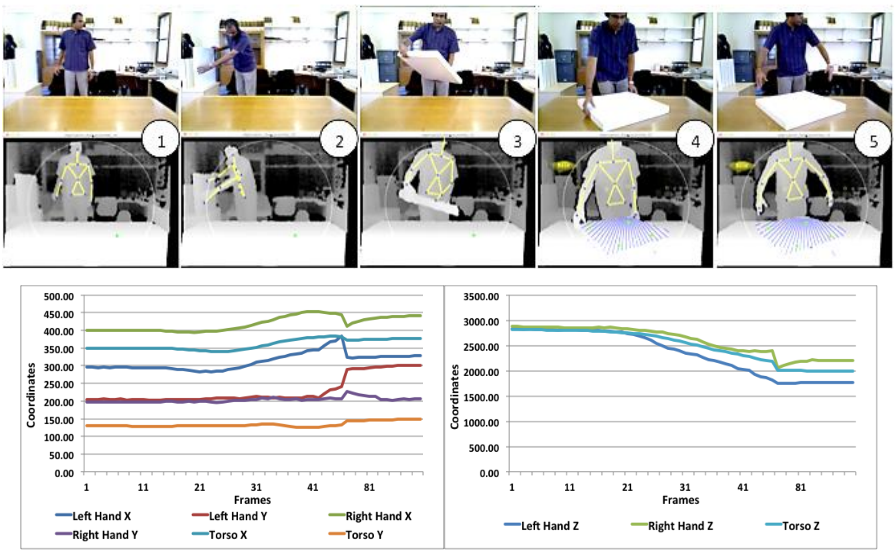


Figure 113: Human action state and its motion chart

The human activity within each action state occurs naturally due to the routine nature of the activity involved during the state without the human having to explicitly follow any instructions. This activity is subconsciously influenced by the affordance presented by the workpiece in the form of signals, such as in this case where the table base presents the 'grasp, manoeuvre and place' affordance to the human. The HWI theory classifies subconscious human activity as skill-based behaviour therefore the extraction of human activity data from within each action state in the 'Decode' step of the framework results in the digitisation of the skill-based task behaviour as shown in Figure 114.

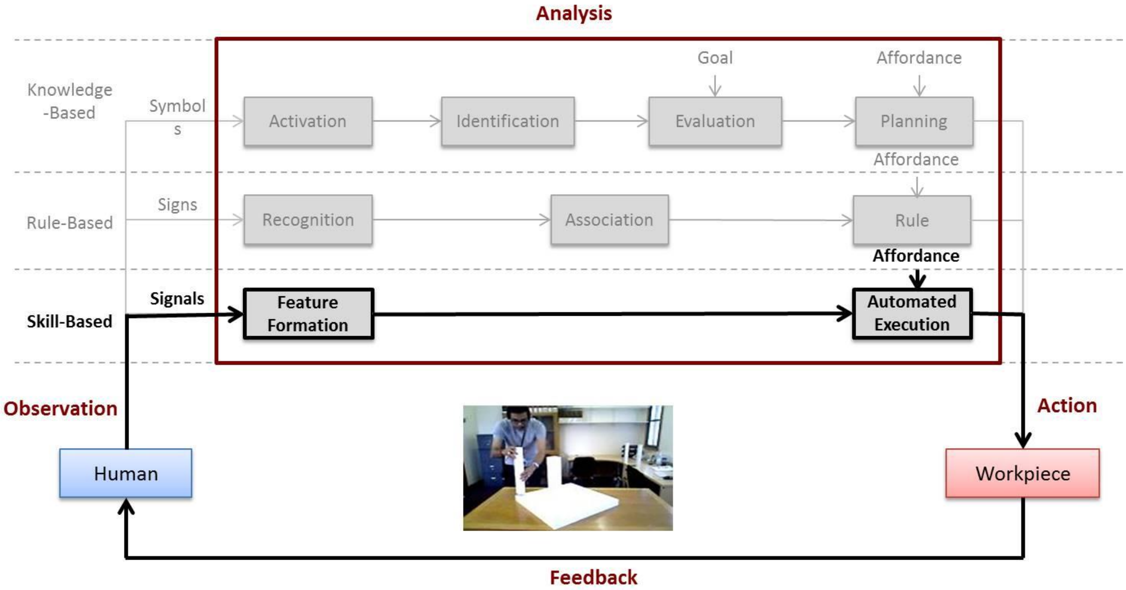


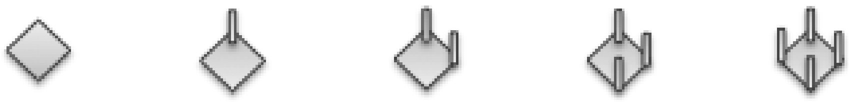
Figure 114: Digitisation of the skill-based human behaviour

8.3.3 Mapping the rule-based behaviour to the framework

The human action states and workpiece progress states are modelled to represent their co-relation using Hidden Markov Modelling (HMM) in the 'Model' step. Multiple observations of different scenarios of the same task result in multiple models representing that task. From these models, for any task scenario, the human action sequences that are most likely responsible for that scenario can be extracted by using the 'Extract' step.

Consider the task scenario used in case study 2 denoted by the following workpiece observation sequence:

1.	2.	3.	4.	5.	6.
NOOOOO	YBOOOO	YBLOOO	YBLLOO	YBLLLO	YBLLLL
No Workpiece					



The human action sequence responsible for the above task scenario as extracted from the task model is:

1.	2.	3.	5.	6.
H_NOOOOO_YBOOOO	H_YBOOOO_YBLOOO	H_YBLOOO_YBLLOO	H_YBLLOO_YBLLLO	H_YBLLLO_YBLLLL

In this task scenario, the workpiece is assembled according to a known sequence prescribed by the assembly rules. From the human action state sequence, it implies that the human has followed the assembly rules during the task in order to decide which action to perform when. The signs observed by the human during the assembly are the configurations of the workpiece before and after the application of every rule. These signs convey the workpiece affordance to the human based on which the human chooses which rule to apply when during the task. Therefore the extraction of human action sequence for a task scenario results in the digitisation of the rule-based behaviour of the human during the task as shown in Figure 115.

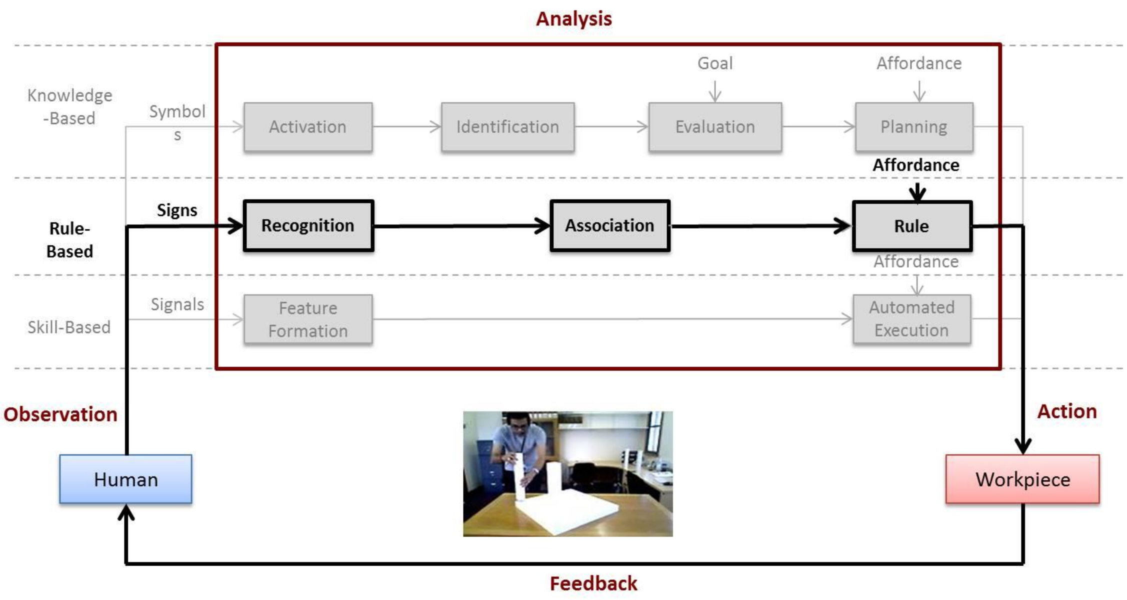


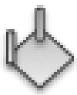
Figure 115: Digitisation of the rule-based human behaviour

8.3.4 Mapping the knowledge-based behaviour to the framework

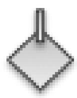
According to the HWI theory, the human displays knowledge-based behaviour when he is solving an unforeseen problem during a task. This behaviour is broken down into four stages, namely, activation, identification, evaluation and planning. The digitisation framework has a feature in the segmentation step wherein a problem-solving session within a task scenario can be identified and isolated for further analysis. This feature was used in the second validation case study where a problem-solving session was segregated from the human-workpiece interaction data. The human action and workpiece states within this session were modelled and the human action sequence responsible for problem solving was extracted. The workpiece progress sequence and the corresponding human action states extracted during the problem solving session are shown below.

Workpiece progress state sequence:

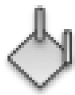
1.
Y_BLOOL
(Wrong workpiece state)



2.
YBLOOO
(Workpiece state restored to the state before the problem)



3.
YBLLOO
(Workpiece state corrected)



Corresponding human action state sequence:

1.
H_YBLOOL_YBLOOO
(Wrongly placed leg removed)

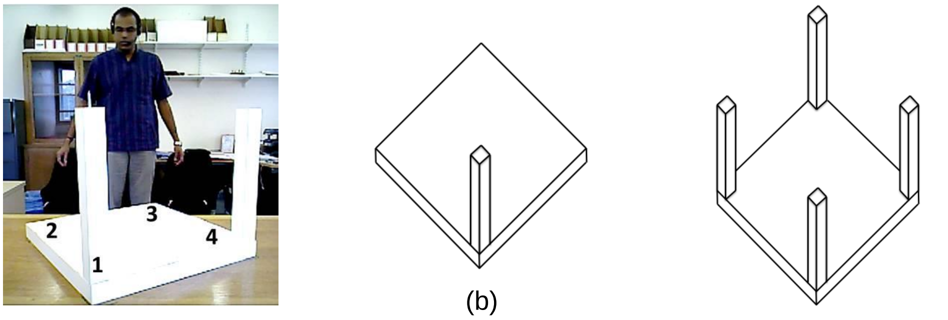
2.
H_YBLOOO
(Workpiece restored to pre-problem state)

3.
H_YBLOOO_YBLLOO
(Leg placed in the correct position)

The '*Activation*' stage of the problem-solving session occurs when the human sees a wrong workpiece configuration resulting from his/her previous action. In this example, the human action results in the workpiece state '$YBLOOL$' which is not the expected state '$YBLLOO$' as per the assembly sequence. The occurrence of the workpiece state '$YBLOOL$' therefore activates the human to respond to the problem.

In the '*Identification'* stage, the human looks for various symbols, in this case the placement of the legs of the workpiece, to detect the wrong state as compared to the expected state.

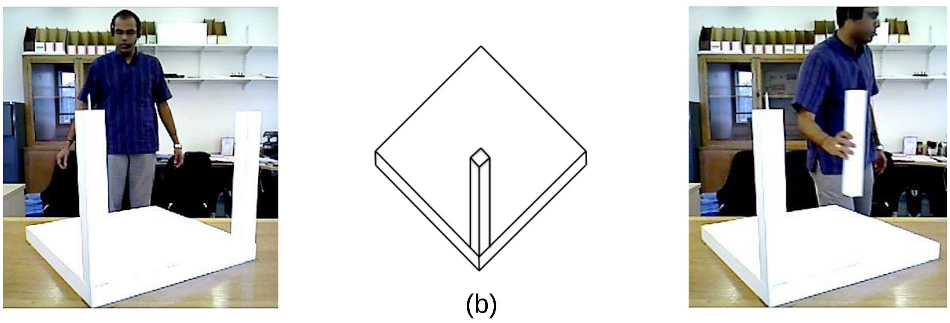
In the '*Evaluation'* stage, the human compares the choices he/she has to correct the problem and makes a choice based on his/her past experiences of solving similar problems and the overall task goal in mind. In the example shown in Figure 116, the human makes a choice of removing the leg placed in position 4 and placing it in position 2, on the basis of reaching the overall goal of completely assembling the table.



(a) (b) (c)

Figure 116: (a) Wrong workpiece configuration (b) Chosen workpiece configuration (c) Overall task goal

In the '*Planning'* stage, the human plans his next action by visualising the target state '$YBLOOO$' and the current affordance presented by the workpiece '$YBLOOL$' and executes the necessary action 'H_YBLOOL_YBLOOO' (Figure 117).



(a) (b) (c)

Figure 117: (a) Current workpiece affordance (b) target workpiece state (c) Execution of chosen action

Therefore, extraction of human action states responsible for solving a problem and extraction of human activity data from within these states results in the digitisation of the knowledge-based behaviour of the human in the task as shown in Figure 118.

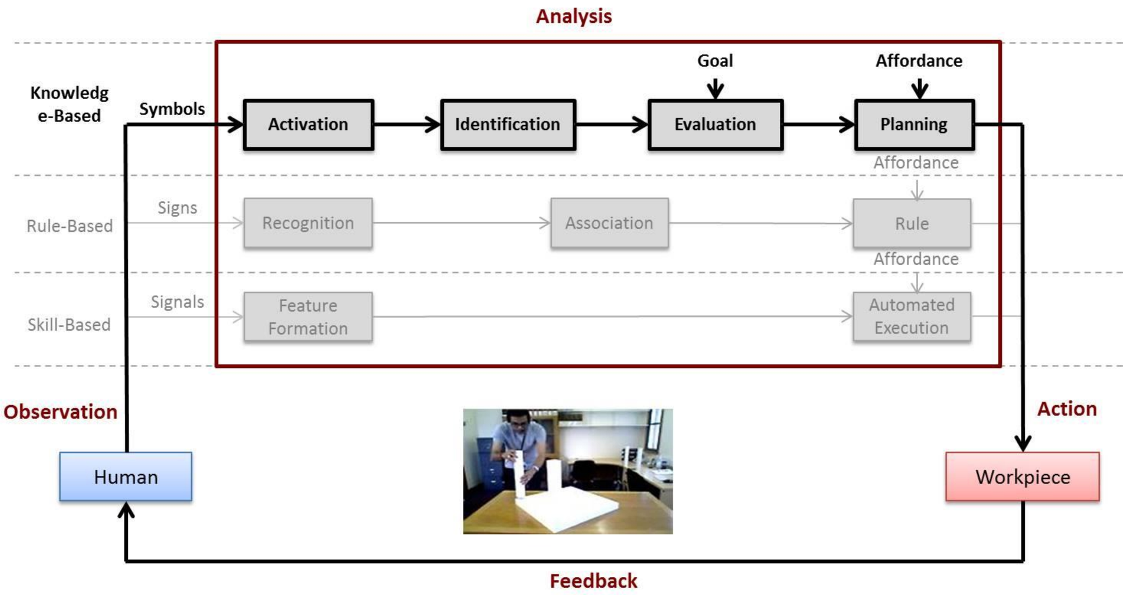


Figure 118: Digitisation of knowledge-based human behaviour

8.4 Mapping of hierarchical task analysis to the framework

In Hierarchical Task Analysis (HTA), the manual task is broken down into sub-tasks, each fulfilling a sub-ordinate goal but all contributing to fulfil the main task goal. The 'Segment' step of the framework breaks the manual manufacturing task down into sub-tasks (human action states) and the subsequent 'Model' step co-relates each action state to the corresponding workpiece progress state thereby mapping the action states to the sub-goals of the task. Using the 'Extract' step of the framework, the complete task execution strategy is extracted in the form of the action state sequence, which maps to the overall goal of the task.

8.5 Mapping of cognitive work analysis to the framework

The human uses his cognitive skills to solve unforeseen problems that occur during the task. The framework supports the segregation and modelling of the problem-solving scenarios from the task from which cognitive work analysis is enabled. It must be noted that not all problem-solving scenarios can be addressed because if any problem is not captured during the task observations, its solution cannot be segregated and modelled. However, it was shown during the validation case studies that the framework is able to extract human action sequences for not only observed task scenarios but also for unobserved task scenarios. Therefore, the framework is able to predict human action sequences for solutions to also those problems that have not been captured. This ability of the framework qualifies it to support cognitive work analysis and three of its five components, namely, strategy analysis (SA), cognitive task analysis (CTA) and worker competency analysis (WCA).

The SA and CTA are jointly performed when the human action states responsible for solving observed and unobserved problems are extracted and the human activity within those states is decoded using the framework. SA corresponds to the sequence of human actions that is planned and executed as a strategy to solve problems during a task whereas CTA corresponds to the control decisions taken by the human in between 2 action states depending on the current state of the workpiece against the target state of the workpiece both of which are captured in the human-workpiece interaction data.

WCA is performed when the human activity within each action state is studied. For each human performing the task, his/her action skills comprising body movements, workpiece manipulation techniques, and the speed and acceleration of hand gestures and reaction skills comprising decision-making to choose certain actions over others depending on the state of the workpiece can be extracted from the human activity data using the framework. Therefore, the competency of each worker performing the task can be analysed and represented using the skill models generated in the 'Model' step of the framework.

8.6 Advantages and limitations of the framework

Each of the 6-steps of the digitisation framework involves the use of innovative ICT methods to enable its functionality. The advantages that these methods bring to the framework are compelling but they also have certain limitations that curtail the full extent to which this framework can serve the digitisation needs of the manufacturing industry.

The 4 focal ICT methods used in the framework are:

1. Human skeletal motion tracking.
2. Object recognition and tracking.
3. Segmentation of continuous human-workpiece interaction data.
4. Modelling the human-workpiece interactions using Hidden Markov Modelling (HMM).

Each of these methods is discussed in detail below:

8.6.1 Human skeletal motion tracking

Human skeletal motion tracking is a major part of the ‘Capture’ step of the digitisation framework. A key hardware tool used in this step is a commodity gaming interface technology, such as the Microsoft Kinect. The Kinect is a depth imaging sensor and an RGB camera packed into a single device and comes with a software development kit with ready-to-use library of functions for human skeletal motion tracking. The commodity price point of less than £200 per unit, markerless and anonymous motion tracking and high portability makes it extremely attractive to use in this framework and on the actual shopfloors of the manufacturing industry. The primary advantage of using the Kinect™ is the easy availability of real-time 3D motion tracking of up to 20 human skeletal joints for up to 2 humans simultaneously per Kinect sensor.

However, there are 3 main limitations of using the Kinect™ for human skeletal motion tracking that negatively influence the implementation of the framework.

Noisy human motion tracking

Both the first and the second generation of the Kinect sensors (Kinect V1 and V2 respectively) are used in this research with the Kinect V2 introduced only in the last phase of research. Therefore, most of the work is done with the Kinect

V1 that has a resolution of 640 x 480 pixels for RGB imaging and 320 x 240 pixels for depth imaging. The human skeletal motion tracking function uses the depth image and matches the depth patterns within each image with human skeleton patterns to extract 3-dimensional positions of up to 20 skeletal joints per image. Since, these positions are captured for up to 30 image frames per second (fps), human motion can be tracked.

