2016

Robust human computer interaction using dynamic hand gesture recognition

Shuai Yang
University of Wollongong

Robust Human Computer Interaction using Dynamic Hand Gesture Recognition

A thesis submitted in partial fulfilment of the requirements for the award of the degree

Doctor of Philosophy

from

UNIVERSITY OF WOLLONGONG

by

Shuai Yang

Bachelor of Engineering (Telecommunication)

School of Electrical, Computer and Telecommunications Engineering

July 2016

Statement of Originality

I, Shuai Yang, declare that this thesis, submitted in partial fulfilment of the requirements for the award of Doctor of Philosophy, in the School of Electrical, Computer and Telecommunications Engineering, University of Wollongong, is wholly my own work unless otherwise referenced or acknowledged. The document has not been submitted for qualifications at any other academic institution.

Signed

Shuai Yang

July 13, 2016

Abstract

Hand gesture recognition has been applied to many fields in recent years, especially in man-machine interaction (MMI) area, which is regarded as a more natural and flexible input than the traditional input, such as, mice and keyboard. Microsoft Kinect camera has also drastically changed the world of human computer interaction based computer vision, due to its low cost and high quality of depth information for visual images. This has made the depth data to become common place at a very low cost allowing myriad of computer vision related applications including hand gesture recognition. Hand gesture recognition research suffered severely from the clutter and skintone regions in any background. With the availability of depth information, background clutter and skintone regions which are not part of the hand gesture can be removed improving the performance of any classification strategy. In this thesis, an overview of hand gesture recognition research up to date is presented, which includes common stages of hand gesture recognition, common methods and technique of each stage, the state of the recent research and summaries of some successful hand gesture recognition models. This article also discusses a novel hand detection strategy based on Kinect camera by combining depth and colour image information. In the detection procedure, the Kalman filter is applied to tracking process to achieve a good detection result. The experiment results in chapter 3 show this detection method is reliable and

stable in the clutter background, and works well in various light conditions.

Gesture recognition is an important and challenging task in the field of computer vision. Starting from the 3D shape of coding gestures, it puts forward a new kind of gesture recognition framework based on depth image. It extracts the space characteristics of a variety of 3D point cloud based on Kinect, including local principal components analysis on point cloud to get the histogram of main component, gradient direction histogram based on local depth difference and depth distribution histogram of local point cloud. Principal component histogram and gradient direction histogram effectively coding the local shape of gestures, depth distribution histogram compensates the loss of the shaping descriptor information. In this thesis, through preliminary training of random forest classifier to filter the characteristics, and characteristics with less influence on classification results are removed, thus the computational costs are reduced. The filtered characteristics are used for training of random forest classifier again to classify gestures. the experiment is carried on two large-scale gesture data sets,which is shown in this thesis, for more difficult ASL dataset, the proposed method has improved the recognition rate of 3.6% then the previous algorithm. This thesis shows a good prospect of hand gesture recognition based its high recognition accuracy and speed.

Acknowledgments

First of all, I would express my deepest gratitude to my supervisor, Dr. Prashan Premaratne, for his patient guidance, constructive suggestions and constant encouragements through my studies. He is a very nice person, we talk about everything, research plan, experiments, thesis, even the life in future. Dr. Prashan has set a good example for me in both research and life.

Special thanks goes to my Co-supervisor Dr. Peter Vial, who gave me a lot of help with both my study and life. Thanks to my friends, Dr. Changlin Yang and Dr. HeWang, they offered many brilliant suggestions to me on my thesis. I wish they all the best.

Finally, I wish to express my sincere gratitude to my family for their great support and love throughout my life. Without their help, it is impossible for me to reach this stage.

Contents

List of Figures

List of Tables

Abbreviations

2D	2 Dimensional
3D	3 Dimensional
ASL	Australian Sign Language
ASL	American Sign language
CMOS	Complementary Metal-Oxide Semiconductor
COG	Center Of Gravity
DTW	Dynamic Time Warping
HCI	Human Computer Interaction
HMI	Human Machine Interaction
HMM	Hidden Markov Model
HOG	Histogram of Oriented Gradient
HSI	Hue, Saturation, Intensity
HSL	Hue, Saturation, Lightness
HSV	Hue, Saturation, Value
ICP	Iterative Closest Point
GM	Gaussian Model
GMM	Gaussian Mixture Model
GPDF	Gaussian Probability Distribution Function
MEMS	Micro-Electro-Mechanical Systems

MM	Markov Model
MMC	Managing Massive City
NN	Neural Network
PDF	Probability Distribution Function
RGB	Red, Green, Blue
SDK	Software Development Kit
SGONG	Self-Growing and Self-Organized Neural Gas
TOF	Time Of Flight
VR	Virtual Reality

Chapter 1

Introduction

1.1 Background

Gesturing, in addition to speech, is an important facet of human communication. Gestures are a common aspect of communication for many individuals in their daily lives: military commanders issue orders through gesture, athletes issue and respond to gestures during sport, and gestures are essential modes of communication in industries where speech is not possible (e.g., during operations that take place underwater.) For individuals with certain disabilities, gestures are the only available method of self-expression [1].

Alongside rapid advancements in computer and consumer electronics technologies, telecommunications, and software applications in recent years, the first generation of human-machine interactions (e.g., the traditional keyboard, mouse, data gloves or other early haptic technologies,) have become increasingly obsolete [2]. These types of contact devices tend to create unnatural experiences with the technology and may require that the user familiarise him or herself with rather complex input rules. In response to this issue, intelligent, multimedia-

based human-computer interaction has emerged as a very popular research field within which human gesture recognition is a hot topic; research in this area relates to iPhone touch sensing, Kinect somatosensory gaming, and other consumer electronics which utilise hand gesture recognition technologies [3].

The main objective of intelligent human-computer interaction is to establish an intuitive, navigable, and interactive human-computer learning environment to which the user provides input easily and from which he or she receives the technologys output as desired [4]. Hand gestures, a basic biometrics feature, represent indispensable recognition technology for intelligent human-computer interaction.

Currently existing hand gesture recognition technologies are mainly hardware-based, i.e., dependent on sensors or visual information the former requires data gloves or other hardware devices, (inherently at the users inconvenience,) and is limited by cost and other unfavourable factors; the latter directly uses the human hand as an input device, and thus provides the user with a more natural, direct, and unconstrained human-machine interactive environment. Visual input to the system lends the user more freedom and rewards him or her with a relatively very realistic interaction experience, making visual hand gesture recognition a potentially very valuable technology.

In fact, visual hand gesture recognition technology is a very popular research topic. Significant achievements have been made in this field, but have been very hard-won due to the fact that gesture recognition research involves artificial intelligence, pattern recognition, probability statistics, computational linguistics [5], and other disciplines of computer vision, image processing, and analysis additionally, each gesture itself is diverse, ambiguous, and unpredictable in both time and space. In short, hand gesture recognition is a challenging and interdisci-

plinary endeavour, and a significant challenge for human-computer interaction developers.