This positioning data for each human joint is affected by the low resolution of the depth image resulting in noisy motion capture data. This high frequency noise must be filtered in order to obtain reliable human motion tracking to correctly capture human actions during a manual-manufacturing task. An example of before and after filtering noise from right hand joint data is shown in Figure 119.

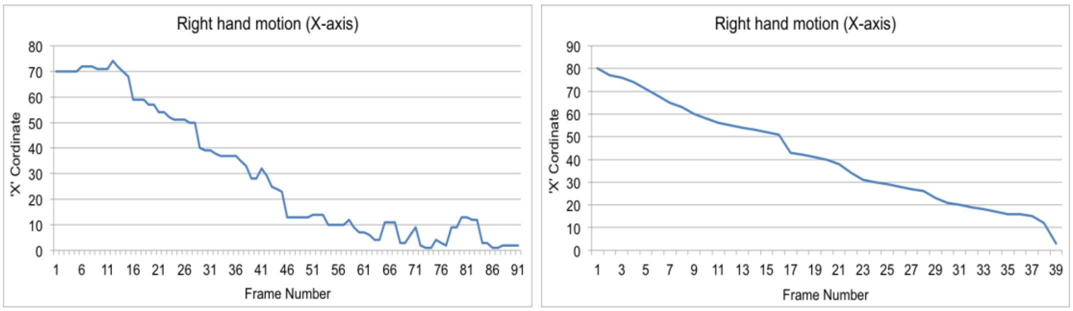
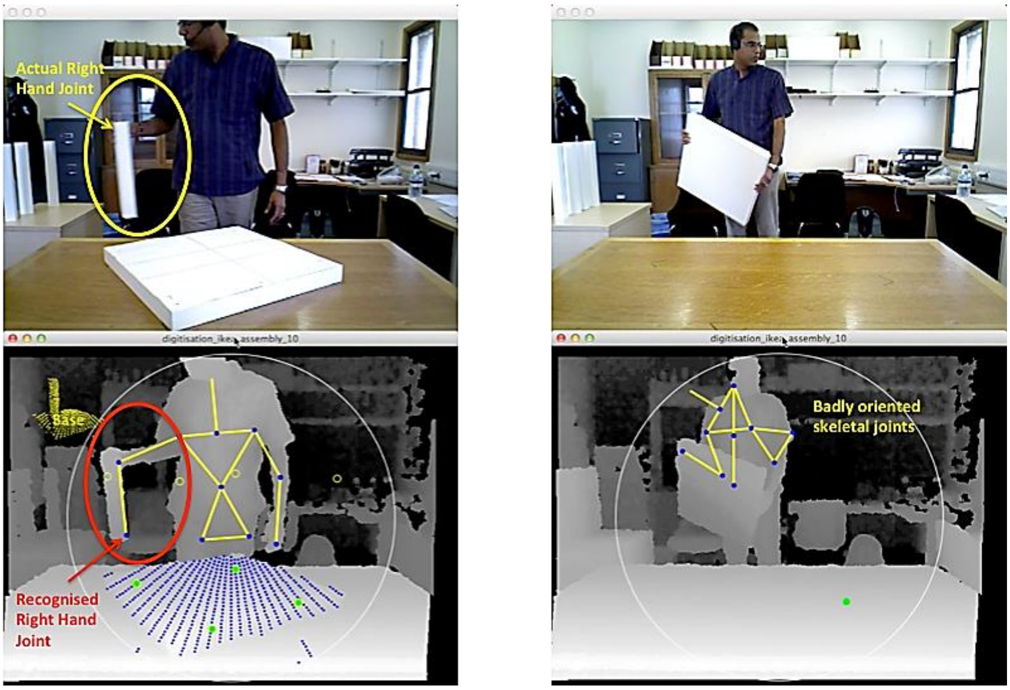


Figure 119: Before and after filtering high frequency noise (different task runs)

Incorrect identification of human skeleton

Since the human skeletal motion tracking function uses depth image patterns to recognise human skeleton, any objects handled by a human that resemble human body parts are sometimes recognised as human body parts. As a result, the skeletal joint position data for that part of the human body is incorrect. Although, this phenomenon is not common, it may pose a problem if it occurs during critical moments of the task. An example of incorrect human skeletal recognition and the resulting error in skeletal joint position data is shown in Figure 120a below.



(a) (b)

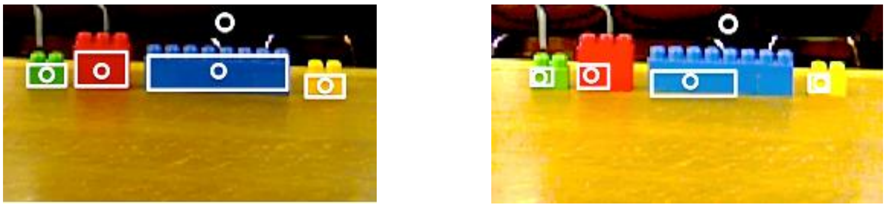
Figure 120: (a) Incorrect recognition of human arm (b) Unreliable skeletal tracking due to a large occlusion

Occlusions affecting human skeletal tracking reliability

Any large object that comes in the way of the Kinect getting an unobstructed view of the human body, also known as occlusion, can hamper the reliability of skeletal tracking data (Figure 120b). The loss of reliability or even the loss of skeletal tracking increases as the size of the occlusion increases. This disadvantage limits the size of the workpiece used in task in order to successfully implement the framework. However, the Kinect is able to provide skeletal tracking even if the entire lower body is occluded from its view, which is demonstrated for all the tasks digitised in this research.

8.6.2 Object recognition and tracking

With the availability of both depth and RGB images at the rate of 30fps providing 3-dimensional position and colour information per image pixel, the Kinect is also used to recognise and track objects in the framework. Before the availability of depth images, even simple object recognition needed complex image processing. The additional depth information provides an additional dimension of detail to be able to distinguish between object surfaces, edges, and detect changes in object surfaces and edges. When depth information about certain object features is not reliably available due to the low resolution of depth images, colour information about the pixels associated with the object is be used to complement the depth information. However, colour recognition suffers from heavy dependency on ambient lighting conditions and any change in these conditions causes the object recognition and tracking function to fail. Therefore, in all the implementations of the framework except in the case of digitising the task environment, the ambient lighting conditions were controlled. The following (Figure 121) shows an example of how lighting conditions can affect object recognition.



(a) (b)

Figure 121: Object recognition result in (a) normal lighting condition (b) brightened condition

8.6.3 Introduction of Kinect V2 into the research

The second generation of the Kinect sensor, hereafter called as the ‘Kinect V2’, and the associated Software Development Kit (SDK) were launched in October 2014 with promising upgrades in depth and RGB imaging, and human skeletal motion tracking quality. These upgrades compelled the author to consider using the Kinect V2 as a possible replacement for Kinect V1 in the digitisation framework. Though a complete comparative study between Kinect V1 and V2 has been conducted and reported by Zennaro (2014) for computer vision applications in robotics, the Kinect V2 was technically evaluated from the point of view of its use in the proposed digitisation framework as a task capture tool as an alternative to Kinect V1 and therefore the two Kinect versions are compared in a 3-part study. In the first part, the specifications and features of the sensors are compared from literature, in the second part the two sensors are used to capture a set of tasks and their performance in skeletal motion tracking is evaluated and compared and in the third part, the object recognition capability of the sensors is compared. The detailed comparison study is presented in Appendix D.

Verdict

From the skeletal tracking and object recognition results, it can be deduced that the Kinect V2 is a better sensor than Kinect V1, in terms of accuracy, precision and reliability. Therefore, the Kinect V2 emerges as a candidate to replace Kinect V1 in the digitisation framework from this point onwards in research.

However, it must also be noted that in all the case studies so far, there is a need to capture the depth and RGB values of the same pixel belonging to a particular object in the 3D scene for object recognition and tracking. This need was easily met with the Kinect V1 because the pixels in the depth image (320 x 240) were easily mapped to the colour image (640 x 480) by a factor of 2 helped by the almost same fields of view of the colour and depth cameras. This pixel mapping is a challenge in the case of the Kinect V2 because the pixels in the colour image (1920 x 1080) do not directly map to those in the depth image (512 x 424) because of the different fields of view of the colour and depth cameras. A separate calibration step is needed to perform this mapping and the additional complexity brought in by the computer graphics techniques involved in performing the calibration meant that the mapping was left out of the scope of this research. It can also be observed from the comparison results that for the Kinect V2, the improvement in skeletal tracking performance is much greater than the improvement in object recognition performance over Kinect V1. Therefore, a decision was made to use the Kinect V2 for its improved skeletal tracking capability and the Kinect V1 for its easy object recognition capability, simultaneously operating the two sensors in the task capture method used in this research from this point onwards. In this way, the advantages of both the sensors are exploited to the best possible extent.

The Kinect V2 was therefore introduced in the framework and the implementation of the ‘Capture’ step of the framework for the Ikea table assembly case study was repeated. Kinect V2 replaced Kinect V1 as the human motion capture tool while Kinect V1 was retained for workpiece tracking. The same setup was used for the composite layup case study. The use of Kinect V2 in the Ikea table assembly case study is presented in Appendix E.

8.6.4 Segmentation of continuous human-workpiece interaction data

According to the proposed digitisation framework, segmentation of human-workpiece interaction data can be performed using 3 methods, namely, time-change based, trajectory-change based and workpiece-change based depending on whether time, human action or workpiece change is the dominating factor in the task.

Time-based segmentation is used when human actions and effects on those actions are gradual in nature and no sudden changes to either during the duration of the workpiece. In this case, the human-workpiece interaction data is divided into segments of ‘n’ time units each. The point to note in this type of segmentation is that should there be an abrupt change in either human action or workpiece state, that change may be subsumed within one state if it occurs in that state’s time period rather than a separate state being created for that sudden change. Creation of a separate state would have made the sudden change detectable and the human actions that led to that change more conspicuous at the time of analysis. Another disadvantage of time-based segmentation is that if no workpiece change happens during a particular time period (task idling), the states generated during this period show no change in the workpiece configurations. This results in duplicate states that do not add any value to the subsequent modelling and knowledge extraction steps of the framework.

Trajectory-change based segmentation works by dividing the human-workpiece interaction data at locations where there are significantly big and or sudden changes in action among the otherwise smooth human actions regardless of whether those changes in action had any effect on the workpiece or not. A threshold to define the level of change in trajectory that warrants segmentation is needed in this method. Using this segmentation method ensures that for all major actions, their individual states are created and changes to the workpiece as a result of big and sudden human actions can be analysed from within the states. However, this segmentation technique is affected by noise in the human motion capture data because any peak in the data that does not represent any change in action will be considered for creation of a state. These states that do not indicate any progress in the task unnecessarily increase the complexity of the modelling and knowledge extraction steps subsequently in the framework.

Workpiece-change based segmentation divides human-workpiece interaction data at instances where the workpiece has undergone prominent changes in state regardless of whether there was significant human action preceding that state or not. A threshold to define the level of prominence in workpiece change that warrants segmentation is needed in this method. This method is the best suited for the framework because the basis of the framework is to determine which human actions states cause what effect on the workpiece during the duration of the task. This segmentation method does not get affected by human motion noise or task idling because states are created on the basis of prominent workpiece changes.

Both the trajectory-change and workpiece-change based segmentation methods suffer from heavy dependency on the threshold values that define the level of change. A smaller than optimum threshold value would result in creation of trivial states that are of no use to the modelling and knowledge extraction steps and a larger than optimum threshold value would result in missing states or merging of two important states into one, thereby losing significant information about the task that could have been extracted. Determining the optimum threshold value thus is a matter of involving the task experts in the segmentation process and using their inputs to either include missed states or exclude trivial states manually from the subsequent modelling and knowledge extraction steps of the framework.

8.6.5 Modelling the human-workpiece interactions using Hidden Markov Modelling

The human-workpiece interactions captured during the manual-manufacturing task are modelled using Hidden Markov Modelling (HMM). HMM is used because there is an observable aspect of manual manufacturing, which are the changes to the workpiece during the task and there is a hidden aspect, which is the human skill that is embedded within the actions during the task. HMM is a widely used tool to analyse and predict time series phenomena such as speech recognition from continuous voice data. A hidden Markov model very closely represents the proposed human-workpiece interaction concept because the Markov assumption in an HMM states that any change in observable states is only due to past hidden states, which is also the concept of the human-workpiece interactions in a manual manufacturing task.

i. Order of the Markov process

In this framework, 1^{st} order Markov process is assumed, which states that the next workpiece state depends only on the previous human action state and none before that. Therefore, a change to the workpiece at any instance is only due to the human action immediately preceding that instance. In most manual manufacturing situations this assumption holds true but in cases where cumulative effects of several past human actions are evident on the workpiece, the 1^{st} order Markov process assumption is not valid. An example of such a case is the polishing task where the heat build-up over multiple polishing passes affects the surface of the workpiece cumulatively or a composite layup task in which effects of plies laid several steps back can affect the layup of the next ply. This change in workpiece cannot be attributed to only the last polishing or ply layup pass but to multiple previous passes. Such tasks cannot be effectively modelled using 1^{st} order Markov assumption made in the framework and the selection of the Markov process order would depend on the manufacturing task being studied.

ii. Presence of zeros in an HMM model

There are many examples where the probability of transition from one particular state to another is zero because that occurrence is physically impossible. For example, in the second validation case study, the human action state 'H_YBLLOO' cannot transition to 'H_YBOOOO' because the task does not allow handling of two workpiece components at the same time. In such a situation, the probability of state transition from 'H_YBLLOO' to 'H_YBOOOO' is zero. However, if such zeros are entered into the HMM model, the forward and backward algorithms for HMM evaluation, the Viterbi algorithm for HMM decoding and the Baum Welch algorithm for HMM training may give erroneous results due to mathematical underflow errors.

In order to avoid this problem, the 'Absolute Discounting' technique is used where a small probability factor *f* is discounted (subtracted) from all states that are assigned a non-zero probability. The probability factor is then distributed equally among states that are assigned zero probability so as to maintain the sum of probabilities to 1. An example of 'Absolute Discounting' is shown below:

Before discounting:

State	A	B	C	D	E
Probability	0	0.2	0	0	0.8

$f = (p(B) + p(E)) / 100 = 0.01$

$p(B) = p(B) - f = 0.19 \; and \; p(E) = p(E) - f = 0.79$

All other states are assigned the probability f. Therefore, the resulting HMM A' contains all non-zero probabilities thereby preventing underflow errors while computing for HMM evaluation, decoding and training. This method does not affect the HMM greatly because the value of factor *f* is significantly lower than the non-zero probability values.

After discounting:

State	A	B	C	D	E
Probability	0.01	0.19	0.01	0.01	0.79

iii. Sparse observation data

Not all task scenarios can be observed by the framework and this depends on how the number of instances of the task captured. This results in sparse observation data in which the probabilities of occurrence of certain state transitions and observations in an HMM are assigned to zero causing mathematical underflow errors as explained earlier. Literature has revealed that the most common way of solving this issue is by using smoothing techniques such as Absolute Discounting, Laplace smoothing, Good-Turing estimation and Shrinkage. A more practical approach is used in this framework in which the task experts assign non-zero state transition and observation probabilities, including for those that are not observed. For those state transitions and observations that are not possible in the task, the Absolute Discounting technique is used to avoid zero probabilities. The resulting HMM is then optimised using the Baum-Welch algorithm.

iv. Long observation sequences

The HMM evaluation, decoding and training algorithms work well for HMM models with a short sequence of states. This is because many mathematical quantities that are generated at intermediate steps of the algorithms would quickly get extremely small as the sequence gets longer, resulting in underflow errors. There are generally two ways to deal with the problem in the literature. In the first method, the entire HMM model can be represented in the logarithm domain in which the product of small quantities is a sum of logarithms of those quantities thereby avoiding extremely small numbers. The second method is to use scaling in all the HMM algorithms. In this step, the mathematical quantities computed at each stage of the algorithms are scaled by a common factor to avoid pushing the quantities towards zero. This way the quantities never get small enough to cause underflow errors. In this framework, the scaling method is used in which at the end of each computing stage, a factor of 10 was used to scale all the probabilities of the HMM.

v. Improving the HMM parameter estimation process

In this framework, the HMM parameters are assigned based on inputs from the task expert. However, these parameters may not be optimum every time and without optimisation the HMM models will not produce correct results. Since the extraction and decoding of manufacturing knowledge depends on the correctness of the HMM model, its parameters must be optimised by training the HMM models.

After comparing the two most commonly used HMM optimisation algorithms, namely, the Viterbi Training algorithm and the Baum Welch algorithm, the latter was selected in this framework as explained in section 5.3. This algorithm uses an observed workpiece state sequence and trains the HMM model by determining the optimum parameters. However, the Baum Welch algorithm converges to a local optimum, which means that for every change in the initial parameters, the optimised parameters will be different even for the same observed sequence.

The local optimum may work in certain situations and may not work in others. For example, in validation case study 2, for the workpiece observation sequence$O_Q = \{YBLOOL,\ YBLOOO,\ YBLLOO,\ YBLLLO,\ YBLLLO,\ YBLLLL\}$, the Baum Welch optimised HMM model produces

$$H_Q = \{H\_YBLOOL\_YBLOOO,\ H\_YBLOOO,\ H\_YBLOOO\_YBLLOO,$$
$$H\_YBLLOO\_YBLLLO,\ H\_YBLLLO\_YBLLLL\}$$

as the most likely human action state sequence responsible for the workpiece observation sequence. In this case, the human action sequence generated by the optimised HMM is correct.

However, for the Lego block assembly example, for the workpiece observation sequence:

$O_Q = \{WP\_B,\ WP\_BG,\ WP\_B,\ WP\_BG,\ WP\_BGR,\ WP\_BGRY,\ WP\_BGR,$

$WP\_BG,\ WP\_B,\ WP\_BR,\ WP\_BRG,\ WP\_BRGY\}$

the optimised HMM produces the human action sequence

$H_Q\ = \{H\_A\_B,\ H\_A\_G,\ H\_D\_G,\ H\_D\_R,\ H\_A\_R,\ H\_D\_Y,\ H\_D\_R,\ H\_D\_G,$

$H\_A\_B,\ H\_A\_R,\ H\_A\_G,\ H\_A\_Y\}$

This output action sequence is incorrect. The correct output sequence should have been

$H_Q\ = \{H\_A\_B,\ H\_A\_G,\ H\_D\_G,\ H\_A\_G,\ H\_A\_R,\ H\_A\_Y,\ H\_D\_Y,\ H\_D\_R,$

$H\_D\_G,\ H\_A\_R,\ H\_A\_G,\ H\_A\_Y\}$

This example illustrates that Baum Welch algorithm is not sufficient to achieve fully optimised HMM models.

Therefore, other means of optimising HMM parameters will have to be investigated. In the literature, the use of Artificial Neural Networks (ANN) to transform the input observation sequence of the HMM so that this transformed sequence better suites the HMM is reported by Bengio et al. (1992). They used the ANN to transform the actual observations and then the transformed observations were fed into the HMM as an input vector. After optimizing the HMM for the transformed observations using the Baum Welch algorithm, significant improvements were achieved. Hassan et al. (2007) have furthered this research by using Genetic Algorithms (GA) to optimise the initial parameters of an HMM whose input observation sequence has been transformed using ANN and demonstrated even better results. The use of ANN and/or GA to optimise HMM models are outside the scope of this study and therefore are identified as future research.

8.6.6 Reproduction of extracted manufacturing knowledge

The ‘Reproduce’ step of the framework provides a platform for reproduction of the manufacturing knowledge extracted and decoded by the framework and for skill transfer. Going beyond exhibiting the extracted task strategy and animating the task with the extracted knowledge constituents augmented within the animation, is not within the scope of this study. The more effective and attractive methods such as the use of virtual reality (VR) for immersive experience of the task or the use of augmented reality (AR) to overlay task knowledge onto real task environments have not been investigated and have been identified as future research.

8.7 Generality of the framework

The digitisation framework is built upon a strong theoretical foundation of representing a manual manufacturing task as a series of human-workpiece interactions which when captured and modelled can enable the extraction, decoding and reproduction, i.e. digitisation of the manufacturing knowledge embedded within the task. This theoretical underpinning isolates the framework from the specifics of the task to be digitised or the end-user application.

The structure of the digitisation framework was designed by categorising the work of other researchers in literature in the area of human skill acquisition into functional research units. It was observed that these units corresponded to the standard informatics steps of data input, data processing, data analysis and data output in that order. Therefore, the digitisation framework was structured into 6 sequential steps that matched the standard informatics steps thereby allowing the framework structure to be fairly generic to accommodate the requirements of digitising most manual manufacturing tasks.

The methods and tools proposed in the framework, such as the Kinect sensors to capture the task or the several object recognition techniques to track workpiece progress during the task, are specific to both the type of tasks that were digitised as well as the era in which this research is carried out. These methods and tools were used off-the-shelf and plugged into the framework without the need to alter the design or the structure of the framework. For example, when the need to use Kinect V2 arose, it was just plugged into the framework to work alongside Kinect V1 without changing the framework and object detection techniques kept changing for different tasks as the workpieces changed but the framework remained the same.

Finally, the tasks chosen for implementing and validating the framework represented different task complexities, people, environments and constraints. The tasks ranged from simple Lego blocks assembly in a controlled environment to the complex composite layups in the clean room and the workpieces ranged from simple geometrical shapes as in the Ikea table components to the deformable pre-impregnated composite plies and the framework was able to accommodate all these requirements further reinforcing its generality.

8.8 Potential applications of the framework

It has been demonstrated that the framework at its fullest extent can be used to capture, extract, decode and reproduce manufacturing knowledge, especially the tacit knowledge such as human skills that is embedded within manual manufacturing tasks. At the very least, some components of the framework can be used independently for example, the ‘Capture’ step can be used to record a task into a video with the human actions and workpiece progress are annotated within the video for task demonstrations. A few application areas for the industry have been identified and briefly explained below.

1. Skill transfer platform

The main aim of the framework is to acquire skills from a human performing a task and digitise it in transferrable form such as in skill models. Therefore, an obvious application area is to build a skill transfer platform based on the framework which will enable a company to acquire skills from its senior experts in a non-obtrusive manner, archive them into skill models and reproduce them in a manner that is most appropriate for skill training. Effective skill training methods use multiple media such as print, videos, animations, immersive game-based training and training on the job by augmenting training data on the real-world task environment.

Currently, video demonstration is a popular choice for skill training and is an obvious comparison candidate for the task knowledge digitisation framework proposed in this research. Though a video provides an easy means to record and playback manual tasks for a leaner's viewing, it does not provide a rich medium to effectively absorb the key learning points. A video reproduction is only a 2-dimensional illustration in which a critical 3rd (depth) dimension of the task is lost and needs multiple recordings from multiple viewing perspectives to fill the dimension gaps. Also, a video demonstration does not explicitly illustrate the motor and control skills used by the human during the task leaving this important knowledge constituent to the individual interpretation of the learner. Finally, a video demonstration does not provide the means to obtain real-time feedback on the performance of the learner while performing the taught task thereby making it difficult to evaluate the efficacy of the skill training programme and compare the skill proficiency of the learner before and after skill training.

A senior representative from Airbus commented "The skill transfer platform enabled by Cranfield's digitisation framework will help us tap into the years and years of training and experience of our senior technicians before we lose those skills forever when the technicians retire".

2. Real-time ergonomic evaluation

The framework is effective in capturing human actions during a manual task and compute the mechanics of body motion such as body postures, angles between different parts of the body that move during the task, speed and acceleration of those movements and orientation of the body with respect to the workpiece being handled. In many instances, human actions may not be ergonomically correct and may lead to musculoskeletal disorders among the workforce over prolonged use of the wrong actions. A real-time ergonomic evaluation platform based on the framework can be used on factory shopfloors to continuously monitor human activity and raise alarms whenever bad ergonomics is detected in addition to reporting them. The platform can thus enable factories to design ergonomically correct workstations to ensure the wellbeing of their workforce.

An engineering manager from Rolls Royce (Marine) commented "confined workspaces such as in submarines compel people to work with bad postures and perform complex manoeuvring of components in cramped areas. A real-time ergonomics platform will enable us to evaluate the ergonomics of tasks in submarine environments and provide vital information to help us design better workspaces that minimise the ill-effects of working in confined spaces”.

3. Real-time remote collaboration platform

Manufacturing is an increasingly global activity with distributed sites for product design, verification, production and customer support. Therefore, geographically dispersed engineering teams are collaborating with each other at a scale never seen before. Commercial unified communication tools such as Skype™ and Webex™ are commonly used but the extent to which teams can use these tools to collaborate and solve a common engineering problem remains limited. These tools only enable the exchange of voice, text, files for communication and a shared whiteboard for collaboration. They do not allow for sharing physical engineering contexts or support collaborative working on engineering workpieces, which are key requirements for engineering-related collaborations.

The framework captures human actions and resulting workpiece changes during the task along with its environment and digitises this information into simple numeric data forms. Therefore the framework digitises the activities within a task while capturing the task context. If tasks occurring at two different sites could be digitised in this manner and the digital task data that is generated could be exchanged across these sites and reproduced in real-time, then it becomes possible for remote teams to collaborate with each other during the task, work on common workpieces and solve a common problem. Since the task data is low-definition, its synchronous exchange does not depend on the network bandwidth and global collaboration becomes truly possible.

A technical services manager from Jaguar Land Rover commented “with increasing global spread of our dealerships, it becomes necessary for our experts to deliver technical support remotely to the dealer technicians to solve unforeseen problems with our customers’ cars as soon as possible. A real-time remote collaboration platform will help us react quickly to technical support requests and significantly reduce the need for our experts to travel”.

A real-time remote collaboration platform is being developed as part of an Innovate UK funded project and the initial results of this work are presented in a conference paper (See ‘List of Publications’).

4. Intelligent automation of complex manual manufacturing tasks

Complex manual manufacturing tasks rely heavily on human intellect and skills, acquired with training and years of experience performing the task. The composite layup is an example of such a task in which hard to manipulate pre-impregnated composite plies are laid over moulds with complex geometries. Experienced layup technicians can plan the layup strategy based on the mould geometry and on the understanding of the deformation characteristics of the composite material, execute the plan successfully by using appropriate ply manipulation techniques suited for different areas of the mould to avoid any undesirable deformities in the ply such as wrinkles and bridges and rectify any deformities that are unavoidable.

In order to automate such a task, the automation solution must be able to learn from the experienced technician in task approach, layup techniques and problem solving. Such a solution needs the intelligence to devise a task strategy by observing the mould geometry and predicting the layup techniques required for the different areas of the mould. It must also be dexterous enough to execute the techniques correctly by manipulating the ply just as how a human technician would. The solution must also keep track of the progress of the layup on the mould, identify problems and then have an approach to solve them.

Before developing the automation solution, the task performed by the experienced technician must be systematically understood, incorporating all possible scenarios, whether or not those scenarios have been observed. The framework has the ability to understand the task by capturing the human-workpiece interactions involved in multiple task scenarios and even predicting such interactions for scenarios that have not been observed. Since all this knowledge is digitised, it can be used to inform the intelligence behind an automation solution in the form of expert systems.

The framework is also able to extract and decode the complex human actions involved in the task such as hand gestures, gesture speed, body postures, body orientations, etc. These actions could now be incorporated into the kinematic control of robot arms that have similar dexterity to human hands. The only knowledge constituent that the framework does not extract is the force used by the human hands while performing the task. Means of acquiring this knowledge is a future topic of research because existing methods such as getting the human to wear force-capture gloves while performing the task are obstructive and not practical.

Finally, it has been demonstrated that the framework is also able to digitise task environments by capturing manufacturing environment and operational data that is critical to the task. In the wheel loading case (section 6.5), the framework was able to provide motion data of the moving wheel hub and misalignment data between the moving wheel hub and the to-be-assembled wheel so that the automation solution could make decisions on when and how to load, continue loading or abort and if and how to correct misalignment in real-time. Therefore, the framework proposed methods to replace human senses in the monitoring of the task environment.

Therefore, by having the human-like expert system control the human-like robotic solution in a task environment that could be continuously monitored by a human-like sensing system, there is a potential for intelligent solutions to be developed for automating complex manual manufacturing tasks.

8.9 Future Work

Future research work has been identified by considering the limitations of this research and suggesting possible ways of addressing them as described below.

Further validation in real manufacturing shopfloor conditions

The framework has been currently tested with about 4 case studies with varying task complexity, constraints, people and environments including one real-world manufacturing task example. According to the author, these case studies are representative of a large number of manual manufacturing tasks in the industry.

However, all the above tasks were conducted in a fairly controlled task environment and the issues that are likely to occur while digitising tasks on real shopfloors are not known. These issues are related to the positioning of the Kinects on the shopfloor, presence of occlusions in the form of machines, gantries, etc., IR or other forms of interference from the machines, presence of dust and oil mists that might affect the Kinect cameras, ethical matters, psychological matters and health and safety matters among others. Therefore, there is a need to exhaustively test the framework in real shopfloor conditions and also conduct research on human psychology to assess human acceptance and adaptation to the framework.

Alternative technologies to complement or even replace the Kinect sensor

The research in its current form has sufficiently addressed the objectives of developing and implementing the framework for the capture and digitisation of manual manufacturing task knowledge. However, certain methods used in this framework will need to be upgraded to achieve better results. One such method is human motion capture and object recognition that records human-workpiece interactions during the task. The first generation of the Kinect sensor (Kinect V1) did not produce reliable results under certain conditions as explained in section 8.6.1 and therefore the second generation of the sensor (Kinect V2) was investigated and used in the 'Capture' step of the framework. However, due to the complexity involved in mapping the depth and colour image pixels, Kinect V2 was not used for workpiece progress tracking even though it was preferred over the Kinect V1 and this constraint compelled the use of both Kinect V1 and V2 in the framework. However as Kinect V1 is already an obsolete product, more work is needed in the area of depth to colour image mapping thereby eliminating the need to use Kinect V1.

Another major issue is the unreliable skeletal tracking by the Kinect in the presence of large occlusions that make certain parts of the human not visible to the sensor. This issue remains to be a concern even with the Kinect V2 because both the sensors need a clear line-of-sight to the human without any obstructions. Therefore, a completely different method such as marker-based or accelerometer-based motion tracking would need to be used to complement the Kinect. The commodity price point factor must be taken into account while investigating other methods because affordability is a big plus point for the industry to adopt the framework.

Enhance the robustness of human-workpiece interaction modelling

The ‘Model’ step is another target for improvement in this research as it stands currently. The issues of zero probability values and the models unable to cater to long task observation sequences will need to be tackled using successfully tried and reported methods from the literature such as Laplace Smoothing and model initialisation using evolutionary algorithms. HMM model optimisation method currently used in this research does not produce sufficiently robust results all the time because the adopted Baum-Welch training algorithm converges only to the local optimum when global optimum is desired. Therefore, other proven optimisation techniques such as genetic algorithms and artificial neural networks can be considered or even a combination of evolutionary algorithms and the Baum-welch algorithm can be tried.

Address other important manufacturing knowledge constituents

The framework currently is not able to extract and decode task knowledge constituents such as the force applied by the human while manipulating the workpiece, lower body weight distribution or the eye movements made during the task. These are important constituents because tactile feedback from the workpiece is equally important as visual feedback. Moreover, the pressure applied by the human feet on the floor while performing the task also reveals important insights into body posture and weight distribution among the legs that have a bearing on the force applied on the workpiece. Therefore unobtrusive methods that capture human force applied on the workpiece and on the floor must be used. Eye movement of the human during a task is an important constituent because it can tell where the human vision is focussed while performing critical parts of the task. Though the framework measures the head bending or glance angle, the actual glance direction obtained by tracking eye movement is not captured. The sound generated during a task, such as machining, provides important insights into the progress and quality of the task. The Kinect sensor used in this research has the capability to capture audio but it is only used in this research to capture human verbal commands. In the future, the Kinect sensor could be used to capture machine sounds as well to add the audible dimension of the task to the extracted manufacturing knowledge constituents.

Another task knowledge constituent that can be captured is the influence of the human being left-handed, right-handed or ambidextrous on the task. To obtain this knowledge, the motion data of the human hands can be mapped on to the workpiece geometry to extract and visualise motion patterns that vary between people of different dexterities. Capturing this knowledge is important when transferring skills from a right-handed expert to a left handed novice because the difference in dexterity can be compensated by the framework to negate any disadvantages that the left-handed novice might face.

Finally, this research captures and extracts only the knowledge associated with a manufacturing task but does not extract the quality factor associated with this knowledge. Quality factor is an important constituent because it could provide a single unified index to evaluate and compare human skills between people of varying builds, body proportions, gender, dexterity and cultural backgrounds. Quality factor is envisaged as a function of key task characteristics such as human motion accuracy vis-à-vis a standard motion pattern for a particular task, precision to record performance consistency, workpiece progress indicator, task result indicator and the time taken per sub-task. This research could investigate the development of such a quality factor for manual manufacturing tasks.

Scaling up for digitisation of tasks with higher cognitive human inputs

The framework in its present form is not able to digitise task knowledge from complex manual manufacturing tasks in which human cognitive inputs overshadow the physical inputs. An example of such a task is the manual assembly of an aircraft engine, in which human must identify the positioning, orientation and mating requirements of assembling intricate parts on to complex engine sub-assemblies whereas the actual assembly actions are rather simple.

In such a task, sophisticated human cognition to identify workpiece features as well as sub-conscious decision making to constantly align and position the workpieces for assembly far outweigh the motor skills needed in the task. Therefore, motion sensing and simple object recognition will not be adequate to extract and decode manufacturing knowledge. In such cases, qualitative methods of obtaining task data such as interviews, questionnaires, task walk-throughs and what-if analysis of known and unforeseen task situations will have to be introduced into the framework. Existing methods used in hierarchical task analysis and cognitive work analysis could be explored for this purpose.

New interface technologies to accommodate the advances in manufacturing technology

New human-machine interface technologies are continuously being developed and introduced at a rapid pace due to the flourishing gaming industry. A constant eye must be kept on the market and on the literature for new motion capture technologies such as 3D wireless accelerometer-based motion sensing at commodity prices, force sensing technologies such as portable and wireless electromyography (EMG), floor pressure sensing technologies for monitoring the pressure applied by human feet on the floor and the weight distribution between the feet, eye tracking technologies as well as new and better human action recognition and object recognition algorithms. The generic nature of the framework allows the plugging in and out of these new technologies to improve its digitisation capability for manufacturing tasks of the future.

8.10 Conclusions

The manufacturing industry, especially in high wage economies, is grappling with issues of global skill shortage and intense competition. Therefore, sustaining slow and costly manual manufacturing operations has become very difficult threatening to reduce the global competitiveness of manufacturing enterprises. The two ways in which manufacturing enterprises can remain globally competitive is by adopting quick and cost-effective skill transfer programmes to rapidly up-skill their workforce in the short term and by adopting intelligent solutions to automate complex manual tasks and use the human workforce in a higher value-added capacity in the long term. This research has the potential to enable both these solutions for the manufacturing industry.

The aim of this research is to develop a framework for digitisation of manual manufacturing task knowledge. It aims to achieve the digitisation by using consumer-grade gaming interface sensors to make the framework cost-effective and technologically advanced at the same time. The framework was developed with a strong underpinning of human-workpiece interaction theory reinforced by seminal research in human behavioural analysis by eminent researchers such as Rasmussen and Gibson. The 6 steps of the framework were designed by studying the flow of informatics process and data adopted by researchers in the human skill acquisition and knowledge capture domains. The methods and tools used in each of the 6 steps were chosen from off the shelf, specifically selecting those that were proven and were cost-effective. For example, gaming interface sensors, such as the Microsoft Kinect™, were chosen for their low-cost, consumer-grade robustness, proven credentials in the gaming world and easy availability even though such sensors are not as capable as their expensive industrial counterparts such as the Vicon™ or the XSens™ motion capture system. The deficiencies in terms of accuracy and precision were compensated to acceptable levels for the case studies chosen by leveraging prior knowledge of the tasks.

The framework was successfully implemented for digitising the task knowledge embedded within a simplified assembly task and for digitising operational data from a task environment. It was also successfully validated using 3 case studies with tasks ranging in complexity from simple to complex, involving different task environments, structures, workpieces, requirements, constraints and people. For all the studies, the framework was able to capture, extract, decode and reproduce all the important manufacturing knowledge constituents embedded within the tasks such as those that make up human action and reaction skills. This demonstrates the generality of the framework to accommodate the digitisation of most manual manufacturing tasks. The framework is also generic with respect to the tools used, which means that newer and better tools can be plugged into the framework without the need to change its basic design and structure.

This research contributes to knowledge in the five main areas, namely, (1) the theory of human-workpiece interactions to decipher human behaviour in manual manufacturing tasks, (2) a cohesive and holistic framework to digitise manual manufacturing task knowledge, especially tacit knowledge such as human action and reaction skills, (3) the use of low-cost gaming interface technology to capture human actions and the effect of those actions on workpieces during a manufacturing task, (4) a new way to use hidden Markov models as digital skill models to represent human ability to perform a complex task and (5) extraction and decoding of manufacturing knowledge constituents from digital skill models. The biggest contribution to research as a combination of all the above is the new ability to unearth and decode human skills that were always considered very difficult to explicitly document and transfer.

The significance of this research is its direct impact to enable faster and cost-effective skill transfer between people, enable detailed analysis of manual tasks on the shopfloor to assess task ergonomics in real-time, enable real-time collaboration between remote engineering teams and enable the intelligent automation of skill-intensive manual manufacturing tasks, all contributing towards enhancing the competitiveness of the manufacturing industry.

REFERENCES

Aachen (2012), “Guide to the Vicon Motion Capturing System”, available at: https://hci.rwth-aachen.de/guide_vicon_old (accessed 01Jan 2013).

Aggarwal, J. and Ryoo, M. S. (2011), "Human activity analysis: A review", ACM Computing Surveys (CSUR), vol. 43, no. 3, pp. 16.

Ali, H., Shafait, F., Giannakidou, E., Vakali, A., Figueroa, N., Varvadoukas, T. and Mavridis, N. (2013), "Contextual object category recognition for RGB-D scene labeling", Robotics and Autonomous Systems.

Andreopoulos, A. and Tsotsos, J. K. (2013), "50 Years of Object Recognition: Directions Forward", Computer Vision and Image Understanding.

Annett, J. (2003), "Hierarchical task analysis", Handbook of cognitive task design, pp. 17-35.

Asada, H. and Asari, Y. (1988), "The direct teaching of tool manipulation skills via the impedance identification of human motions", Proceedings of IEEE International Conference on Robotics and Automation, 1988, IEEE, pp. 1269.