1.2 Hand Gesture Recognition Overview

Any one hand gesture is practically impossible to define, as gestures differ in time and space and across different cultural backgrounds. Broadly speaking, any gesture is a conscious hand movement that includes motion of the fingers, palm and wrist intended to convey information. In the field of gesture recognition, gestures are considered to extend to the arms as well as the hands when used to produce a variety of postures or movements [6].

A computer or other machine which recognises human gestures is one that can be effectively controlled by the movement of the hands and arms of the user [7]. Gestures come naturally to human beings, as they are an important part of the way we communicate, so the use of gestures for human-computer interactions is very easy to learn. Ideally, users can command machines to complete complex tasks using a single posture or relatively simple, continuous, dynamic hand gestures. There are many ways to classify hand gesture recognition, several of which are discussed individually below [1].

1. ***Static gesture recognition and dynamic hand gesture recognition.***

   Performing a gesture is an active process which takes place over some amount of time, but a hand posture is presented instantly accordingly, hand

gesture recognition can be considered either static or dynamic. Static gesture recognition instantly synthesises the relative position of the hands and arms at a given time point, dynamic gesture recognition reads the changing positions of the hands and arms over a certain continuous time frame [8]. The figures below show static gestures (Fig. 1.1) and dynamic hand gestures (Fig. 1.2).

Figure 1.1: Static gestures

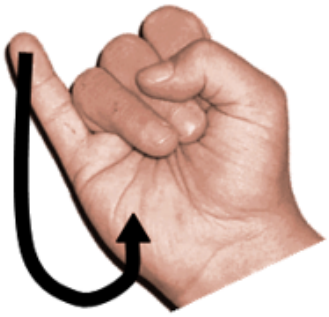

Figure 1.2: Dynamic gestures (use hand or finger to make a motion trajectory)

2. ***Non-vision based hand gesture recognition and vision based hand gesture recognition.***

Gesture recognition image acquisition methods likewise can be divided into two categories: motion-sensor-based gesture recognition, and vision-based gesture recognition. Motion-sensor-based systems include contact devices such as mice, styluses, and data gloves [9]. The advantages of device-based gesture recognition systems is touch pads or software algorithms can be integrated to make the system nearly universally applicable; but such systems cannot recognise nuanced finger movements, (in addition to the added inconvenience for the user, as discussed above.) The data glove gesture recognition system, for example, uses motion sensors to monitor movements and time information of several points on the device; real-time gesture recognition is thus possible due to high input speed, high recognition rate, and three-dimensional response to the human hand in space. The gloves are cumbersome and expensive, however, and altogether not appropriate for the typical consumer.

Vision-based gesture recognition systems typically involve several processes: image acquisition and pre-processing, gesture segmentation, gesture modelling, and gesture recognition [10]. Hand gestures are input to a camera which checks, tracks, and analyses the gestures, after which the associated machine reads the gestures from the camera frame to define certain features.

According to said features, the system then selects the appropriate classification of the gesture to define it appropriately. Static gesture recognition reads only one image and does not require trajectory information to extract its

features, while a full sequence of gestures must be read including tracking and segmenting the frames as necessary to realise dynamic gesture recognition. extraction of features does not include trajectory information. System-generated descriptions ultimately drive the specific applications which are controlled by the gestures [1].

Hand gesture recognition requires minimal user training; ideally, the machine reads users intention automatically to complete the communication between itself and the user seemingly instinctively. Hand gesture recognition, (which, again, is a quite popular research area,) can be expected to have a more extensive range of applications in coming years mainly in the following aspects.

In virtual environments

The hand gesture recognition system is, in effect, an interactive device which can be used to simulate and control complex processes such as virtual manufacturing and assembly [11], driving training, product design, and even virtual training for complex processes such as surgery.

In multimedia user interfaces

Hand gesture recognition applications also are daily consumer goods which can be built into a variety of smart devices to enhance user experiences, such as TV controllers [12], robotics, and small remote aircraft and other toys. Intelligent human-computer interaction systems are also commonly understood to represent next-generation mobile communications equipment in cars, homes, and workplaces.

In sign language applications

Communication typically occurs silently in the human-computer interaction interface, which may benefit humans who communicate silently, e.g., through sign

language. Hand gesture recognition systems also may serve as a bridge between sign language speakers and non-speakers, allowing signers to speak with others using an electronic device in other words, a portable, unobtrusive, and practically infinitely knowledgeable translator.

1.2.1 Basic Hand Gesture Recognition Process

The process through which a computer vision-based system recognises hand gestures can be divided into four stages [13]. In the first stage, one or multiple cameras obtain image data, then according to the data model, check for hand gesture input data in the stream. Once the computer detects that a hand gesture is present, segmentation is employed to derive the posture only and remove any background; the posture information is then used employed in the feature extraction stage, where it is classified appropriately (i.e., the ultimate goal of the gesture recognition process). During the identification or classification stage, according to the model parameters, the system classifies the hand gestures as-received to generate the appropriate description. Finally, based on said description, the system drives the specific application per the users request.

1. ***Hand Gestures Segmentation***

   The smooth surface of the human hand readily creates highlights and shadows under any light source; this characteristic plus background interference leaves no particularly efficient way to manually segment gestures from the background. There are several currently existing approaches to gesture segmentation, however, including changing the contrast between the hand and

the background (through gloves or environmental backdrops, for example,) which can work but at the expense of limiting the freedom of hand movements [1]. A simple hand gesture segmentation is shown below in Fig. 1.3.

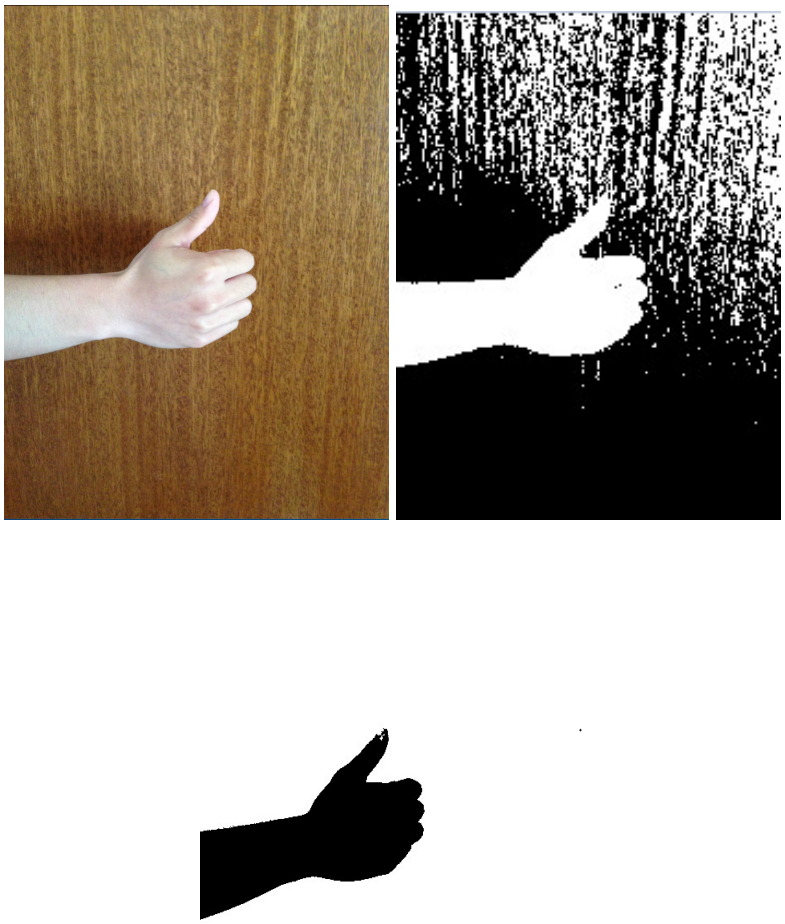

Figure 1.3: a simple segmentation process

Hand gesture databases can also be built to include different time, shadow, angles, and luminance conditions to match the input hand gestures with a variety templates, but this method is quite time-consuming [14]. Contour tracking, a common method based on Snake model hand gesture segmentation, applies the Snake model to track the movement and sophisticated shape changes of the object (i.e., the hand) by exploiting the relation be-

tween the background noise and contract ratio. The background difference method, i.e., background subtraction, can eliminate background images to make the target image more clear, but results in substantial error if a pixel corresponding to the background image and hand gesture space is located in the same target point. The colour model which works based on skin colour information can also be used to separate hand gestures from the background by placing the image of the hand in appropriate colour spaces. This method was explored extensively by previous researchers Chai and Nagan [15], and Habili and Lim [16].

2. ***Hand Gestures Modelling***

Currently existing gesture modelling methods fall into two categories: the first based on the gesture modelling performance, and the other based on 3D modelling.

Performance-based gesture modelling analyses the performance characteristics of the images of hand and arm movements in a sequence to build a gesture model. This category includes a few subcategories: These include the use of grey scale images to build the hand gesture model, the use of whole hand information, or the use of the movement image as simply a template. Hand and arm gestures can also be used to establish a deformable template model comprised of a collection of certain points on the body contour this method can provide difference symbol points to create approximate hand gesture contours. Image attributes, including the outline of the properties extracted from the image, image moments, eigenvectors, and regional characteristics of the image histogram parameters can also be utilised to establish the gesture model.

The 3D information-based approach aims to perfectly restore the 3D information associated with the original, natural hand movement as-performed by the user [17]. The performance-based approach discussed above cannot restore the 3D hand movement information, but instead creates a 2D projection of the 3D hand movement, and hence inherently loses a part of the gesture information. In practical applications, 3D gesture modelling requires more parameters and comes with higher computational complexity than performance-based modelling [18]; further, the feature extraction process can cause distortion of the model parameters, so to ensure relatively simple calculations and high system identification efficiency, most gesture recognition models are designed based on hand movement performance.

3. ***Hand Gesture Feature Detection***

The gesture feature detection process involves both locating and extracting certain features of gestures, and relevant approaches to the process can be divided into three categories: colour-based locating, movement based locating, and multi-mode locating technologies. The multi-mode locating approach is also called multi-cue locating [19], as it combines colour, movement, and other visual cues as its target as opposed to a single target cue. The parameters and basic elements of any of these computation models are similar regardless of which approach is utilised all are intended to detect hand gesture features, of course and typically target a combination of grey scale images, binary images (Upper left image in Fig. 1.4), boundaries, contours (Upper right image in Fig. 1.4), finger position and palm center (Lower image in Fig. 1.4).

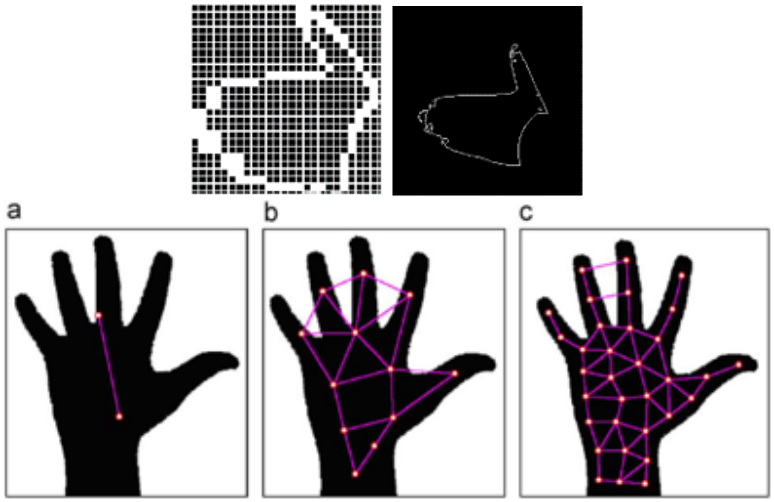

Figure 1.4: Three images are the results while using binary image, hand contours, and palm center feature extraction method, respectively.

4. ***Hand Gestures Recognition***

Gesture recognition involves classifying the point or trajectory in the model space to a certain subset of the space. The hand gesture recognition process, as mentioned above, can be either static or dynamic: static recognition identifies postures and associates them with meaning, while dynamic recognition also recognises the trajectory of postures and motions and associates them with meaning.

Static gestures are relatively simple and distinguishable. In a sense, static gesture recognition matches the gesture to the existing template with no relation to the time or time frame in which the gesture was made. Static hand gesture recognition is strongly dependent on hand feature extraction, including image attributes such as contour, margin, image moments, image feature vectors, histogram regions, and others [1]. The main approaches to static gesture recognition include the elastic graph matching method,

Hausdorff distance template matching, and others based on statistical and neural network methods.

Dynamic gesture recognition requires that temporal and spatial information be gathered in addition to the users hand and arm positions. The vast majority of dynamic gestures in the parameter space occur on a track: Different users perform gestures at different speeds, creating a non-linear wave track on the timeline. Many researchers have addressed the issue of eliminating these non-linear dynamic fluctuations. According to different treatments on the timeline, the existing dynamic gesture recognition technologies can be divided into three categories: Hidden Markov Model (HMM) recognition, Dynamic Time Warping (DTW) recognition, and the timeline compression approach [20].

1.3 Problem Space and Motivation

Hand gesture recognition has been applied in the human-computer interaction (HCI) field for some time [21], particularly as a method of communication for sign language speakers [22]. Compared to traditional HCI inputs such as keyboards and mice, hand gestures are more natural and flexible; they represent considerable potential in terms of digital control, real-time input, and communication among users with certain disabilities [23]. In recent years, vision-based hand gesture recognition techniques have grown increasingly popular compared to contact-type techniques, which can diminish hand movement flexibility. The rapid and extensive development of many hardware devices, such as smart phones, has allowed computer vision the opportunity to play an important role in HCI [24].