Asif, U., Bennamoun, M. and Sohel, F. (2013), "Real-time pose estimation of rigid objects using RGB-D imagery", Proceedings of 8th IEEE Conference on Industrial Electronics and Applications (ICIEA), 2013, IEEE, pp. 1692.

Baum, L. E., Petrie, T., Soules, G. and Weiss, N. (1970), "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains", The annals of mathematical statistics, pp. 164-171.

BBC (2011), Microsoft Kinect 'fastest-selling device on record', available at: http://www.bbc.co.uk/news/business-12697975 (accessed 01 May 2015).

Bekey, G. A., Liu, H., Tomovic, R. and Karplus, W. J. (1993), "Knowledge-based control of grasping in robot hands using heuristics from human motor skills", IEEE Transactions on Robotics and Automation, vol. 9, no. 6, pp. 709-722.

Bengio, Y., De Mori, R., Flammia, G. and Kompe, R. (1992), "Global optimization of a neural network-hidden Markov model hybrid", IEEE Transactions on Neural Networks, vol. 3, no. 2, pp. 252-259.

Boodidhi, S. (2011), "Using smoothing techniques to improve the performance of Hidden Markov's Model".

Borenstein, G. (2012), "Making things see: 3D vision with kinect, processing, Arduino, and MakerBot." O'Reilly Media, Inc.

Buckingham, R. and Newell, G. (1996), "Automating the manufacture of composite broadgoods", Composites Part A: Applied Science and Manufacturing, vol. 27, no. 3, pp. 191-200.

Burns, R. (2000), "Introduction to Research Methods" (International edition), Sage Publications Limited, London.

Calinon, S. and Billard, A. (2005), "Recognition and reproduction of gestures using a probabilistic framework combining PCA, ICA and HMM", Proceedings of the 22nd international conference on Machine learning, ACM, pp. 105.

Calinon, S., Guenter, F. and Billard, A. (2007), "On learning, representing, and generalizing a task in a humanoid robot", IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, vol. 37, no. 2, pp. 286-298.

Cawley, S. L. and Pachter, L. (2003), "HMM sampling and applications to gene finding and alternative splicing", Bioinformatics (Oxford, England), vol. 19 Suppl 2, pp. ii36-41.

Chaaraoui, A. A., Padilla-López, J. R., Climent-Pérez, P. and Flórez-Revuelta, F. (2014), "Evolutionary joint selection to improve human action recognition with RGB-D devices", Expert Systems with Applications, vol. 41, no. 3, pp. 786-794.

Chen, C., Jafari, R. and Kehtarnavaz, N. (2015), "Improving Human Action Recognition Using Fusion of Depth Camera and Inertial Sensors", IEEE Transactions on Human-Machine Systems, vol. 45, no. 1, pp. 51-61.

Chen, H., Eakins, W., Wang, J., Zhang, G. and Fuhlbrigge, T. (2009), "Robotic wheel loading process in automotive manufacturing automation", § IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2009, IEEE, pp. 3814.

Chen, L., Wei, H. and Ferryman, J. (2013), "ReadingAct RGB-D action dataset and human action recognition from local features", Pattern Recognition Letters.

Chen, S. F. and Goodman, J. (1999), "An empirical study of smoothing techniques for language modeling", Computer Speech & Language, vol. 13, no. 4, pp. 359-393.

Cheng, G., Huang, Y., Wan, Y. and Buckles, B. P. (2015), "Exploring Temporal Structure of Trajectory Components for Action Recognition", International Journal of Intelligent Systems, vol. 30, no. 2, pp. 99-119.

Cho, C., Kang, S., Kim, M. and Song, J. (2005), "Macro-micro manipulation with visual tracking and its application to wheel assembly", International Journal of Control Automation and Systems, vol. 3, no. 3, pp. 461.

Coast, D. A., Stern, R. M., Cano, G. G. and Briller, S. A. (1990), "An approach to cardiac arrhythmia analysis using hidden Markov models", IEEE Transactions on Biomedical Engineering, vol. 37, no. 9, pp. 826-836.

Davenport, T. H. and Prusak, L. (1997), “Information ecology: Mastering the information and knowledge environment”, Oxford University Press.

De Tommaso, D., Calinon, S. and Caldwell, D. G. (2012), "A Tangible Interface for Transferring Skills", International Journal of Social Robotics, vol. 4, no. 4, pp. 397-408.

Dempster, A. P., Laird, N. M. and Rubin, D. B. (1977), "Maximum likelihood from incomplete data via the EM algorithm", Journal of the Royal Statistical Society, Series B (Methodological), pp. 1-38.

Dou, J. and Li, J. (2013), "Moving object detection based on improved VIBE and graph cut optimization", Optik-International Journal for Light and Electron Optics, vol. 124, no. 23, pp. 6081-6088.

Duan, F., Tan, J. T. C., Zhang, Y., Watanabe, K., Pongthanya, N., Sugi, M., Yokoi, H. and Arai, T. (2007), "Analyze assembly skills using a motion simulator", Proceedings of International Conference on Robotics and Biomimetics, (ROBIO), 2007, IEEE, pp. 1428.

Duan, F., Zhang, Y., Pongthanya, N., Watanabe, K., Yokoi, H. and Arai, T. (2008), "Analyzing human skill through control trajectories and motion capture data", Proceedings of IEEE International Conference on Automation Science and Engineering (CASE), 2008, IEEE, pp. 454.

Duncan, G. (2014), “Kinect 1 vs. Kinect 2; a quick side-by-side reference”, available at: http://channel9.msdn.com/coding4fun/kinect/Kinect-1-vs-Kinect-2-a-side-by-side-reference (accessed 25 May 2015).

Elkington, M., Bloom, D., Ward, C., Chatzimichali, A. and Potter, K. (2015), "Hand layup: Understanding the manual process", Advanced Manufacturing: Polymer & Composites Science, (Accepted but not published)

Engineering and Physical Sciences Research Council (EPSRC). (2014), “Autonomous Manufacturing Workshop Report - 4th September 2014”, available at: https://www.epsrc.ac.uk/newsevents/pubs/autonomous-manufacturing-workshop-report/ (accessed 17 Jul 2015).

Ezzedine, H. and Kolski, C. (2005), "Modelling of cognitive activity during normal and abnormal situations using Object Petri Nets, application to a supervision system", Cognition, Technology & Work, vol. 7, no. 3, pp. 167-181.

Faria, D. R., Martins, R., Lobo, J. and Dias, J. (2012), "Extracting data from human manipulation of objects towards improving autonomous robotic grasping", Robotics and Autonomous Systems, vol. 60, no. 3, pp. 396-410.

Ferguson, D. (2000), "Therbligs: the key to simplifying work", The Gilbreth.

Field, M., Pan, Z., Stirling, D. and Naghdy, F. (2011), "Human motion capture sensors and analysis in robotics", Industrial Robot: An International Journal, vol. 38, no. 2, pp. 163-171.

Funahashi, T., Fujiwara, T. and Koshimizu, H. (2011), "Development of visual inspection robot based on motion and mind behaviors of expert inspector", Proceedings of 4th International Conference on Human System Interactions (HSI), 2011, IEEE, pp. 169.

Georgilas, I. and Tourassis, V. (2008), "An intuitive framework to automate feature-rich manual tasks and manufacturing processes", Proceedings of IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), 2008, IEEE, pp. 1244.

Gibson, J. J., “The ecological approach to visual perception”, Lawrence Erlbaum Associates, 1986, New Jersey.

Grudic, G. Z. and Lawrence, P. D. (1996), "Human-to-robot skill transfer using the SPORE approximation", Proceedings of IEEE International Conference on Robotics and Automation, 1996, Vol. 4, IEEE, pp. 2962.

Guerra-Zubiaga, D. A. and Young, R. I. (2008), "Information and knowledge interrelationships within a manufacturing knowledge model", The International Journal of Advanced Manufacturing Technology, vol. 39, no. 1-2, pp. 182-198.

Gummesson, E. (1991), “Qualitative Methods in Management Research”, Sage Publications Limited, London.

Gunendran, A. G. and Young, R. (2010), "Methods for the capture of manufacture best practice in product lifecycle management", International Journal of Production Research, vol. 48, no. 20, pp. 5885-5904.

Hashimoto, H., Yoshida, I., Teramoto, Y., Tabata, H. and Han, C. (2011), "Extraction of tacit knowledge as expert engineer's skill based on mixed human sensing", Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication, 2011, IEEE, pp. 413.

Hassall, M. E. and Sanderson, P. M. (2012), "A formative approach to the strategies analysis phase of cognitive work analysis", Theoretical Issues in Ergonomics Science, ahead-of-print, pp. 1-47.

Hassan, M. R., Nath, B. and Kirley, M. (2007), "A fusion model of HMM, ANN and GA for stock market forecasting", Expert Systems with Applications, vol. 33, no. 1, pp. 171-180.

Hicks, B., Culley, S., Allen, R. and Mullineux, G. (2002), "A framework for the requirements of capturing, storing and reusing information and knowledge in engineering design", International Journal of Information Management, vol. 22, no. 4, pp. 263-280.

Howard, A. M., Park, H. W. and Kemp, C. C. (2008), "Extracting play primitives for a robot playmate by sequencing low-level motion behaviors", Proceedings of the 17th IEEE International Symposium on Robot and Human Interactive Communication, 2008, IEEE, pp. 360.

Hu, J., Zheng, W., Lai, J., Gong, S. and Xiang, T. (2015), "Exemplar-based recognition of human-object interactions", IEEE Transactions on Circuits and Systems for Video Technology. (Accepted but not published)

Huang, Z., Nagata, A., Kanai-Pak, M., Maeda, J., Kitajima, Y., Nakamura, M., Kyoko, A., Kuwahara, N., Ogata, T. and Jun, O. (2014), "Automatic Evaluation of Trainee Nurses' Patient Transfer Skills Using Multiple Kinect Sensors", IEICE Transactions on Information and Systems, vol. 97, no. 1, pp. 107-118.

Ikeuchi, K. and Suehiro, T. (1994), "Toward an assembly plan from observation. I. Task recognition with polyhedral objects", IEEE Transactions on Robotics and Automation, vol. 10, no. 3, pp. 368-385.

Inaba, T., Hayashizaki, M. and Matsuo, Y. (1999), "Design of a human-machine cooperation system to facilitate skilled work", 1999 IEEE International Conference on Systems, Man, and Cybernetics 'Human Communication and Cybernetics', Vol. 4, 12 October 1999 through 15 October 1999, Tokyo, Japan, IEEE, United States, pp. IV.

Jenkins, D. P., Stanton, N. A., Walker, G. H., Salmon, P. M. and Young, M. S. (2008), "Applying cognitive work analysis to the design of rapidly reconfigurable interfaces in complex networks", Theoretical Issues in Ergonomics Science, vol. 9, no. 4, pp. 273-295.

Jiang, Y. and Saxena, A. (2014), "Modeling high-dimensional humans for activity anticipation using gaussian process latent CRFs", Robotics: Science and Systems, RSS.

Kawashimo, T., Sato, N., Doyo, D., Anse, M. and Tabe, T. (2009), "A Skill Transfer Method for Manual Machine Tool Operation Utilizing Cutting Sound", in Human Interface and the Management of Information. Designing Information Environments, Springer, pp. 77-86.

Khoshelham, K. and Elberink, S. O. (2012), "Accuracy and resolution of kinect depth data for indoor mapping applications", Sensors, vol. 12, no. 2, pp. 1437-1454.

Kikuchi, T., Hamada, H., Nakai, A., Ohtani, A., Goto, A., Takai, Y., Endo, A., Narita, C., Koshino, T. and Fudauchi, A. (2013), "Relationships between Degree of Skill, Dimension stability and Mechanical Properties of Composites Structure in Hand Lay-Up Method", The 19th International Conference on Composite Materials, Montreal, Canada, pp. 8034.

Kikuchi, T., Tani, Y., Takai, Y., Goto, A. and Hamada, H. (2014), "Biomechanics Investigation of Skillful Technician in Spray-up Fabrication Method", in

Digital Human Modeling. Applications in Health, Safety, Ergonomics and Risk Management, Springer, pp. 24-34.

Kinect MSDN (2013), “Kinect for Windows Programming Guide”, available at https://msdn.microsoft.com/en-us/library/dn782037.aspx, (accessed 10 Dec 2014).

Kirwan, B. and Ainsworth, L. K. (1992), “A guide to task analysis: the task analysis working group”, CRC press.

Kjellström, H., Romero, J. and Kragić, D. (2011), "Visual object-action recognition: Inferring object affordances from human demonstration", Computer Vision and Image Understanding, vol. 115, no. 1, pp. 81-90.

Koo, S., Lee, D. and Kwon, D. "Incremental object learning and robust tracking of multiple objects from RGB-D point set data".

Kosuge, K., Fukuda, T. and Asada, H. (1991), "Acquisition of human skills for robotic systems", Proceedings of the IEEE International Symposium on Intelligent Control, 1991, IEEE, pp. 469.

Kruger, V., Herzog, D., Baby, S., Ude, A. and Kragic, D. (2010), "Learning actions from observations", Robotics & Automation Magazine, IEEE, vol. 17, no. 2, pp. 30-43.

Kuniyoshi, Y., Inaba, M. and Inoue, H. (1994), "Learning by watching: Extracting reusable task knowledge from visual observation of human performance", IEEE Transactions on Robotics and Automation, vol. 10, no. 6, pp. 799-822.

Lange, F., Werner, J., Scharrer, J. and Hirzinger, G. (2010), "Assembling wheels to continuously conveyed car bodies using a standard industrial robot", Proceedings of IEEE International Conference on Robotics and Automation (ICRA 2010), IEEE, pp. 3863.

Lengeling, T. (2014), “KinectPV2”, available at: http://codigogenerativo.com/ code/kinectpv2-k4w2-processing-library/ (accessed 10 Dec 2014).

Liu, H., Philipose, M., Pettersson, M., & Sun, M. T. (2014). “Recognizing object manipulation activities using depth and visual cues”, Journal of Visual Communication and Image Representation, Vol. 25 no. 4, pp. 719-726.

Liu, A., Nie, W., Su, Y., Ma, L., Hao, T. and Yang, Z. (2014), "Coupled hidden conditional random fields for RGB-D human action recognition", Signal Processing, vol. 112, pp. 74-82.

Makio, K., Tanaka, Y. and Uehara, K. (2007), "Discovery of skills from motion data", in New Frontiers in Artificial Intelligence, Springer, pp. 266-282.

Maimone, A. and Fuchs, H. (2012), "Reducing interference between multiple structured light depth sensors using motion", Virtual Reality Short Papers and Posters (VRW), 2012 IEEE, pp. 51.

Marfia, G., Roccetti, M., Marcomini, A., Bertuccioli, C. and Matteucci, G. (2012), "Reframing Haute Couture Handcraftship: How to Preserve Artisans' Abilities with Gesture Recognition", in Advances in Computer Entertainment, Springer, pp. 437-444.

Marian, R. M. (2003), "Optimisation of assembly sequences using genetic algorithms", Ph.D. Thesis. Adelaide, Australia: University of South Australia.

Matsuki, N. (2010), "Acquisition of skills on the shop-floor", Synthesiology English edition, vol. 3, no. 1, pp. 77-85.

Microsoft, Corp. (2014), “Kinect for Windows”, available at: http://www.microsoft.com/en-us/kinectforwindows/meetkinect/default.aspx (accessed 19 Feb 2015).

Montesano, L., Lopes, M., Bernardino, A. and Santos-Victor, J. (2008), "Learning Object Affordances: From Sensory--Motor Coordination to Imitation", IEEE Transactions on Robotics, vol. 24, no. 1, pp. 15-26.

Nechyba, M. C. and Xu, Y. (1995), "Human skill transfer: neural networks as learners and teachers", Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems - Human Robot Interaction and Cooperative Robots, 1995, Vol. 3, IEEE, pp. 314.

Nefian, A. V. and Hayes, M. H. (2000), "Maximum likelihood training of the embedded HMM for face detection and recognition", Image Processing, 2000. Proceedings. 2000 International Conference on, Vol. 1, IEEE, pp. 33.

Nickols, F. (2000), "The knowledge in knowledge management", The Knowledge Management Yearbook, 2000–2001.

Nof, S. Y., Wilhelm, W. and Warnecke, H. (1997), “Industrial Assembly”, First edition, Chapman and Hall, London.

Nonaka, I. (1991), "The knowledge-creating company", Harvard business review, vol. 69, no. 6, pp. 96-104.

Nonaka, I. and Takeuchi, H. (1995), The knowledge-creating company: How Japanese companies create the dynamics of innovation, Oxford university press.

Ntouskos, V., Papadakis, P. and Pirri, F. (2012), "A Comprehensive Analysis of Human Motion Capture Data for Action Recognition.", VISAPP (1), pp. 647.

Okuda, T. (2007), "A new transforming method from tacit knowledge to explicit knowledge by using humanoid robot for verbalization", 2007 IEEE International Professional Communication Conference, IPCC, 1 to 3 October 2007, Seattle, WA.

Oxford Economics (2012), Global Talent 2021: “How the new geography of talent will transform human resource strategies”, Oxford.

Padrón, M., Irizarry, María de los A, Resto, P. and Mejía, H. P. (2009), "A methodology for cost-oriented assembly line balancing problems", Journal of Manufacturing Technology Management, vol. 20, no. 8, pp. 1147-1165.

Pan C (2005), “Integrating CAD files and automatic assembly sequence planning”, PhD thesis, Iowa State University, ProQuest.

Pena, M., Lopez, J. and Osorio, R. (2007), "Robot Vision Methodology for Assembly Manufacturing Tasks", Electronics, Robotics and Automotive Mechanics Conference, 2007. CERMA 2007, IEEE, pp. 289.

Polyani, M. (1966), "The tacit dimension".

Rabiner, L. (1989), "A tutorial on hidden Markov models and selected applications in speech recognition", Proceedings of the IEEE, vol. 77, no. 2, pp. 257-286.

Rabiner, L. and Juang, B. (1986), "An introduction to hidden Markov models", ASSP Magazine, IEEE, vol. 3, no. 1, pp. 4-16.

Rahman, S. A., Song, I., Leung, M., Lee, I. and Lee, K. (2014), "Fast action recognition using negative space features", Expert Systems with Applications, vol. 41, no. 2, pp. 574-587.

Rasmussen, J. (1980), "The human as a system component", Human interaction with computers, pp. 67-96.

Rasmussen, J. (1981), "Models of mental strategies in process plant diagnosis", in Human detection and diagnosis of system failures, Springer, pp. 241-258.

Rasmussen, J. (1983), "Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models", IEEE Transactions on Systems, Man and Cybernetics, no. 3, pp. 257-266.

Ren, S., & Sun, Y. (2013), “Human-object-object-interaction affordance”, IEEE Workshop on Robot Vision (WORV 2013), IEEE, pp. 1-6.

Robson, C. (2002), “Real World Research”, Blackwell, Oxford.

Roth, E. (2009), "Understanding cognitive strategies for shared situation awareness across a distributed system: An example of strategies analysis", Applications of Cognitive Work Analysis. Boca Raton, FL: CRC Press, Taylor & Francis Group, pp. 129-147.

Sakaida, Y., Chugo, D., Yamamoto, H. and Asama, H. (2008), "The analysis of excavator operation by skillful operator-extraction of common skills", SICE Annual Conference, 2008, IEEE, pp. 538.