Though the research on vision-based hand gesture recognition has been extensive and many valuable achievements have been made, to date, the reliability and practicality of these hand gesture recognition systems are still problematic. The biggest challenge in advancing the field lies in reliably tracking hand movements in order to fully recognise dynamic gestures under complex lighting conditions and background clutter, as mentioned above. Among the wide variety of vision-based hand gesture recognition methods, some can achieve very good recognition rates in certain restrictive environments, but may not be applicable yet to real-world situations. Remaining challenges include the correct segmentation of input images, proper feature extraction, and quick and accurate gesture classification.

Background interference which occurs during the segmentation process, such as lighting, brightness, similar colours, overlapping areas, or similar objects in the background, can generate very divergent segmentation results. The human eye is able to distinguish foreground and background very easily while machines simply cannot; if the machine does not have a sufficient recognition rate (i.e., one comparable to the human eye,) and very fast reaction speed, the application will be restricted considerably. For ordinary web cameras, for example, segmentation results tend to be poor due to issues with colour-based, motion-based, and depth-based information extraction [25].

There are many other issues yet to be solved in terms of the applicability of these technologies. For one, the gesture model must first be established appropriately before proper gesture identification is possible, and there currently exists no perfect method of doing so. The conversion from 3D gestures to 2D images, further, may tax the systems computation ability. Also, as discussed above, the system must locate and track trajectories correctly to realise dynamic gesture recognition, which represents a quite complex process [26]. An appropriate recognition

model also must have sufficiently high recognition efficiency without unnecessary computation complexity, (i.e., high reaction speed,) to be practically useful.

In fact, dynamic hand gesture recognition is especially problematic and especially in terms of recognition rate. Motion blur and quiescent camera conditions pose problems: the first occurs while tracking a dynamic gesture in motion, mainly due to energy accumulation in the imaging process, and the second is caused by the movement of the camera itself during the process of filming [27]. Changes in target size, which occur when the distance between the hand and camera changes continuously, can also create changes in the tracked target gesture and thus deviation in the tracking results. Environmental factors, (e.g., light, background colour, and skin colour,) can also cause interference which directly affects tracking accuracy or even lead to tracking failure. Speed changes or rapid changes in the direction of movement can cause the subsequent frame to be predicted inaccurately. Most tracking algorithms fail to meet real-time requirements; stable and accurate algorithms generally have high complexity that is not suited to the real-time requirements of a practical tracking algorithm. Gesture occlusion is also a problem due to the natural motion of the hand, certain gestures inherently include partial occlusion which affects the systems recognition accuracy [26].

1.4 Contributions

In this thesis, all the above problems are carefully addressed. The main contributions, in terms of algorithms developed to address these problems, the details of the said contributions are as follows.

Hand gesture recognition has been applied to many fields in recent years, especially the HCI field, as it is considered a more natural and flexible manner of input than traditional devices. My thesis includes an overview of hand gesture recognition research to date, including the common stages of hand gesture recognition, common methods and techniques employed in each stage, the state of the recent research, and summaries of a few relatively successful hand gesture recognition models. There also is a compulsory description of the concept of hand gesture recognition, which includes key points on static and dynamic gesture recognition, remaining challenges to further development of these technologies, comments on each stage of the recognition process, and the drawbacks and advantages of each mainstream hand gesture recognition system.

Overlapping skin colour areas, such as an arm and face in the same input image, sizable affect the recognition rate of vision-based hand gesture recognition systems. Because these recognition systems generally use colour modelling methods to classify the foreground and background, neighbouring pixels with similar colour in any space are placed into one category. As discussed above, there are ways to minimise skin colour area interference (wearing gloves, etc.) but they are not particularly user-friendly. This thesis outlines a relatively simple method of eliminating skin colour area overlapping which relies on camera depth to accurately classify the foreground and background without colour interference. The associated algorithms, the colour model thresholding strategy (or depth distribution histogram,) require a relatively pure environment to gather useful gesture information and segment input images accurately; this unfortunately means a high level of computation and time consumption.

This thesis explores a thresholding strategy that the author believes can solve the above problem. Specifically, the algorithm was designed to allow the hand gesture system to achieve accurate segmentation relatively quickly and at the cost

of relatively little computation complexity.

In this thesis, the author outlines a model which integrates depth and colour information to segment necessary gestural information step-by-step, where each step contains a few (fairly simple) restraints. The main idea of the proposed approach is the use of a colour modelling method to roughly define the gesture area, followed by the use of depth information to distinguish the skin colour or noise areas, and finally the colour model again to accurately determine the gesture area. Simulation results showed that this method uses less time and less complex computation to achieve accurate segmentation outputs compared to traditional methods.

A key assumption of hand gesture recognition is that the system is able to directly realise gesture segmentation. To relax this assumption, in this thesis, the author shows that the proposed step-by-step process eliminates the need to directly define the gesture area, which achieves a corresponding output that allows the system to accurately detect hand gestures.

To solve the recognition problem, the author proposes a Kalman-filter-based fusion of depth and colour information to overcome the restraints typically present in the hand gesture recognition environment. This robust system employs the colour model with depth information as input and segmentation methods, then the Kalman filter as a hand gesture detection tool. Assisted by the Kalman filter, the system can detect input gestures by predicting the next movement from the previous movement; this reduces recognition failure when any parts of the input gesture frames are lost, and forward-senses the data until coverage and connectivity are guaranteed.

In addition, thesis also contains a discussion on a thresholding and recognition

algorithm developed for hand gesture recognition through which depth information helps the user to interact with the system even in a high-interference environment. Simulation results showed that this robust system can work under different environments, as it is strongly tolerant of noise, complex lighting conditions, and background clutter.

Kinect technology is also applied to the proposed system to enhance gesture recognition rate, specifically, a centralised algorithm is presented which allows Kinect to function correctly within the whole HCI environment and to take advantage of its high reaction speed and accurate sensor (which includes depth information.) Due to the low cost and intuitiveness of Kinect, this system is also affordable and user-friendly to implement.

To summarise, the proposed gesture recognition method works based on multiple spatial characteristics obtained through Kinect sensor data and elegant algorithms which synthesise said data. A principal component histogram and gradient direction histogram describe the shape of gestures in different scales, and a depth distribution histogram embodies the depth distribution of the gestures. Accordingly, the author calculated the importance of certain gestures through the preliminary training of random forests and filtered characteristics in an experiment on two large-scale gesture data sets. The results showed that compared to pre-existing gesture recognition algorithms, the integrated hand gesture recognition method proposed in this thesis effectively improves gesture recognition effects.

1.5 Publications

The work in this thesis has resulted in the following papers:

1. **S. Yang**, P. Premaratne, P. Vial and Q. Alshebani *Robust hand gesture detection by fusion of depth and colour information using kinect*, Computer Modeling and New Technologies, Vol 18(12B) 127-132, 2014.

2. **S. Yang**, P. Premaratne and P. Vial. *Hand gesture recognition: An overview*, 5th IEEE International Conference on Broadband Network and Multimedia Technology (IC-BNMT 2013), pp. 63-69., 2013.

3. **S. Yang** and P. Premaratne. *Dynamic gesture recognition method of Kinect fusion fast entropy SVM*, *under review*

4. P. Premaratne, **S. Yang**, Z. Zou and N. Bandara. *Dynamic Hand Gesture Recognition Framework*, Intelligent Computing Methodologies, Springer International, pp834-845, 2014.

5. P. Premaratne, **S. Yang**, P. Vial and Z. Iftikhar. *Dynamic Hand Gesture Recognition using Centroid Tracking*, Intelligent Computing Theories and Methodologies, Lecture Notes in Computer Science, Springer International, pp623-629, 2015.

6. P. Premaratne, **S. Yang**, P. Vial and Z. Iftikhar. *Centroid Tracking Based Dynamic Hand Gesture Recognition using Discrete Hidden Markov Models*, Neurocomputing Journal, Accepted the final version in March 2016.

7. Q. Al-shebani, P. Premaratne, P. Vial and **S. Yang**. *The Feasibility of Implementing a Face Recognition System Based on a Gabor Filter and Nearest Neighbor Techniques*, FPGA Device for Door Control Systems. Journal of Computers, 10-2. pp115-129, 2015.

8. Z. Iftikhar, P. Premaratne, P. Vial and **S. Yang**. *Robust Segmentation of Vehicles under Illumination Variations and Camera Movement*, Accepted for Springer Lecture Notes in Computer Science, 2015.

1.6 Thesis Structure

1. *Chapter 2.* This chapter includes a literature review of existing hand gestures recognition approaches in both conventional static gestures recognition and dynamic gestures recognition.

2. *Chapter 3.* This chapter proposes the fusion algorithm to detect hand gestures using depth and colour information.

3. *Chapter 4.* This chapter presents the fast entropy SVM algorithm to address the dynamic hand gesture problem,

4. *Chapter 5.* This chapter concludes the thesis, and provides a summary of research outcomes .

Chapter 2

Literature Review

This chapter reviews prior works on hand gesture recognition. The focus will be on state of art of basic and latest hand gesture recognition model, the description also includes some new hand gesture devices, such as Time of Flight (TOF) camera (shown in Fig. 2.1)and Kinect.

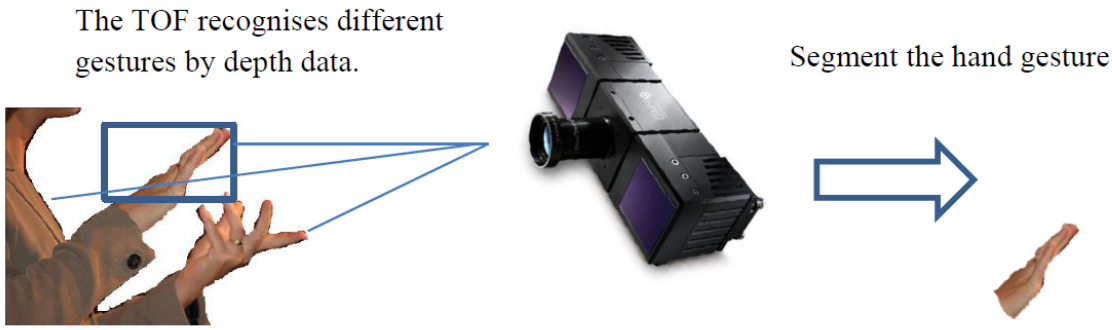

Figure 2.1: Time of Flight (TOF) camera

2.1 non-vision Based Hand Gesture Recognition

According to different approaches of acquiring hand gesture information, hand gesture recognition can be divided into two groups; one is non-vision based recognition, such as data gloves, shown in Fig. 2.2, and another is vision based recognition. As hand is a deformable object, it cannot be represented by one simple model, besides, human hand tracking and recognition are easily influenced by environment factors such as, luminance, colour and so on, so many previous research are about non-vision based recognition, especially on data glove devices [28].

Figure 2.2: One typical data glove called Immersion CyberGrasp

Data glove (Shown in Fig. 2.2) is a multi-function Virtual-Reality (VR) device comprising of many sensors on the glove. Through software mapping, the glove can reach into the computer to move, clutch and rotate the virtual objects. The latest release of this product is capable of registering finger bends for each finger. The glove accurately transmits hand gesture to the computer in real time, and then receives feedback from virtual environment to the operator. It provides the user a direct and universal human-computer interaction mode [28].

Ruize Xu et al. proposed a method based on data glove theory using a contact type

sensor, MEMS 3-axes accelerometer (shown in Fig. 2.3) [29] to recognise seven hand gestures that includes up, down, left, right, tick, circle and cross. The hand motions are captured by the accelerometer in three vertical directions and then the data were transferred to a computer via Bluetooth. Before features extraction process, the system will segment the hand gesture from the input data. Because of the complexity of the gestures from different people, an 8 digit sign sequence is extracted as common features using the accelerometer. The system recognises gestures by comparing the templates in the database.

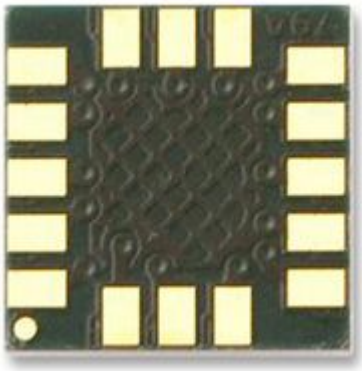

Figure 2.3: MEMS 3-axes accelerometer
Source: http://au.element14.com

This method has two constraints, one is this system requires people to wear sensors sacrificing natural feeling of hand gestures; another is that this system strikes a balance between accuracy and number of gestures by resorting to only seven gestures. Because it is only based on the MEMS accelerometers, the accuracy is acceptable when the database is small. If this system is trained on more hand gestures, the accuracy would decrease due to erroneous classification. The experiment shows the recognition rate of this model is 95.6%, the recognition of each gesture range from 91%-100%.

2.2 Vision Based Hand Gesture Recognition

In recent years, more and more research has concentrated on vision based hand gesture recognition. Compared to non-vision based recognition (data glove or electro-magnetic waves etc.), as contact type devices reduces the flexibility of hand movements, vision based recognition are more natural and comfortable for the user. Based on the data glove and electromagnetic waves, researchers developed a new kind of colour glove (also called colour makers) (in Fig. 2.4) [30]and one non-contact optical sensor chip used for hand gesture recognition.

Figure 2.4: Colour Glove

2.3 Computer Vision Based Hand Gesture Recognition

Getting a computer to accurately interpret a human hand gesture or posture is a non-trivial task. With the rapid development of computer hardware, the computational ability of a computer has achieved tremendous growth over the last

decade. This has facilitated a computer to be used for HCI giving people freedom to naturally and flexibly input information. In the latest research, computer vision-based hand gesture recognition plays a very important role in the human computer interaction (HCI) area [31].