Salmon, P., Jenkins, D., Stanton, N. and Walker, G. (2010), "Hierarchical task analysis vs. cognitive work analysis: comparison of theory, methodology and contribution to system design", Theoretical Issues in Ergonomics Science, vol. 11, no. 6, pp. 504-531.

Sattar, A., Bakhsh, Q., & Sharif, M. (2014), “Industrial Automation and Manufacturing Systems: Concepts and Applications”, Advanced Materials Research, vol. 903, pp. 291-296.

Shao, L., Ji, L., Liu, Y. and Zhang, J. (2012), "Human action segmentation and recognition via motion and shape analysis", Pattern Recognition Letters, vol. 33, no. 4, pp. 438-445.

Shi, J. (2008), "Preliminary analysis of conveyor dynamic motion for automation applications", Proceedings of the 8th Workshop on Performance Metrics for Intelligent Systems, ACM, pp. 156.

Shi, J. and Menassa, R. (2010), "Flexible robotic assembly in dynamic environments", Proceedings of the 10th Performance Metrics for Intelligent Systems Workshop, ACM, pp. 271.

Shon, A. P., Grochow, K. and Rao, R. P. (2005), "Robotic imitation from human motion capture using gaussian processes", Proceedings of 5th IEEE-RAS International Conference on Humanoid Robots, IEEE, pp. 129.

Shotton, J., Sharp, T., Kipman, A., Fitzgibbon, A., Finocchio, M., Blake, A., Cook, M. and Moore, R. (2013), "Real-time human pose recognition in parts from single depth images", Communications of the ACM, vol. 56, no. 1, pp. 116-124.

Skordos, A., Monroy A C. and Sutcliffe, M. (2005), "Drape optimisation in woven composite manufacturing", Proceedings of the 5th International Conference on Inverse Problems in Engineering, Vol. 3, pp. S09.

Skubic, M. and Volz, R. A. (2000), "Acquiring robust, force-based assembly skills from human demonstration", IEEE Transactions on Robotics and Automation, vol. 16, no. 6, pp. 772-781.

Smeenk, R. (2015), “Real life Portal; a holographic window using Kinect”, available at: http://smeenk.com/category/kinect/ (accessed 10 May 2015).

Smith, E. A. (2001), "The role of tacit and explicit knowledge in the workplace", Journal of knowledge Management, vol. 5, no. 4, pp. 311-321.

Stanton, N. A. (2006), "Hierarchical task analysis: Developments, applications, and extensions", Applied Ergonomics, vol. 37, no. 1, pp. 55-79.

Such, M., Ward, C., Hutabarat, W. and Tiwari, A. (2014), "Intelligent composite layup by the application of low cost tracking and projection technologies", Procedia CIRP, vol. 25, pp. 122-131.

Sung, J., Ponce, C., Selman, B. and Saxena, A. (2012), "Unstructured human activity detection from rgbd images", Proceedings of IEEE International Conference on Robotics and Automation (ICRA 2012), IEEE, pp. 842.

Sutton, C. and McCallum, A. (2011), "An introduction to conditional random fields", Machine Learning, vol. 4, no. 4, pp. 267-373

Koppula, H. S., Gupta, R., & Saxena, A. (2013). “Learning human activities and object affordances from rgb-d videos”. The International Journal of Robotics Research, vol. 32 no. 8, pp. 951-970.

Takamatsu, J., Tominaga, H., Ogawara, K., Kimura, H. and Ikeuchi, K. (2000), "Extracting manipulation skills from observation”, Proceedings of

IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, 2000, Vol. 1, IEEE, pp. 584.

Tsai, Y., Li, J., Lee, J. and Jywe, W. (2012), "Development and Implementation of Automatic Scraping Mechanism", Advanced Science Letters, vol. 8, no. 1, pp. 211-215.

Tuomi, I. (1999), "Data is more than knowledge: Implications of the reversed knowledge hierarchy for knowledge management and organizational memory", Proceedings of the 32nd Annual Hawaii International Conference on Systems Sciences HICSS-32, 1999, IEEE, pp. 12.

Uddin, M. Z., Lee, J. and Kim, T. (2010), "Independent shape component-based human activity recognition via Hidden Markov Model", Applied Intelligence, vol. 33, no. 2, pp. 193-206.

UK Commission for Employment and Skills' (UKCES) (2013), "Employer Skills Survey 2013", available at: https://www.gov.uk/government/publications/ukces-employer-skills-survey-2013-supplementary-documents (accessed Jul/17).

Vicente, K. J. (1999), "Cognitive work analysis: Toward safe, productive, and healthy computer-based work", CRC Press.

Watanuki, K. (2008), "VR Mediated Skill Transfer", ASME.

Weinland, D., Ronfard, R. and Boyer, E. (2011), "A survey of vision-based methods for action representation, segmentation and recognition", Computer Vision and Image Understanding, vol. 115, no. 2, pp. 224-241.

Wiig, K. M. (1994), "Knowledge Management Foundations: Thinking about Thinking-how People and Organizations Represent, Create, and Use Knowledge", Schema Press Limited.

Winterbotham M., Vivian D., Shury J., Davies B. and Genna K. (2014), "UK Commission's Employer Skills Survey 2013: UK Results, Evidence Report 81", UK Commission for Employment and Skills, London.

Wohlhart, P. and Lepetit, V. (2015), "Learning Descriptors for Object Recognition and 3D Pose Estimation", arXiv preprint arXiv:1502.05908.

Xsens (2015), “Xsens MVN full-body, wearable motion capture solutions”, available at: https://www.xsens.com/products/xsens-mvn/ (accessed 19 Feb 2015).

Xu, Y. and Yang, J. (1995), "Towards human-robot coordination: skill modeling and transferring via hidden Markov model", Proceedings of IEEE International Conference on Robotics and Automation, 1995, Vol. 2, IEEE, pp. 1906.

Yamamoto, Y., Hashimoto, T., Okubo, T. and Itoh, T. (2001), "Task analysis of ultra-precision assembly processes for automation of human skills", , 2001. Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, 2001, Vol. 4, IEEE, pp. 2093.

Yang, J, Xu, Y. and Chen, C. S. (1994), “Hidden markov model approach to skill learning and its application to telerobotics”, IEEE Transactions in Robotics and Automation, vol. 10 (5), pp:421-631.

Yang, X., Zhang, C. and Tian, Y. (2012), "Recognizing actions using depth motion maps-based histograms of oriented gradients", Proceedings of the 20th ACM international conference on Multimedia, ACM, pp. 1057.

Yeasin, M. and Chaudhuri, S. (2000), "Toward automatic robot programming: learning human skill from visual data", Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, vol. 30, no. 1, pp. 180-185.

Yin, R. K. (2014), “Case study research: Design and methods”, Sage publications.

Yoon, C., Cheon, M. and Park, M. (2013), "Object tracking from image sequences using adaptive models in fuzzy particle filter", Information Sciences, vol. 253, pp. 74-99.

Yoon, S. M. and Kuijper, A. (2013), "Human action recognition based on skeleton splitting", Expert Systems with Applications, vol. 40, no. 17, pp. 6848-6855.

Yoshida, I., Teramoto, Y., Tabata, H., Han, C. and Hashimoto, H. (2011), "Tacit knowledge extraction of skillful operation from expert engineers", IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), 2011, IEEE, pp. 554.

Young, R. (2003), "Informing decision-makers in product design and manufacture", International Journal of Computer Integrated Manufacturing, vol. 16, no. 6, pp. 428-438.

Zennaro, S. (2014), "Evaluation of Microsoft Kinect 360 and Microsoft Kinect One for robotics and computer vision applications".

Zhang, R., Wei, F. and Li, B. (2014), "E2LSH based multiple kernel approach for object detection", Neurocomputing, vol. 124, pp. 105-110.

Ziaeefard, M., & Bergevin, R. (2015), “Semantic human activity recognition: a literature review” Pattern Recognition, vol. 8, no. 48, pp. 2329-2345.

Appendix A: Literature review for the wheel loading case study

The difficulty of using a rigidly programmed industrial robot to load wheels on a moving vehicle body has been recognised in literature. Since industrial robots must be pre-programmed with little flexibility in their task execution, it is difficult for them to cater to complex requirements, as in wheel loading (Chen et al., 2009). A few articles have reported attempts to automate wheel loading by proposing industrial sensor based methods to replace the human skills of simultaneously tracking the moving vehicle body to anticipate the precise aligning and loading moves for successful assembly.

Cho et al. have reported the use of a visual tracking manipulator using a camera on the wheel gripper mounted on an industrial robot that loads the wheel to track the centre of the wheel hub on the moving vehicle body (Cho et al., 2005). The visual tracking method is divided into macro tracking that monitors the velocity of the moving vehicle body and micro tracking that monitors the fine positional errors to assist in precision wheel loading. However, there is no mention of misalignments being identified. Chen et al. have reported a method of visual servoing to track the motion of the vehicle body in two axes to determine the wheel-loading instance and position (Chen et al., 2009). Force sensors that measure the loading force along all 3 axes are used for precise control of the final robot movement towards the wheel hub to perform loading according to set values of compliant contact forces between the robot tool and the wheel hub. Misalignment between the wheel and the wheel hub is also checked by the visual servoing system and transformation is applied to correct it. Shi (2008) has reported a preliminary analysis of dynamic conveyor motion and presented the typical motion characteristics of industrial conveyors such as speed, acceleration and multi-axis deviations in motion. Based on that study, Shi and Menassa (2010) have proposed a method in which a coarse vision camera tracks the general motion characteristics of the moving vehicle body with lower accuracy and a fine vision camera to track the deviations in vehicle body motion just before loading is performed. A vision camera placed at the end of the industrial robot arm that loads the wheel is used to locate the wheel hub studs for alignment. Lange et al. (2010) have also proposed a coarse and fine sensing system and a compliant force-torque sensor in the robot end-effector to control the loading step to compensate for final temporal or spatial offsets. Predictive modelling of robot motion trajectory in addition to the computed trajectories based on vision inputs is used to enhance loading precision. The camera on the robot end-effector identifies the positions of the wheel hub studs with respect to the positions of the wheel bores to determine misalignment.

In all of the above articles, industrial vision systems such as costly stereo-vision cameras are used for object tracking and feature recognition. These systems require computationally expensive image processing and pattern matching algorithms. Also because of their use of colour values of pixels for isolating target objects from the background, ambient light might affect image-processing accuracy and therefore active computational intervention is required to compensate for changes in lighting conditions. Thirdly, vision systems can only provide effective object tracking along two axes and additional force sensors are required to provide the same along the third axis.

In this study, inexpensive Kinect sensors and depth-based object recognition methods in the framework are used to track and obtain motion characteristics of the moving wheel hub along all 3 axes simultaneously. Depth data is also used to recognise alignment features on the stationary wheel and the moving wheel hub. Since depth data is provided by Infra-Red (IR) and not visible light (RGB) imaging, the proposed technique does not depend on ambient light conditions. The motion data obtained from the Kinect sensors for the moving wheel hub is also compared to that obtained from a highly accurate laser motion tracker. The main objective of this comparison is to gauge the accuracy and precision of the consumer-grade Kinect sensor vis-à-vis its expensive industrial counterpart.

Appendix B: Detailed results of the wheel loading case study

The results of the task environment digitisation are presented in this section in the following order:

1. Identification of wheel features and measurement of the angular positions of the wheel bores.
2. Motion tracking of the moving wheel hub and identification of the angular positions of the wheel studs for the following programmed motion patterns:
    a. Linear motion along x-axis with no deviations in y and z axis.
    b. Jerky motion along x-axis with no deviations in y and z axis.
    c. Linear motion along x-axis with sinusoidal deviation in y-axis.
    d. Linear motion along x-axis with sinusoidal deviation in z-axis.
    e. Linear motion along x-axis with sinusoidal deviation in y and z-axis.

Identification of wheel features and measurement of the angular positions of the wheel bores

The Kinect sensor captures depth images of the stationary wheel at the rate of up to 30 frames per second. From within each depth image, the 4 bores of the wheel are recognised and their angular positions, represented by the angle of the bore located within the 90^o to 180^o quadrant (the 'first bore'), are measured. To improve the accuracy of this method, the angle obtained is cumulatively averaged over 45 depth frames before it is recorded. 10 iterations of the experiment are conducted and the results are tabulated in Table 31.

Table 31: First wheel bore angle and its standard deviation (10 iterations)

Iteration	1	2	3	4	5	6	7	8	9	10	Mean	Standard Deviation
Angle	102.95	102.98	102.99	103.01	103.03	103.08	103.25	103.42	103.57	103.73	**103.20**	**0.28**

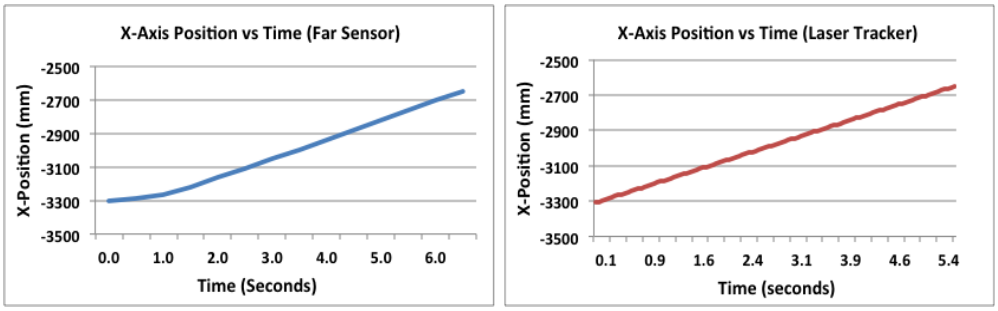
Motion tracking of the moving wheel hub and identification of the angular positions of the wheel studs

The far and near sensors track the motion of the wheel hub by continuously detecting the centre point of the hub and recording its x, y and z coordinates along with its speed in the direction of motion (x-axis). In the far sensing zone, the far sensor tracks the position and speed of the wheel hub whereas in the near sensing zone, the near sensor tracks its motion and identifies the angular positions of the studs of the moving wheel hub.

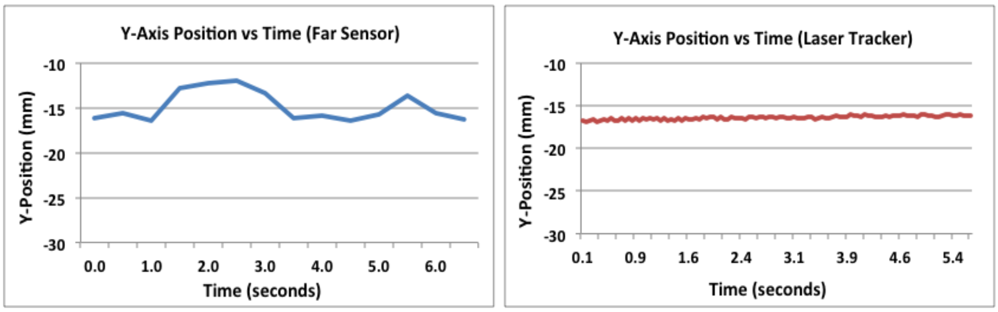
The motion tracking data obtained from the far and near sensors is compared to that obtained from the laser tracker that tracks the same motion. Since the laser tracker and the depth sensor are not synchronised during motion tracking, the two sets of data cannot be plotted and visualised on the same chart. The motion tracking results for the five simulated motion patterns are presented below. Since each motion pattern is run for 10 iterations, the wheel hub position and speed values are averaged over the 10 iterations.

1. Linear motion along x-axis at 67mm/s with no deviations in y and z axis

<u>In the far sensing zone:</u> Wheel hub motion tracked along all three axes by the far sensor and the laser tracker is presented in charts shown in Figure 122. The corresponding speed computed from the far sensor and laser tracker motion data is plotted in the charts shown in Figure 123.



(a)



(b)

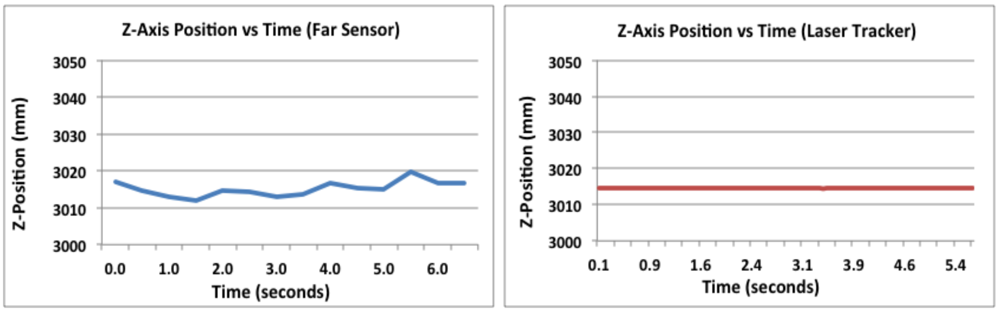
(c)


Figure 122: Wheel hub positions - far sensor and the laser tracker along (a) x-axis, (b) y-axis, and (c) z-axis

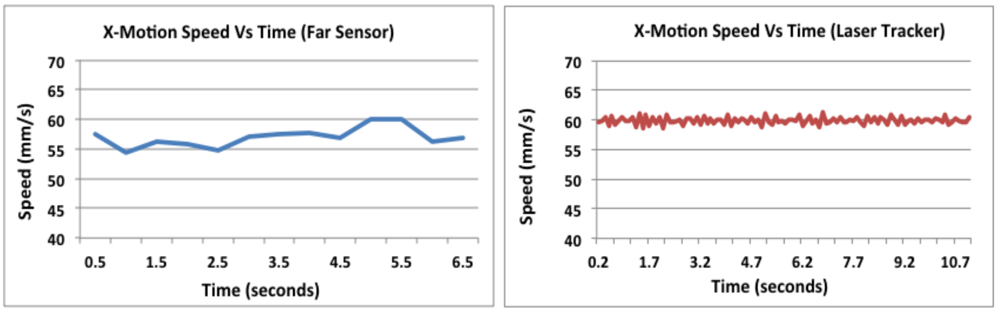
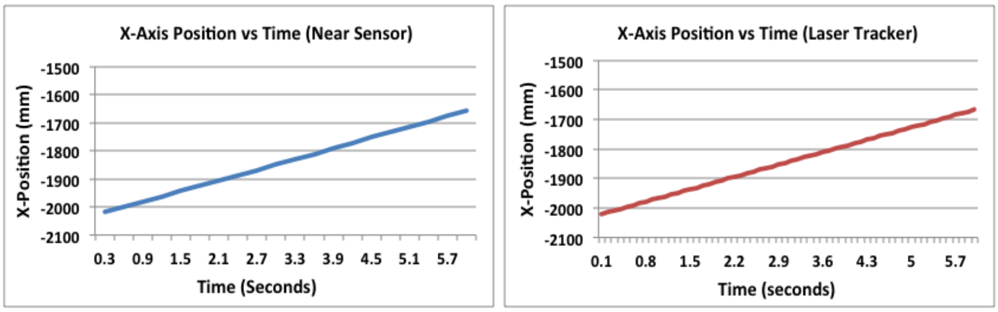


Figure 123: Wheel hub motion speed (x-axis) - far sensor and the laser tracker

<u>In the near sensing zone</u>: Wheel hub motion tracked along all three axes by the near sensor and the laser tracker is presented in charts shown in Figure 124. The corresponding speed computed from the near sensor and laser tracker motion data is plotted in the charts shown in Figure 125. In this zone, the angular position of the wheel hub stud located in the 90^o to 180^o quadrant (the 'first stud') is also measured for 10 iterations as shown in Table 32.