A computer vision-based hand gesture recognition system can be divided into four parts (Fig. 2.5) [1]

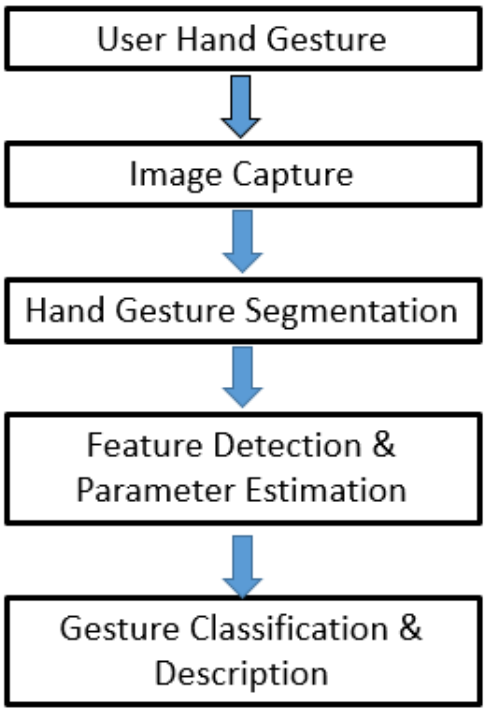

Figure 2.5: Vision- based hand gesture recognition system

First stage uses one or multiple cameras to obtain image data, and then according to the data model, check the input data stream if it has hand gesture information. Once the computer detects that a hand gesture is present, segmentation is used to derive the posture only removing any background. This is then used for feature extraction stage which will be used for classification as the final goal of the process. During identification or classification stage, according to parameters of

the model, system classifies the received hand gestures to generate hand gesture description. Finally, based on the description, the system drives the specific application.

2.3.1 Image Segmentation

Image segmentation is the first step of hand gesture recognition; it aims to isolate the hand gesture from the input image. The segmentation methods are mainly depended on skin colour detection or grey scale (colour or depth) information. It has been very well established that skin segmentation can be better performed on HSV or YC_rC_b colour space [32] [33].

2.3.2 Colour Space Model

This feature has played a larger role in the detection of gestures, often through the clustering of a depth threshold can easily distinguish the approximate area of gesture. In this regard, the main research questions focus on how to determine the depth of the initial gesture depth threshold and clustering. Gaussian model(GM) is a very important continuous probability distribution function in mathematics, physics and engineering and other fields which describes an aggregate value of a single distributed random variables. It uses a Gaussian Probability Density Function (GPDF) (normal distribution curve) to accurately quantify things [34]. Using a number of formalised Gaussian probability density functions based on a mathematical model finally formed. A GM was usually classified to single Gaussian model and Gaussian Mixture Model (GMM) depending on the number of Gaussian probability density function used in the model [35].

Typically using three relatively independent components describes the colour, feature vector consists of three separate components of the composition constitutes a further three-dimensional vector space, which is the colour space.

However, from a different perspective, you can find three different components to describe the colour, so it can produce different colour spaces, but the object itself is being described and as unique.

These describe methods represent different perspectives on the same object. According to the representations of each individual component, colour space can be divided into three categories [10], first colour space, that the three components represent some independent colour perception, according to a variety of different combinations may represent colour perception, such as $RGB(red, green, blue)$, $CMYK$ and $CIEXYZ$, etc. Second, the colour, the light separate space, i.e. represented by a luminance component, two component represents the colour, such as YUV, YC_rC_b, $L*a*b$, $L*u*v$. Third, the intensity, saturation, hue, colour space type, which was used to describe the colour saturation and hue perception, the colour space based on human perception of colour to distinguish, it is more in line with people's intuitive sense, so it was also known as perceptual colour system, such as HSI, HSL, HSV *and* LCH.

In the current gesture recognition system, various colour spaces are used. This section from the three types of colour space, respectively, to select the most commonly used, namely, $RGB, YCrCb, HSV$ colour space, and then compare these three colours in the colour space of cohesion, a non-skin colour separability of other indicators performance under the final decision to use the gesture segmentation of the colour space.

2.3.2.1 RGB space

Visual trichromatic theory was put forward by the Young in 1809 (Fig. 2.6), the main argument is the human retina contains a red, green and blue sensitive cones, different colours reflect different wavelengths, and different wavelength cone cells of the retina influence also vary [36].

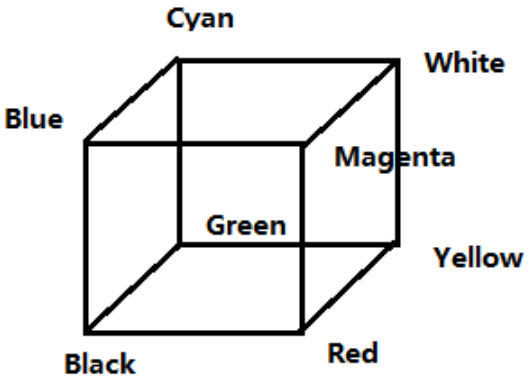

Figure 2.6: RGB colour space

When the cone cells of the different information transfer to the central nervous system, humans have a sense of a certain colour.

Due to technical limitations, early visual trichromatic theory can only stay in the hypothesis stage. Later, with the development of science and technology, the doctrine has been confirmed by many experiments [36].

Currently using a very wide range of RGB colour space is the basis of visual trichromatic theory, the colour space by the superposition of red, green and blue

colours generates almost all colour which human beings can perceive.

2.3.2.2 HSV (Hue, Saturation, Value) Space

HSV colour space has been devised to interpret the way humans describe colour. Hue refers to different colours (such as red, green), S refers to saturation (for instance, the difference between deep blue and light blue), and V refers to value (also considered as brightness). To represent the image accurately with intuitive values, HSV colour space was widely used for image segmentation. Assume the (r, g, b) are the coordinates of red, green and blue respectively. The values are from 0 to 1 [37]. *Max* represents the maximum value among $r, gandb$. *Min* represents the minimum value among the $r, gandb$. The h means the hue angle (between 0-360), S means the saturation, and then these parameters can be calculated as follows.

$$h = \begin{cases} 60 * \frac{g-b}{max-min} + 0 & if \ \ max = r \ \ and \ \ g \geq b \\ 60 * \frac{g-b}{max-min} + 360 & if \ \ max = r \ \ and \ \ g < b \\ 60 * \frac{b-r}{max-min} + 120 & if \ \ max = g \\ 60 * \frac{r-g}{max-min} + 240 & if \ \ max = b \end{cases} \tag{2.1}$$

$$s = \begin{cases} \frac{max-min}{max} = 1 - \frac{min}{max} & otherwise \\ 0 & if \ \ max = 0 \end{cases} \tag{2.2}$$

$$v = max \tag{2.3}$$

Mokhtar M. Hasan et al. proposed a new system to achieve higher segmentation accuracy rate, which is based on HSV colour space method. This system divided

the input image into blocks to extract features. The basic concept is to find the proper block number to achieve the high performance of segmentation rate. They considered all blocks from 1x1 up to 23x23 block size [32].