(a)


(b)

(c)

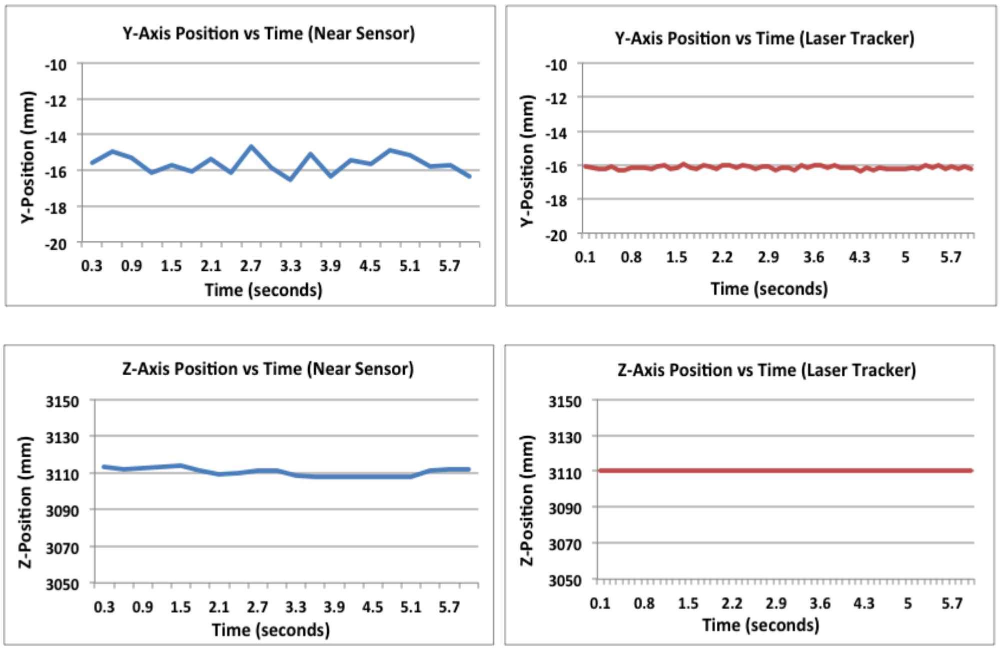


Figure 124: Wheel hub positions - near sensor and the laser tracker along (a) x-axis, (b) y-axis, and (c) z-axis

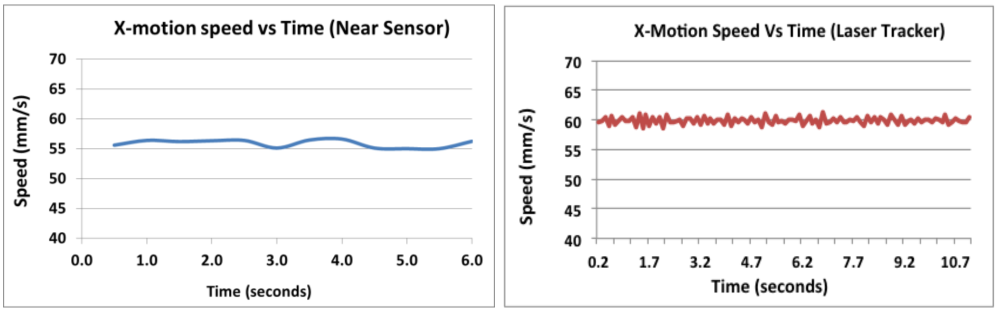


Figure 125: Wheel hub motion speed (x-axis) - near sensor and the laser tracker

Table 32: First wheel hub stud angle and its standard deviation (10 iterations)

Motion Pattern	Angular Position of First Wheel Hub Stud (degrees)											
	1	2	3	4	5	6	7	8	9	10	Mean	Std. Deviation
Linear motion in x-axis with no deviations in y and z-axis	335.31	334.94	335.72	335.56	333.67	334.57	334.90	335.30	335.40	335.78	335.12	0.63

2. Jerky motion along x-axis with no deviations in y and z axis

Since jerky motion is along the x-axis only, the y-axis and z-axis motion tracking charts are not shown.

<u>In the far sensing zone:</u> Figure 126 and Figure 127 show the motion and speed charts produced by the far sensor and the laser tracker respectively.

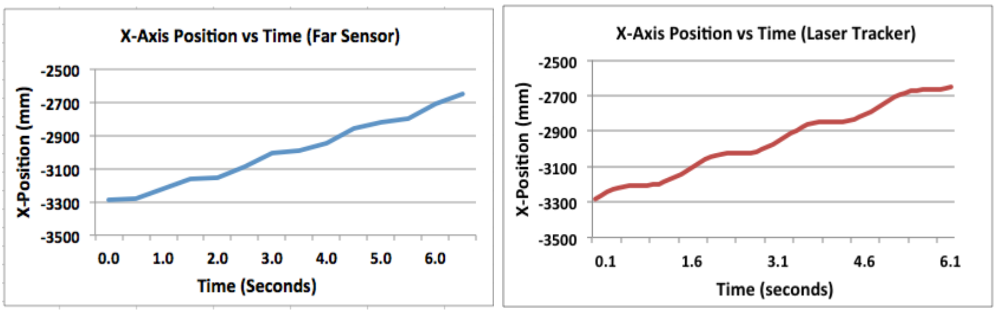


Figure 126: Wheel hub positions - far sensor and the laser tracker (x-axis)

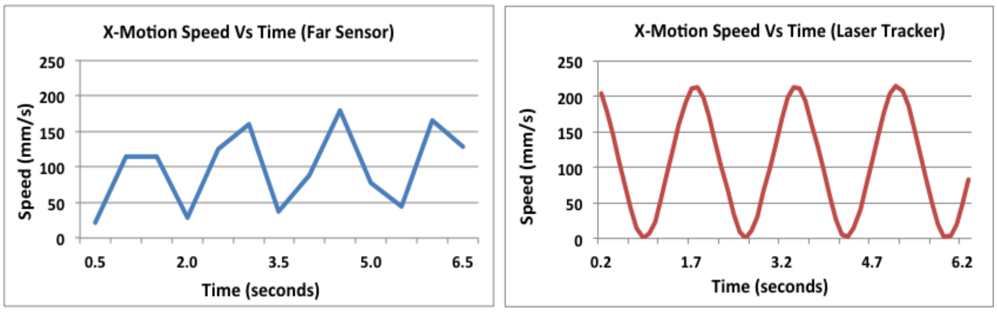


Figure 127: Wheel hub speed - far sensor and the laser tracker (x-axis)

<u>In the near sensing zone:</u> Figure 128 and Figure 129 show the motion and speed charts produced by the near sensor and the laser tracker respectively. Table 33 shows the angular positions of the wheel hub measured over 10 iterations for this motion pattern.

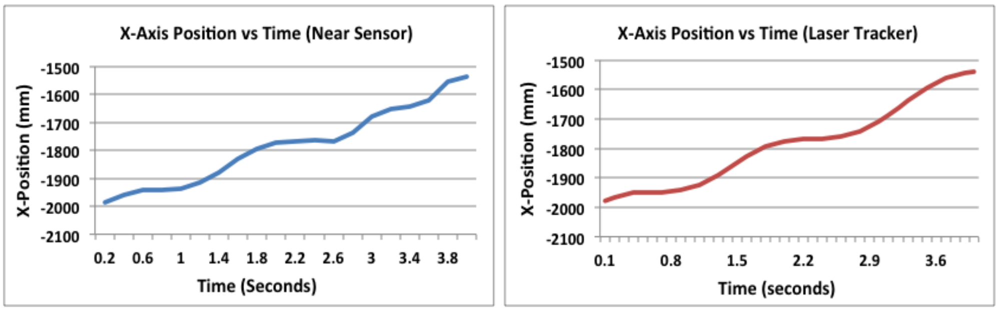


Figure 128: Wheel hub positions - near sensor and the laser tracker (x-axis)

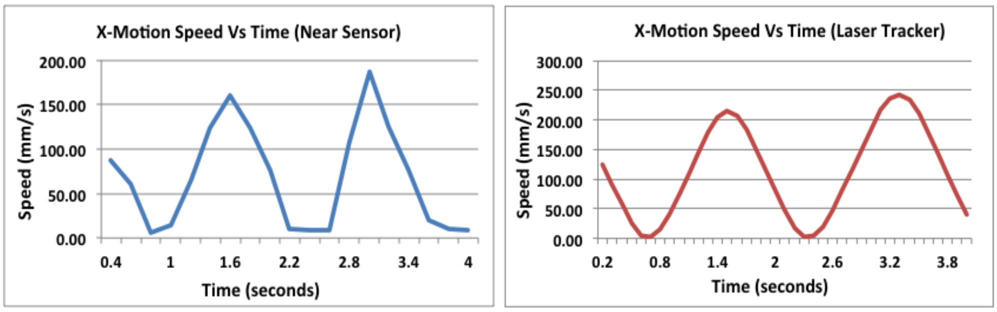


Figure 129: Wheel hub speed - near sensor and the laser tracker (x-axis)

Table 33: First wheel hub stud angle and its standard deviation (10 iterations)

Motion Pattern	Angular Position of First Wheel Hub Stud (degrees)											
	1	2	3	4	5	6	7	8	9	10	Mean	Std. Deviation
Jerky motion in x-axis with no deviations in y and z-axis	336.70	335.16	334.07	336.26	335.67	333.18	338.56	334.36	336.08	335.79	335.58	1.51

3. Linear motion at 67mm/s along x-axis with deviations in y-axis

Since the oscillations are along the y-axis only, x-axis and z-axis motion tracking charts are not shown.

<u>In the far sensing zone:</u> Figure 130 shows the motion charts produced by the far sensor and the laser tracker.

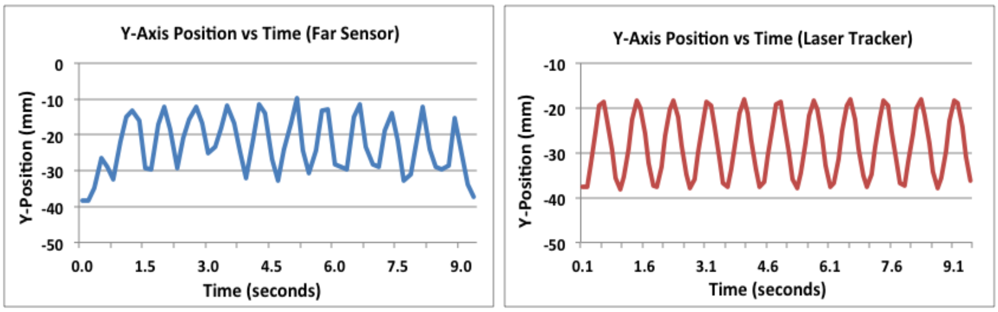


Figure 130: Wheel hub positions - far sensor and the laser tracker (y-axis)

<u>In the near sensing zone:</u> Figure 131 shows the motion charts produced by the near sensor and the laser tracker. Table 34 shows the angular positions of the wheel hub measured over 10 iterations for this motion pattern.

Table 34: First wheel hub stud angle and its standard deviation (10 iterations)

Motion Pattern	Angular Position of First Wheel Hub Stud (degrees)											
	1	2	3	4	5	6	7	8	9	10	Mean	Std. Deviation
Linear motion in x-axis with deviations in y-axis	337.70	336.16	334.07	336.26	335.67	333.18	335.56	337.36	336.08	334.79	335.68	1.39

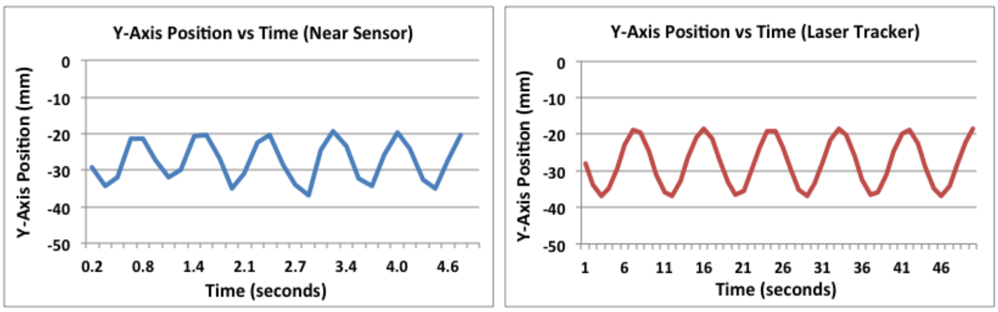


Figure 131: Wheel hub positions - near sensor and the laser tracker (y-axis)

4. Linear motion at 67mm/s along x-axis with deviations in z-axis

Since the oscillations are along the z-axis only, x-axis and y-axis motion tracking charts are not shown.

In the far sensing zone: Figure 132 shows the motion charts produced by the far sensor and the laser tracker.

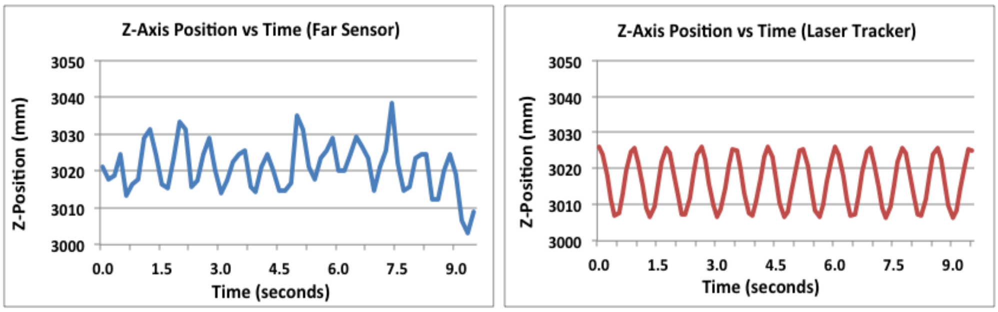


Figure 132: Wheel hub positions - far sensor and the laser tracker (z-axis)

In the near sensing zone: Figure 133 shows the motion charts produced by the near sensor and the laser tracker. Table 35 shows the angular positions of the wheel hub measured over 10 iterations for this motion pattern.

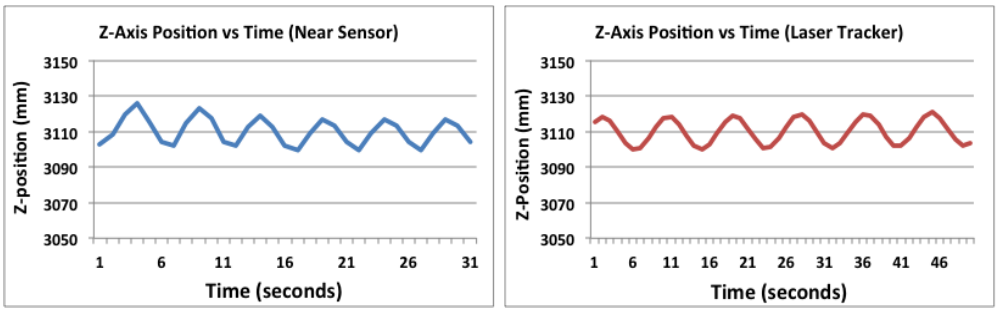


Figure 133: Wheel hub positions - near sensor and the laser tracker (z-axis)

Table 35: First wheel hub stud angle and its standard deviation (10 iterations)

Motion Pattern	Angular Position of First Wheel Hub Stud (degrees)											
	1	2	3	4	5	6	7	8	9	10	Mean	Std. Deviation
Linear motion in x-axis with deviations in z-axis	335.68	332.52	336.59	336.76	334.60	337.22	337.66	335.45	335.94	337.02	335.94	1.52

Appendix C: Sensor setup for wheel loading case study

The use of Kinect sensors in a real manufacturing environment to obtain live moving assembly data is relatively new in literature. Little is known on the optimum sensor positioning parameters and their influence on data capture precision and accuracy, such as the perpendicular distance of the sensor from the objects to be tracked and the sensor face plane angle with respect to the assembly line plane. Therefore, experiments are conducted to determine the effects of variations in sensor positions on the measured data and to obtain the optimum setup for data capture (Figure 134).

Figure 134: The experiment setup for optimising sensor positioning parameters

<u>The impact of distance of the sensor from the observed object</u>

It was observed that at a distance of 950mm and above, the features of the wheel were too small to be rendered in the depth image whereas the minimum distance below which the feature recognition algorithm does not work is 700mm. Therefore, the distance between the sensor and the wheel was varied from 700mm to 950mm and the optimum distance of 850mm was obtained with the least standard deviation of 0.28^{o} (Figure 135).

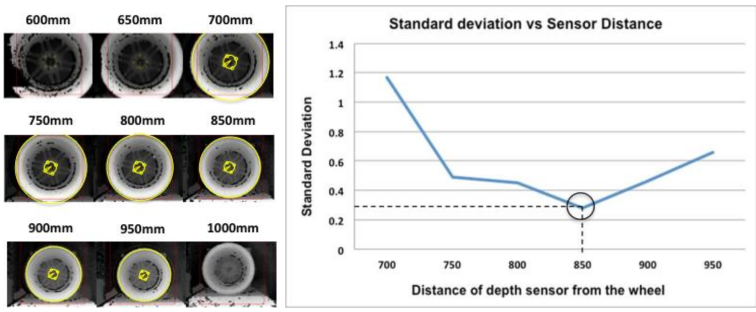


Figure 135: Impact of sensor distance on feature recognition precision

<u>The impact of sensor face angle with respect to the object plane</u>

The feature recognition algorithm uses depth values of the pixels corresponding to the object being tracked to recognise features and measure its angular positions. Therefore, it is expected that the sensor face (Figure 136) is perfectly parallel (relative angle of zero) to the object face plane at all times.

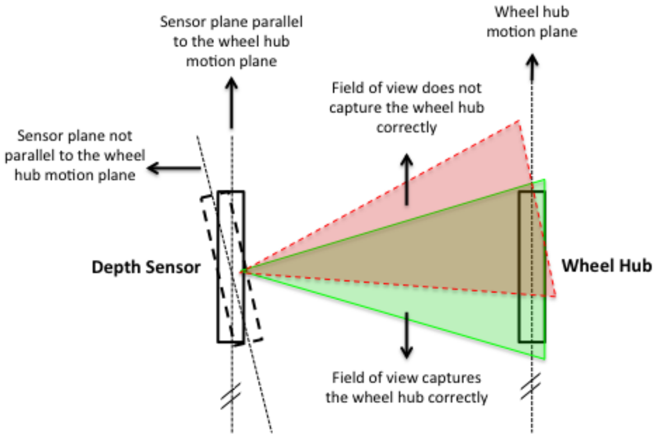


Figure 136: Impact of sensor face plane angle on the depth image capture

However, this is difficult to achieve in a real manufacturing scenario and therefore, it is necessary to find the angle range within which the proposed method can work. The effect of sensor face angle with respect to the wheel face plane on feature recognition effectiveness was investigated by varying the sensor face angle from -20^{o} to 20^{o}. The results below show that the feature recognition works reliably only within the -10^{o} to 10^{o} range (Figure 137).

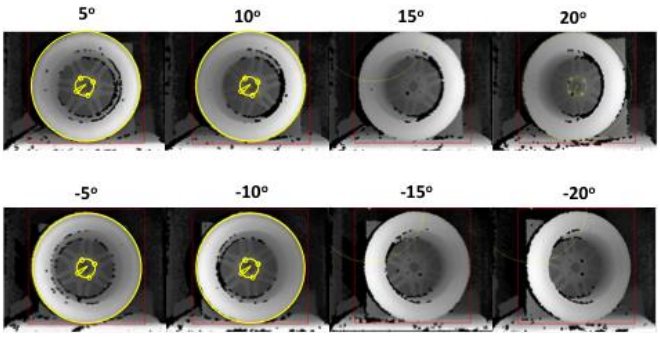


Figure 137: Alignment feature recognition at different sensor face plane angles

<u>Impact of number of frames used for averaging</u>

In this study, the cumulative averaging technique is used to reduce the measurement errors while obtaining the angular positions of the alignment features on the wheel and the moving wheel hub. The sensor produces 30 depth image frames per second and the algorithm processes each image to recognise the features and measure their angular positions. Because of the noise present in the sensor depth data, measurement obtained from only one frame does not suffice. Therefore, angular positions measured from multiple frames are averaged to determine the final angular positions. The number of frames averaged was varied from 1 frame (no averaging) to 120 frames and the optimum number of frames was found to be 45 with the least standard deviation of 0.17^{o} (Figure 138). Averaging over 45 frames results in a delay of 1.5 seconds to obtain the angular position result, which is satisfactory.

Angle of the First Wheel Bore									
Iteration	Number of Frames Averaged								
	1	15	30	45	60	75	90	105	120
1	100.77	100.75	101.14	101.25	101.31	101.28	101.38	101.23	101.17
2	101.23	100.76	101.36	101.63	101.75	101.40	101.74	101.36	101.75
3	102.60	100.85	101.48	101.70	101.75	101.43	101.75	101.36	101.75
4	100.54	100.89	101.71	101.75	101.75	101.75	101.75	101.39	101.75
5	101.78	101.32	101.75	101.75	101.75	101.76	101.75	101.57	101.75
6	101.51	101.51	101.76	101.75	101.77	101.76	101.76	101.60	101.75
7	102.00	101.97	101.76	101.81	101.79	101.77	101.87	101.75	101.76
8	102.95	102.06	101.84	101.75	101.79	101.78	101.88	101.76	101.79
9	101.59	102.23	102.14	101.81	101.80	101.82	101.88	101.80	101.82
10	101.71	102.25	102.24	101.84	102.00	101.82	102.04	101.83	101.90
Mean	**101.67**	**101.46**	**101.72**	**101.70**	**101.75**	**101.66**	**101.78**	**101.57**	**101.72**
Std. Dev.	**0.74**	**0.63**	**0.33**	**0.17**	**0.17**	**0.20**	**0.17**	**0.22**	**0.20**

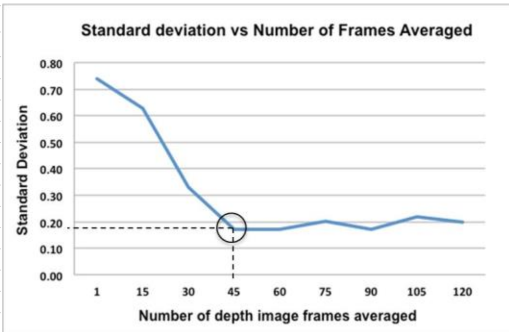
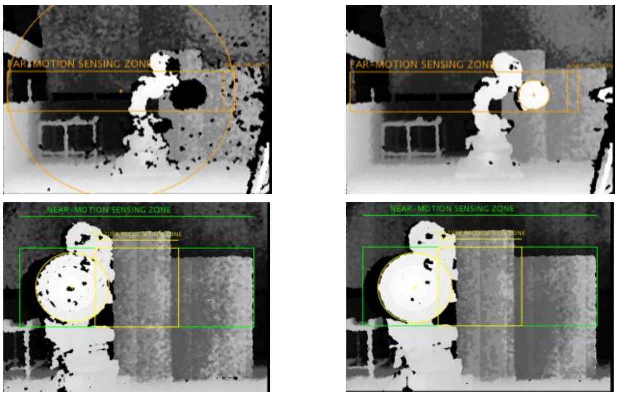


Figure 138: Number of frames averaged impact the data accuracy

Impact of IR interference between the two Kinect sensors

A depth sensor is an Infra-Red (IR) light emitting device, which measures depth by processing the IR waves that are reflected back to it from the surfaces in its view. Therefore, when two or more sensors are used to observe the same scene, the IR waves emitted by the sensors interfere with each other causing significant noise in depth data obtained from both sensors (Maimone and Fuchs, 2012). Due to this constraint, the far and near sensors are time-multiplexed to avoid their simultaneous operation (Figure 139).



(a) (b)

Figure 139: Depth images (a) with IR interference and (b) without IR interference

Accuracy of depth sensor motion tracking

In this work, motion data captured by the depth sensors is compared with that obtained from the industry standard laser tracker. Hence, the error in depth sensor motion data is computed relative to the motion data produced by the laser tracker. The motion parameters used for the relative error computation are values of motion speed, deviation amplitude and deviation frequency for all 5 motion patterns averaged over 10 iterations each (Table 36).

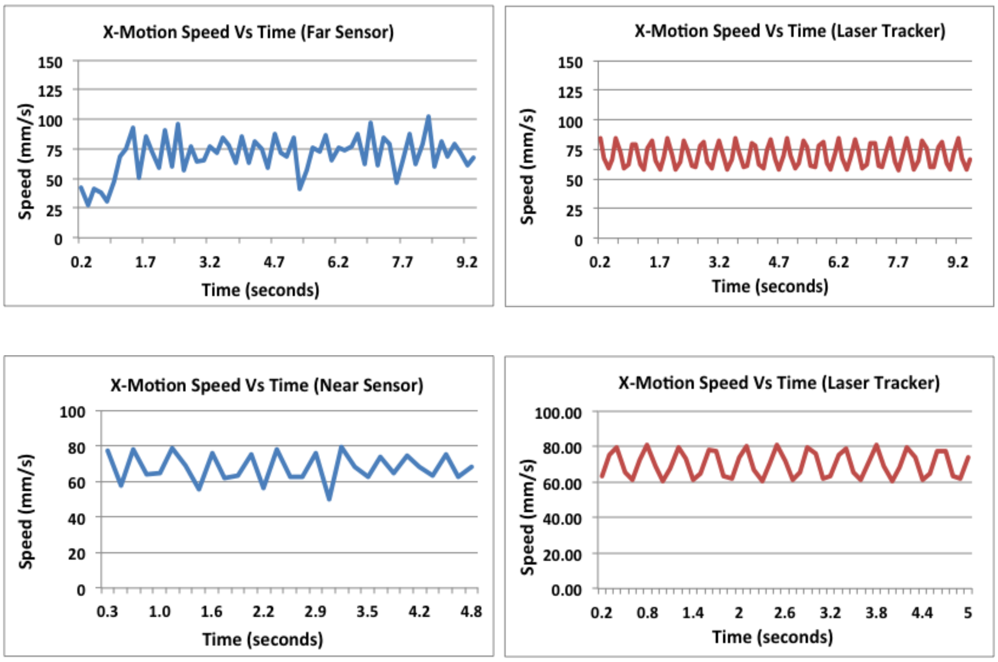
Table 36: Relative errors in depth sensor motion tracking data

Sr. No.	Motion Scenario	Average Speed Error (mm/s)		Average Deviation Amplitude Error (mm)		Average Deviation Frequency Error (Hz)	
		Far Sensor	Near Sensor	Far Sensor	Near Sensor	Far Sensor	Near Sensor
1	Linear motion along x-axis with no deviations	2.88	3.47	-	-	-	-
2	Jerky motion along x-axis with no deviations	3.05	8.06	-	-	-	-
3	Linear motion along x-axis with deviations in y-axis	0.15	2.52	2.62	1.62	0.12	0.002
4	Linear motion along x-axis with deviations in z-axis	2.25	3.41	4.29	2.35	0.13	0.038
5	Linear motion along x-axis with deviations in y-axis and z-axis	2.89	3.57	4.22	2.78	0.16	0.138

From the above results it can be noted that the near sensor is more accurate in motion tracking than the far sensor due to its closer proximity to the moving wheel hub that enables it to capture better depth images of the wheel hub. It can also be noted, that the near sensor is able to better track the motion deviations of the wheel hub with lower deviation frequency error than the far sensor. Therefore, there is significantly less lag in tracking motion deviations of the moving hub in the near sensing zone than in the far sensing zone while maintaining the error difference between them. Finally, from the error values of motion pattern 4 and 5, it can be observed that the depth sensors are less accurate in tracking motion in the depth axis (z-axis) than in the other two axes. This phenomenon could be linked to the way in which the sensors calculate the depth values of pixels in the 3D scene by way of interpolation based on the structured light technique (Cruz et al., 2012) rather than absolute depth measurement.

The measurement of wheel hub speed along the direction of motion is critical to determining the position and time at which to load the wheel. In this study, the error in measurement of average wheel hub speed ranges from 0.15mm/s to 8.06mm/s (Table 36).

Contrary to expectation, the far sensor average speed errors are lower than those of the near sensor for all motion patterns. However, on closer observation the error spread along the entire tracked motion is more erratic for the far sensor than that of the near sensor, an example of which is shown in Figure 140.



(a)

(b)

Figure 140: Speed values computed by (a) the far sensor and (b) the near sensor and the laser tracker

Appendix D: Comparison study of Kinect V1 and Kinect V2

Comparison of specifications and features

The specifications and features of the Kinect sensors that are useful in the context of this research are compared in Table 37 where the data is obtained from Duncan (2014) and Smeenk (2015).

Table 37: Comparison between Kinect V1 and Kinect V2

Specification/Feature	**Kinect V1**	**Kinect V2**
Colour (RGB) camera resolution and frame rate	640 x 480 pixels @ 30 frames per second (fps)	1920 x 1080 @ 30 fps
Depth camera resolution and frame rate	320 x 240 @ 30 fps	512 x 424 @ 30 fps
Depth imaging technology	Structured light technique (Cruz et al., 2012)	Time of flight technique (Conde et al., 2014)
Maximum depth distance	≈ 4.5m	≈ 4.5m
Minimum depth distance	40cm	50cm
Field of view (colour camera)	62^o x 48.6^o (≈ 10 x 10 pixels per degree)	84.1^o x 53.8^o (≈ 22 x 20 pixels per degree)
Field of view (depth camera)	58.5^o x 46.6^o (≈ 5 x 5 pixels per degree)	70.6^o x 60^o (≈ 7 x 7 pixels per degree)
No of full skeletons tracked	2	6
No of skeletal joints tracked per person	20 (does not include thumb)	26 (including thumb)

It is evident from the table above that on all but one parameter the Kinect V2 is better than Kinect V1. Enhanced resolutions of colour and depth images result in both bigger viewing areas and also the higher precision of the images themselves improving the colour image detail and depth image detail by a factor of about 4 and 2 respectively. The increased accuracy in depth imaging is also because of the time-of-flight method used by Kinect V2 to retrieve the actual depth values of the pixels as opposed to the structured light method used by the Kinect V1 resulting in interpolated depth values of pixels.

The skeletal motion capture capability of the Kinect V2 is also better than that of the Kinect V1 in terms of higher number of skeletons/persons being tracked simultaneously and the higher number of skeletal joints tracked per person. The increased accuracy of the depth imaging is also anticipated to result in increased reliability and precision of skeletal motion tracking, a claim that is evaluated in the next section.

Comparison of skeletal motion tracking capability

The skeletal motion tracking of the two Kinect sensors was compared in an experiment that measured two performance parameters that are vital to the proposed digitisation framework.

1. **Skeletal tracking accuracy and precision:** the ability of the Kinect sensor to accurately and precisely provide the spatial positions of the human's hands within the 3D scene captured by the sensor.
2. **Skeletal tracking reliability:** the ability of the Kinect sensor to <u>continuously</u> and <u>correctly</u> track the human skeleton throughout the duration of the task being captured. This parameter is measured in terms of the percentage of the total task time that the skeleton was tracked and tracked correctly.

Skeletal tracking experiment setup

In this experiment, the human picks up an object and places it on three pre-defined locations, numbered 1, 2 and 3, on a table, first using the right hand and then the left hand. The X and Y positions of these locations are based on the screen coordinates in pixels whereas the Z position is based on the distance of these locations from the Kinect sensor. The skeletal tracking accuracy could be measured only along the Z-axis because the Kinect measurements could be compared with measurements taken by a measuring tape. The skeletal tracking precision however was measured along all 3 axes. The human's skeleton is tracked by the Kinect sensor, which records the spatial positions of the human's hands throughout the task. The sensor was placed at a height of 1.2m from the floor and the 3 locations; L1, L2 and L3 for object placement were at a distance of 900mm, 1200mm and 1500mm respectively from the sensor (Figure 141). 10 iterations of the experiment were conducted with the Kinect V1 and then with Kinect V2 and the tracking data obtained was analysed.

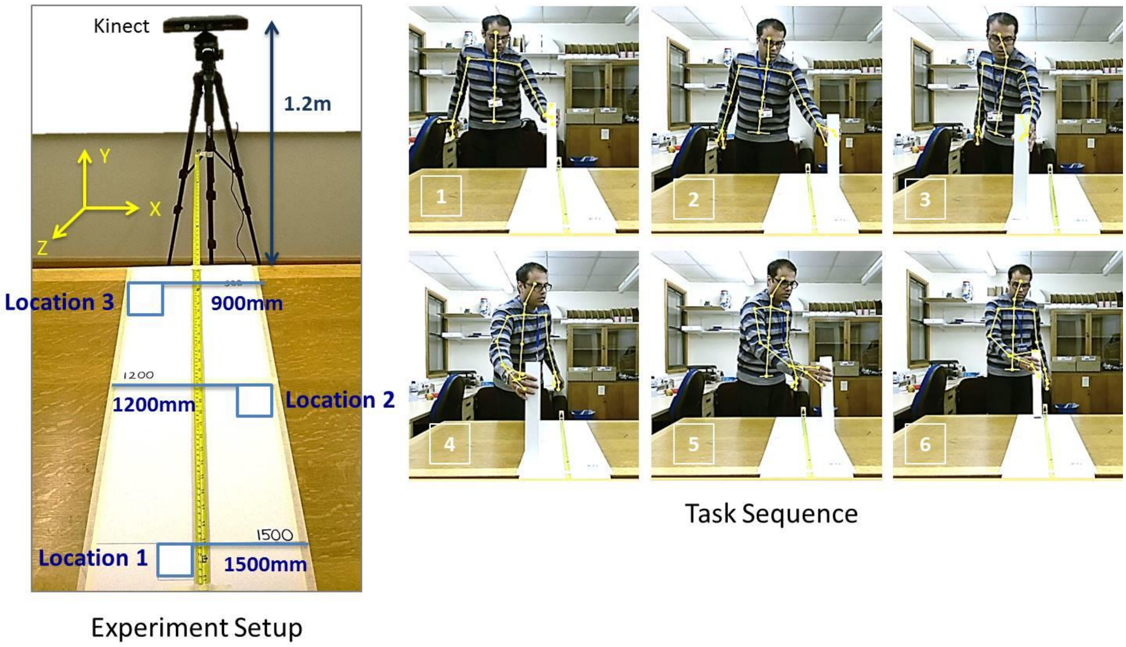


Figure 141: Experiment setup for skeletal tracking comparison between Kinect V1 and V2

Results

Skeletal tracking accuracy and precision

The spatial positions of the human's left and right hand captured by the Kinect sensor for the entire duration of the task are stored in a CSV file without any post-processing. These positions are plotted against the image frame number from which they came in the following motion charts. The charts also show the 3 pre-defined locations at which the objects were placed by the human hands.

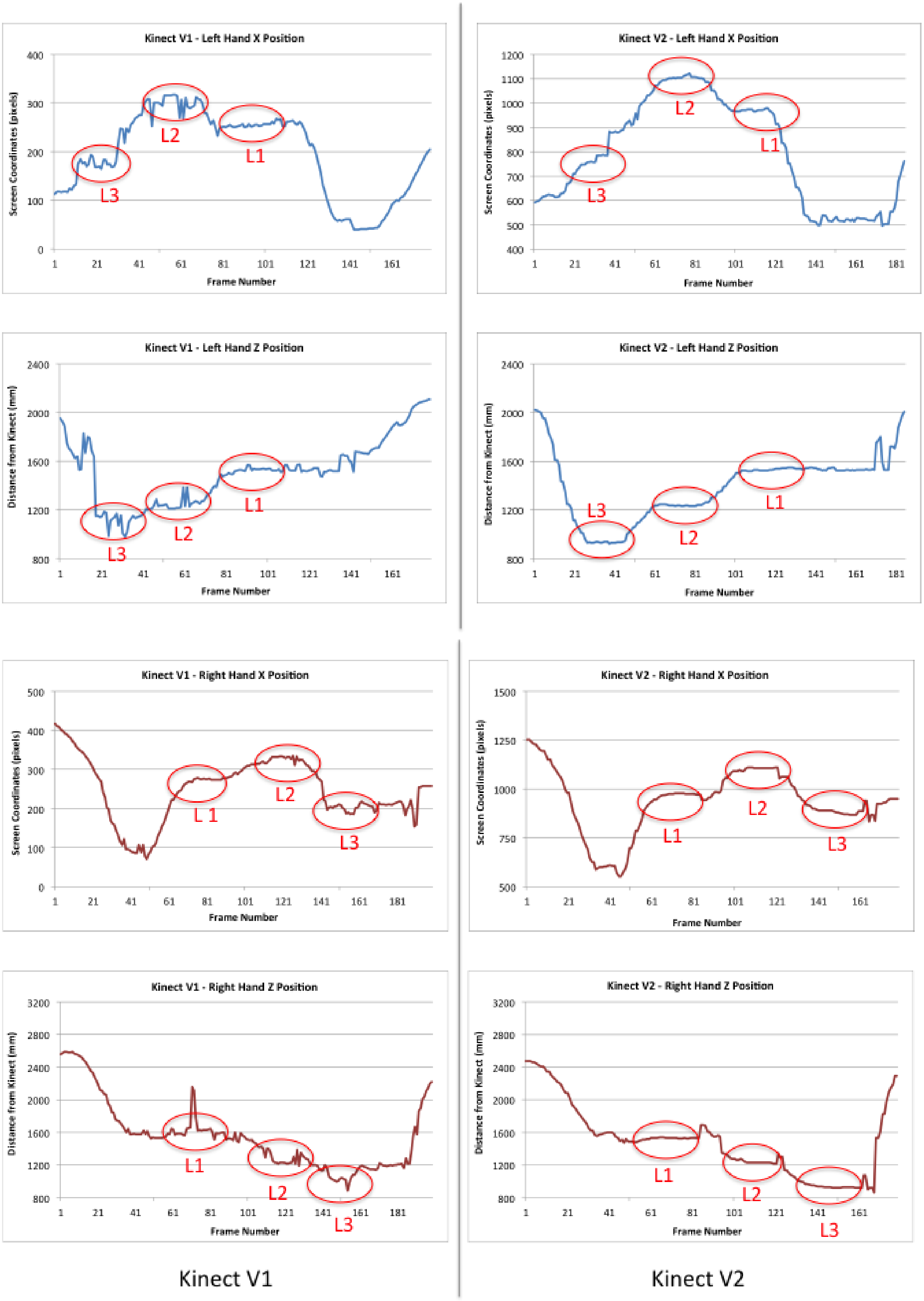


Figure 142: Motion plot of the human hands during the task captured by Kinect V1 (left) and Kinect V2 (right)

From the hand motion plots above, it is evident that the skeletal tracking data provided by Kinect V2 contains significantly less noise then that by Kinect V1 and therefore is more reliable. The z-positions of the left and right hands as obtained from the two Kinect sensors when the human places the object at the 3 pre-defined locations (L1, L2 and L3) is presented in Table 38. The values are averaged across the 10 task iterations and are compared against the known location values to gauge the accuracy of skeletal tracking and the standard deviation of values across 10 iterations is computed to gauge the precision of skeletal tracking. It must be noted that the object is 50mm in width and breadth and therefore for the sake of simplifying the error computation, the hand z-position are compared against the z-positions of the object location plus 50mm.