2.3.2.3 YC_bC_r space

YC_bC_r is also a widely used colour space method during image segmentation (shown in Fig. 2.7). Y represents luminance. C_b and C_r represent blue-difference and red-difference chroma components respectively [38]. RGB values can be transformed to YC_bC_r colour space as follows.

$$\begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65.481 & 126.553 & 24.966 \\ -37.797 & -74.203 & 112 \\ 112 & -93.786 & -18.214 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{2.4}$$

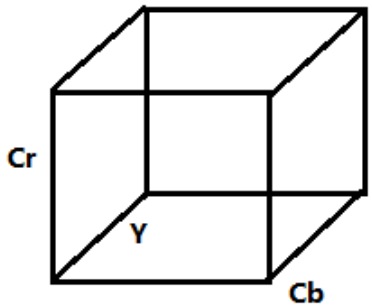

Figure 2.7: YC_bC_r space

Shuying Zhao et al. proposed an improved algorithm of hand gesture recognition for intricate background. This paper illustrates the segmentation method based on YC_bC_r colour space, and also compared four segmentation methods,

HSV, YC_bC_r, N_rg_r *and* N_rg_g. Then authors built a model using Gaussian distribution. This model is highly capable of rejecting the effect of near-skin colour objects [39].

2.3.2.4 Skin Colour Detection

The skin colour detection techniques are commonly based on thresholding techniques. According to different areas where the threshold value is obtained, the segmentation method can be divided into two groups, one is the threshold value only used in one region and the other is the threshold value used in the whole image.

The threshold value is used to classify the points which have similar features on the image into the same class [40]. Assume input image is $f(x, y)$, the output is $f'(x, y)$, the threshold is T, the segmentation process using threshold can be described using the following formula:

$$f'(x,y) = \begin{cases} a_1 & f(x,y) < T \\ a_2 & f(x,y) \geq T \end{cases} \tag{2.5}$$

The threshold-based segmentation process finally produces two value images. The features of the image is usually the colour, grey scale or other (X factor, such as, one particular characteristic) information of the image. Assume the threshold is described as follow:

$$T(C(x,y), G(x,y), X(x,y)) \tag{2.6}$$

When $T = T(G(x, y))$, it means the threshold is only related to the grey scale of the isolated image.

When $T = T(G(x, y), X(x, y))$, which means the threshold will be decided by the grey scale and other characteristics.

When $T = T(C(x, y), G(x, y), X(x, y))$, implies that the threshold will be decided by the position, grey scale and other characteristics.

Because the input hand gestures are flexible and complicated, there is no universal method that can be applied to all gesture segmentation process. Much research has been conducted to find a proper threshold to get a good segmentation results.

Luigi Lamberti et al. proposed using colour glove to recognise hand gestures. During the segmentation process, authors choose the least computationally complex segmentation method, the thresholding - based method. The basic concept is to classify pixels into different (seven) classes by using different thresholds. But the authors did not reveal the details as to how they set up thresholds [41].

E. Stergiopoulou et al. proposed a neural network shape fitting technique to recognise hand gestures. Authors used YC_bC_r colour space to segment the hand region. The thresholding selection is discussed in detail. They further drew up a map of the chrominance of skin colour by using a training set, while the training set includes several images about white hand poorly illuminated, white hand well illuminated and black hand well illuminated. They determined range of C_b and C_r values which is narrow and are very consistent with the existing data. It can minimise the noise and maximum skin colour detection rate [42].

2.3.2.5 Depth Data

After capturing the image, the input image should be segmented, in order to obtain the gesture. The common methods of hand gesture segmentation are colour limitation or skin colour detection. The colour limitation method usually limits the environment by wearing coloured markers or using a fixed colour background. This approach leads to inflexibility of hand movements. However, it has much higher recognition accuracy. Skin colour detection method can directly separate the skin colour area from the input image, but it will be easily affected by the complex background and poor illumination. Besides, hand and face or other similar colour parts cannot be overlapped [43] [44].

Availability of depth information of image objects can overcome these difficulties easily. The grey scale of the pixel in depth image is only related to the distance between surface plane and camera, so the grey scale will not be affected by space colour, illumination and other colour factors. Besides, combining value of grey scale with horizontal and vertical coordinates, in a certain space, can be used to represent the 3D coordinates of an object. In other words, the depth data can be used for gesture recognition in 3D space.

In 1990s, the development of time of flight (TOF) (Fig. 2.8) camera resulted in measuring the depth information of an object by calculating the time of light flying.

Compared to the traditional 2D camera, TOF can easily determine background and foreground. It has unique advantages in target recognition and tracking. But TOF camera also features high price and low resolution.

Figure 2.8: new Time of Flight camera system.
Source: http://lttm.dei.unipd.it/nuovo/research/ToF.html

In 2010, Microsoft developed the somatosensory peripheral 3D camera Kinect for Xbox 360. It uses structured light coding techniques to obtain the depth information from the captured image. Kinect includes one RGB camera, one infra-red camera and one infra-red emitter. The emitter can send infra-red laser, when the laser irradiates the coarse objects, it will form highly random diffraction spots, called laser speckle. The laser speckle will result in different patterns from different imaging distances. When the laser speckle irradiates the entire space, the whole image pixels are tagged. The infra-red camera is used to transmit received tags to the internal graphic processor unit [45]. This module produces the depth image with a resolution up to 640*480 and a low price. Besides, because of the additional graphic processor, it does not need too much computation ability of a computer. This has lead researchers to use Kinect for hand gesture recognition research [46].

K.K. Biswas proposed gesture recognition using Microsoft Kniect [47]. The Kinect produces depth images of subjects (this is grey scale image). This grey scale image is used for isolating the hand region from the input image by using auto thresholding method. According to the depth histogram, the threshold is set to be the first fall down point which is the valley after reaching the first peak.

2.3.3 Feature Extraction

The features are the useful information that leads to classification. Some features have properties of invariance. Rotation, scale or translation. Illumination differences will not lead to misclassification. The features are extracted from the hand posture once it is isolated using image segmentation.

2.3.3.1 Feature Vectors

Good feature vector would be paramount for good classification. For this reason, much research is focused on forming effective features vectors. Features vector of the segmented image can be extracted in different ways according to particular application. Some methods used the shape of hand such as hand contour and silhouette, while others used fingertips position, palm center, etc. A common method extracts the feature vector by dividing the segmented image into fixed block size and each block represents the brightness value in the image [48]. Many experiments were applied to decide the proper block size that can achieve good recognition rate. In practical applications, it cannot be guaranteed that the image will not have translation, scale and rotation. These factors will highly affect the recognition accuracy.

Rajesh et al. proposed a skin colour based method ($L * a * b*$ Colour Space) to isolate the hand gesture from the input image [50]; each hand gesture is centered to extract features. These features used to mark and count peaks and valleys of

each gesture. Meanwhile the system divides input image into sixteen parts to find positions of peaks and valleys. They combine number of Peaks and Valleys with its position of a gesture in the image to recognise the American Sign Language [49]. The recognition method is neural network. This method has one constraint, the hand should be well placed with proper ankle, but it does not require wearing colour gloves or sensor gloves. The test result shows this method can reach 100% recognition rate when applied to American Sign Language.