Table 38: Spatial z-positions of the human hands during object placement

Hand	**Kinect V1 (mm)**			**Kinect V2 (mm)**		
	L1	**L2**	**L3**	**L1**	**L2**	**L3**
Left	1519	1211	910	1540	1237	927
Error	**31**	**39**	**40**	**10**	**13**	**23**
Std. Dev.	**39**	**45**	**51**	**15**	**19**	**22**
Right	1523	1216	915	1537	1239	935
Error	**27**	**34**	**35**	**13**	**11**	**15**
Std. Dev.	**31**	**41**	**45**	**17**	**15**	**19**

From these results it can be noted that the skeletal tracking of Kinect V2 is considerably more accurate and precise than Kinect V1. The above values are for the human's hand joints but the same is likely to be true for the rest of the skeletal joints because from past observations, the hand joint values are the most error and noise prone of all the skeletal joint positions provided by both Kinect V1 and V2 sensors.

Skeletal tracking reliability

This parameter is the ability of the Kinect sensor to continuously and correctly track the human skeleton throughout the duration of the task. To simplify the process of counting the number of times the skeleton was tracked incorrectly, the total task time of 18 seconds is divided into 9 equal intervals of 2 seconds each. The tracked skeleton, which can be seen overlaid on top of the human in the RGB video (Figure 141), is visually monitored during each interval and all intervals that show at least one instance of incorrect tracking are counted as bad intervals. The skeletal tracking reliability is then obtained by calculating the percentage of bad intervals among the total 9 intervals.

Both Kinect V1 and V2 sensors were able to continuously track the human skeleton for all 9 intervals during all 10 iterations of the task. Therefore, the tracking availability percentage is 100% for both the sensors. However, the Kinect V1 sensor correctly tracked the skeleton for an average of only 4 out of the 9 intervals over the 10 iterations of the task, resulting in tracking correctness percentage of 44.4%. Whereas, the Kinect V2 sensor correctly tracked the skeleton for an average of 8 out of the 9 intervals over the 10 iterations of the task, resulting in tracking correctness percentage of 88.8%, which is double that of the Kinect V1 sensor. Therefore, in skeletal tracking accuracy, precision and reliability, the Kinect V2 is superior in performance to Kinect V1.

Comparison of object recognition capability

Object recognition is not an out-of-the-box feature provided by the Kinect sensors. This feature is developed by writing bespoke RGB and depth image processing algorithms that identify a pattern of pixels belonging to objects O1, O2 and O3 in the 3D scene by tracking changes in their RGB or depth values. The quality of object detection of the Kinect V1 and V2 sensors can therefore be compared by analysing the RGB and depth values that are associated with the same object that is placed at the same location with respect to the two sensors.

An experiment was setup for this purpose in which three objects of the same kind were placed at different distances from the Kinect sensor, which is mounted on a tripod at a height of 1.2m from the floor. An imaginary horizontal line is drawn that crosses all the 3 objects at the same height and the RGB and depth values of all the pixels belonging to that line are captured and analysed

(Figure 143). Each experiment was run for 10 iterations and then repeated by changing the Kinect sensor from V1 to V2.

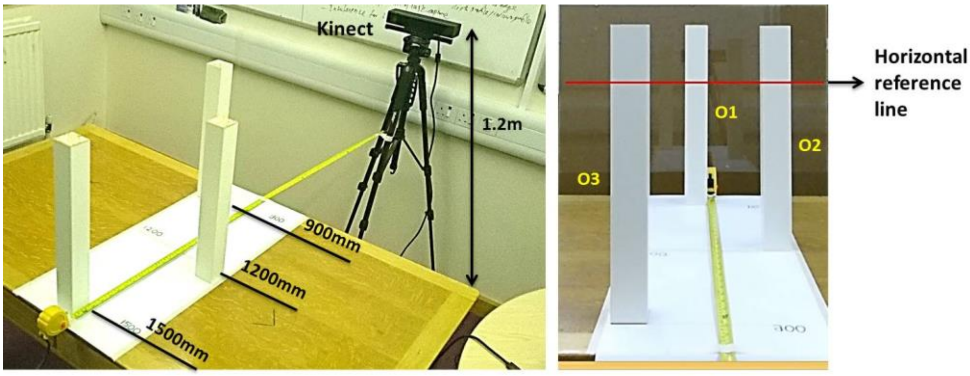


Figure 143: Experiment setup for object recognition comparison

Results

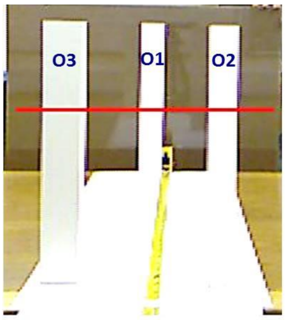
<u>Colour and depth profile</u>

The red, green and blue values of all the pixels on the horizontal reference line are recorded from 10 colour images of the same scene taken by the Kinect V1 and V2 sensors. The depth values of the same pixels were also recorded from the 10 depth images and compared with the actual values. These values and their standard deviations across the 10 readings are tabulated in Table 39.

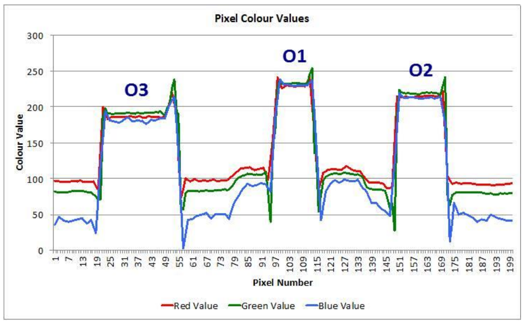
Table 39: Average colour and depth values of objects O1, O2 and O3

Object	Kinect V1				Kinect V2			
	Red (Std. Dev.)	Green (Std. Dev.)	Blue (Std. Dev.)	Depth (Error) (Std. Dev.)	Red (Std. Dev.)	Green (Std. Dev.)	Blue (Std. Dev.)	Depth (Error) (Std. Dev.)
1	215.0 (0.43)	218.2 (0.32)	210.4 (1.00)	1522.5 (22.5) (1.34)	192.0 (0.40)	204.7 (0.39)	208.5 (0.57)	1507.0 (7.0) (1.38)
2	231.2 (0.68)	233.6 (0.67)	230.2 (0.69)	1217.9 (17.9) (1.24)	202.8 (0.74)	214.6 (0.50)	218.3 (0.70)	1195.0 (5.0) (0.71)
3	189.1 (0.54)	194.5 (0.32)	183.7 (1.00)	908.0 (8.0) (0.32)	177.2 (0.65)	189.3 (0.38)	193.0 (0.83)	907.0 (7.0) (0.00)

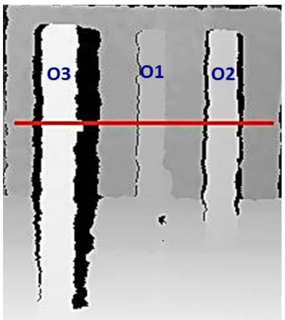
The colour and depth images and the corresponding charts with colour and depth values plotted against the pixel number for the two Kinect sensors are shown in Figure 144.



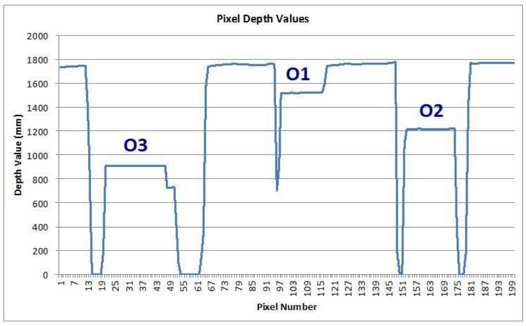
RGB Image



Colour Values of Pixels on the Horizontal Line

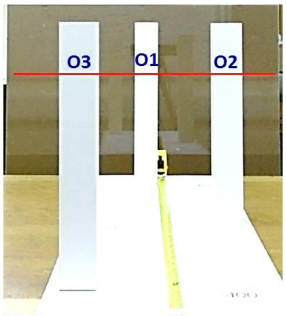


Depth Image

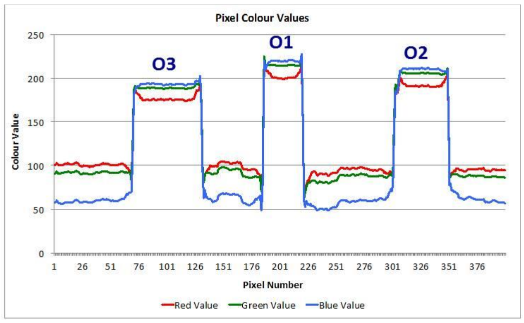


Depth Values of Pixels on the Horizontal Line

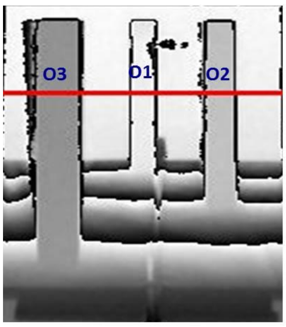
Kinect V1



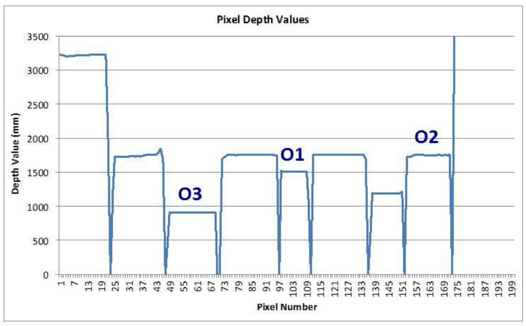
RGB Image



Colour Values of Pixels on the Horizontal Line



Depth Image



Depth Values of Pixels on the Horizontal Line

Kinect V2

Figure 144: Colour and depth profiles of the horizontal reference line

From the above colour and depth profiles, it can be observed that the Kinect V2 performs slightly better than Kinect V1 both in terms of the reproducibility of pixel colour and depth values across the 10 experiment runs as well as the accuracy of the pixel depth values. The images produced by Kinect V2 look sharper and provide a greater level of detail as compared to Kinect V1 due to higher resolution of its colour and depth cameras.

Appendix E: Kinect V2 in the Ikea table assembly case study

A change in the original experiment setup was required to capture this task. The new setup (Figure 145) uses the two versions of the Kinect simultaneously.

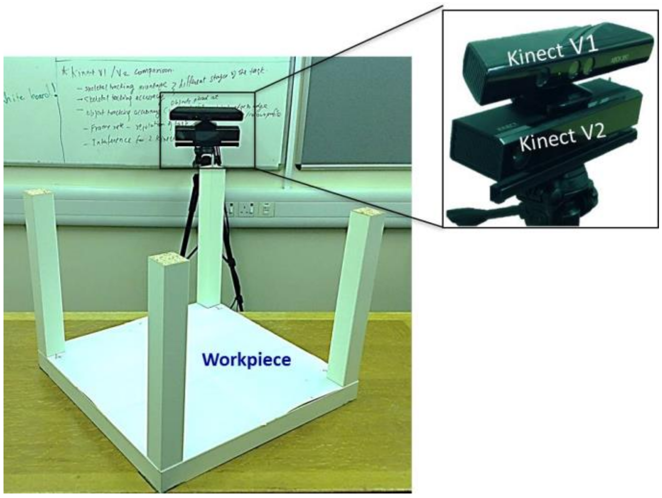
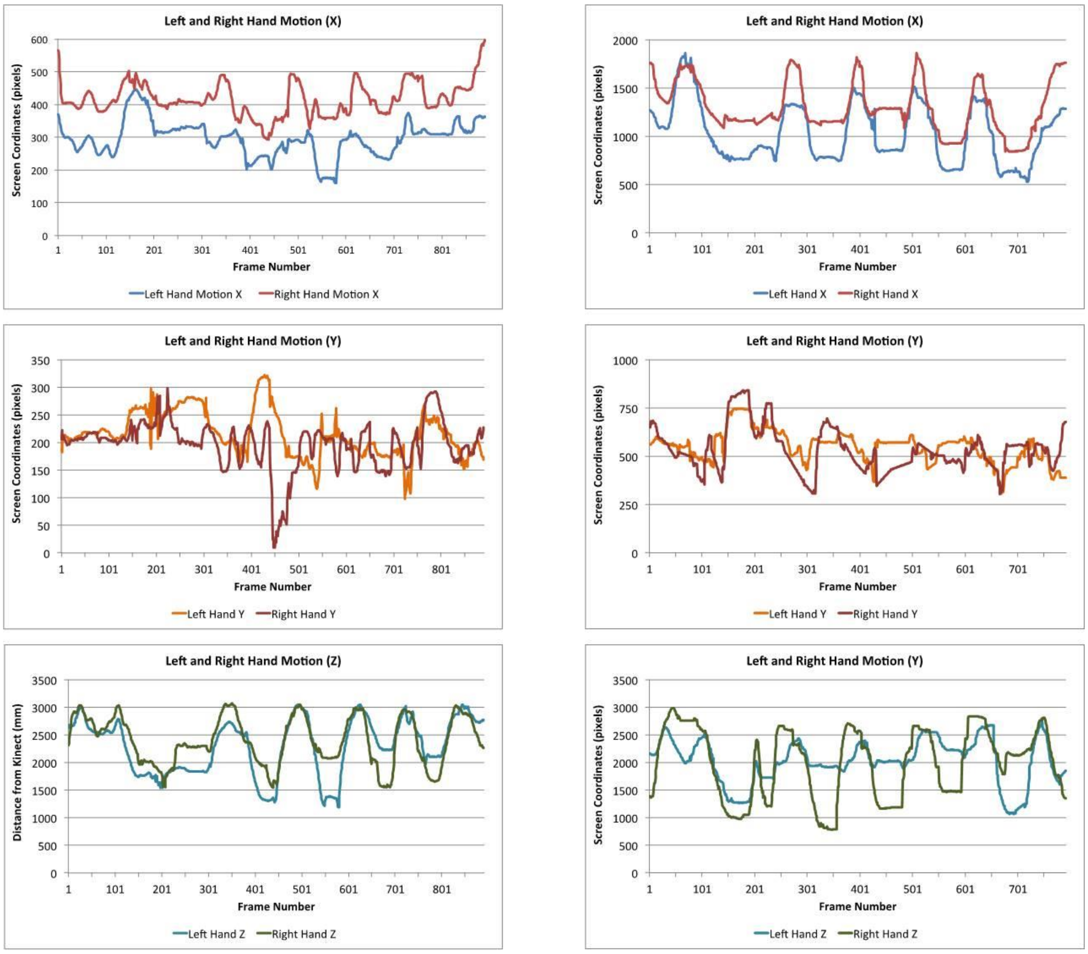


Figure 145: New experiment setup for digitisation of Ikea table assembly task

Only the ‘Capture’ step (first step) of the digitisation framework was repeated with the new experiment setup. The main aim was to obtain better human skeletal tracking data in terms of accuracy and reliability with the Kinect V2 and replace the original Kinect V1 data in the already identified and modelled human action states. No other change in the framework or its implementation in this case study is required as a result of introducing the Kinect V2.

Human action capture with the Kinect V2

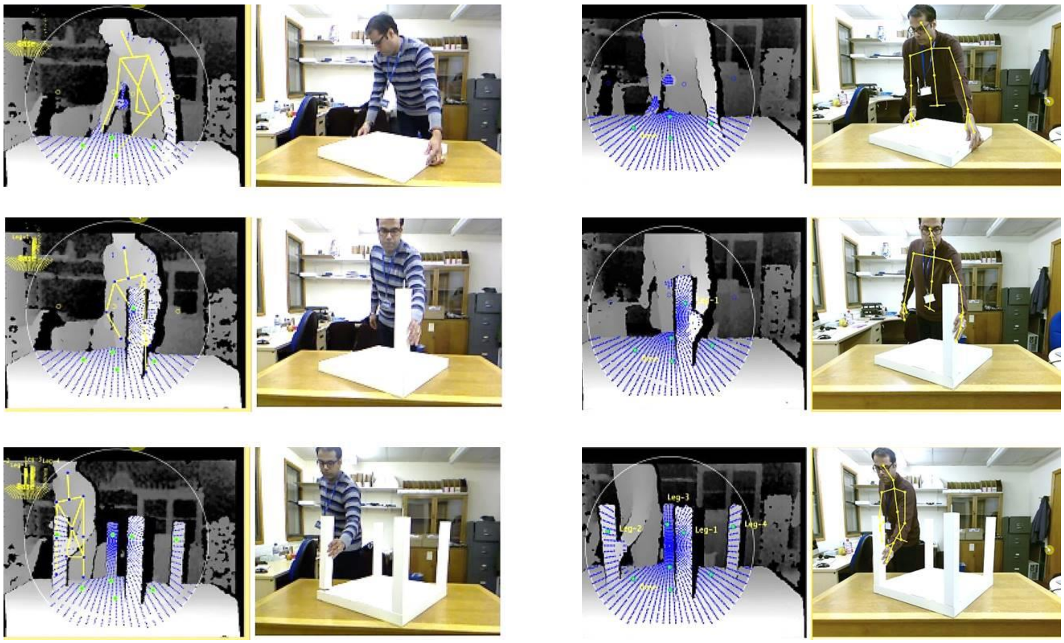
New skeletal joints such as wrist, hand tip and thumb were captured along with the other standard upper body joints by using the Kinect V2. The tracking data for the hand motion is shown in Figure 146 below along with the original Kinect V1 data for visual comparison.



Kinect V1 Kinect V2

Figure 146: Hand motion charts for Kinect V1 and V2

The skeletal tracking data of Kinect V2 is more representative of the actual human action performed during the table assembly task. While this fact may not be clearly inferred from the hand motion charts above, the following images (Figure 147) demonstrate some instances where the Kinect V1 failed to provide correct hand motion tracking unlike the Kinect V2, which correctly tracked the human skeleton for the entire duration of the task.



Kinect V1 (incorrect skeletal tracking) Kinect V2 (correct skeletal tracking)

Figure 147: Skeletal tracking differences between Kinect V1 and V2

Thus the Kinect V2 has proven to be an effective and reliable tool to capture human actions during the assembly task. Kinect V1 continues to be used for workpiece progress tracking until a method is found to correlate the depth image pixels with the colour image pixels within the development platform used in this research.