Hamid A. Jalab et al. proposed an algorithm to recognise static gestures using wavelets [51]. Moreover, it implemented a feed-forward three-layer neural network with back propagation training algorithm during recognition process. Compared to the traditional Fourier methods, the wavelets-based algorithm has advantages in discontinuities of a signal. But this model is only applicable to static gestures. The experimental test result shows the classification accuracy of 97% (one hundred and twenty gestures have been trained and sixty gestures are used for testing).

A. Malima et al.proposed a fast algorithm for robot control [52]. Authors used a fast algorithm, which can recognise a series of gestures applied for a robot control application. This system consists of hand region segmentation stage (skin detection method), locating the fingers stage (count the number of farthest distance between palm centre to finger point) and gestures classification stage. However, there are two prominent constraints in this system; when the system counts the number of fingers, it cannot differentiate different gestures with the same number of fingers, and moreover, this system can only be used for postures.

2.3.3.2 Hu Moment Invariants

The conception of moment invariants was proposed by Hu. Hu proposed 7 famous invariant moments which have been used in many image classification approaches, especially image comparison [53] and matching [54]. Seven invariant moments are extracted from an image which is not affected by translation, scale or rotation. Our research team has effectively used this approach for many classification problems in the past [55]. Equations 7-13 are the well-known Hu moments, which is a set of seven moment invariants are derived from the second order and third order moments[56].

$$\theta_1 = \eta_{20} + \eta_{02} \tag{2.7}$$

$$\theta_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \tag{2.8}$$

$$\theta_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \tag{2.9}$$

$$\theta_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \tag{2.10}$$

$$\theta_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12}) \begin{bmatrix} (\eta_{30} + \eta_{12})^2 - \\ 3(\eta_{21} + \eta_{03})^2 \end{bmatrix} + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03}) \begin{bmatrix} 3(\eta_{30} + \eta_{12})^2 - \\ (\eta_{21} + \eta_{03})^2 \end{bmatrix} \tag{2.11}$$

$$\theta_6 = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \tag{2.12}$$

$$\theta_7 = (3\eta_{21}-\eta_{03})(\eta_{30}+\eta_{12})\left[\ (\eta_{30}+\eta_{12})^2 - 3(\eta_{21}+\eta_{03})^2\ \right]-(\eta_{30}-3\eta_{12})(\eta_{21}+\eta_{03})\left[\begin{array}{c} 3(\eta_{30}+\eta_{12})^2- \\ (\eta_{21}+\eta_{03})^2 \end{array}\right] \tag{2.13}$$

Hu completed the 7 depend invariant moments theoretical framework. J. Flusser think the low order moment invariants includes most useful data [73], but the high order needs more calculation, so it will be affected by the noise. Hence, generally, The I_1, I_2, I_3, I_4 are the most widely used features in hand gesture recognition area.

η_{pq} are the normalised central moments, which can be calculated as:

$$\eta_{pq} \ = \ U_pq \ / \ U'_0 0 \tag{2.14}$$

$$r \ = \ [\ (p+q)/2\ ] + \ 1 \quad and \quad p+q = 2,3,... \tag{2.15}$$

There are further improvements to the basic Hu moments as reported by [57]. The major difference in this approach is to use contour invariant moments.

Lihong Li et al. proposed a system on pattern recognition, which has a high recognition rate by using discrete moment invariant algorithm. The basic idea of this algorithm is to combine the original moment invariants with contour moment invariants. The experimental result shows the recognition rate of the system can achieve over 98% [57].

2.3.4 Gesture Recognition

Recognition of hand gesture is the last stage of the hand gesture recognition system. After modelling and analysis of the input image, the system starts to recognise the gesture. Recognition process is affected by the feature extraction method and classification algorithm. Statistical tools are usually used for gesture classification. Neural network has been widely applied in the field of hand gesture extraction and recognition. In figure 2.9 shows the basic recognition process. Before the recognition stage, the system should be trained with enough data so that a new feature vector can be classified with good accuracy.

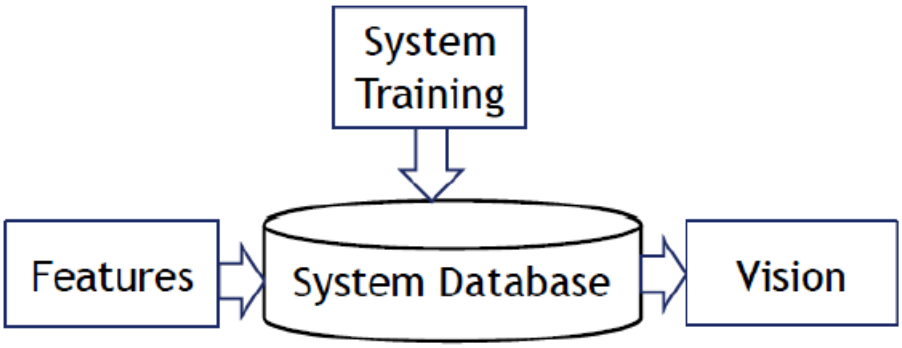

Figure 2.9: Gesture recognition process

2.3.4.1 HMM (Hidden Markov Model)

Up to now, many methods have been applied in hand gesture recognition area. As the dynamic hand gesture can be treated as the continuous motion on time period, HMM can also be used in hand gesture recognition, which is becoming popular.

A HMM is a Markov Model (MM) with hidden parameters, which used the hidden parameters to ensure the state. The variants resulted from states transitions are visible. However, each state cannot be visible; each potential output has a probability distribution. Hence, the sequence of output includes the hidden information of the sequence of state.

In the context of hand gesture recognition area, a HMM can be shown as in figure 2.10. X_1 to X_i represent the input states, which includes a set of hand positions in each state. The T_ab, T_ba represent the state transitions, which means the probability of transferring from one state to another state. The Y_1 to Y_i represent the corresponding outputs, which include one specific posture or one gesture. The database of HMM has many samples per single gesture, the relation between the number of samples and the accuracy is directly proportional and between number of samples and the speed is inversely proportional.

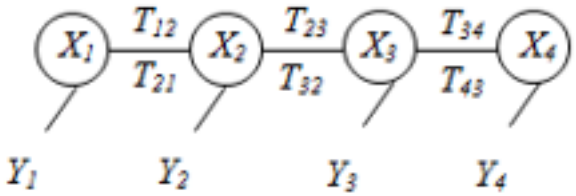

Figure 2.10: the Hidden Morkov Model (HMM) example

Byung-woo Min et al. proposed a system of hand gesture recognition using HMM. They used 8 digit code to represent feature vectors. Each state is characterised by two sets of probabilities, a transition probability and either a discrete output probability distribution or a continuous output probability density functions. They set the number of states for recognition by counting the number of different states included in one gesture. In their work, a 4 - state HMM was used for hand gesture

recognition [58].

2.3.5 Neural Network

Neural Networks have been designed to mimic how neurons behave in human body (fig. 2.11). The artificial Neural Networks as they are known for self-learning and high anti-noise ability. Techniques based on Neural Networks have been widely applied in the field of hand gesture recognition.

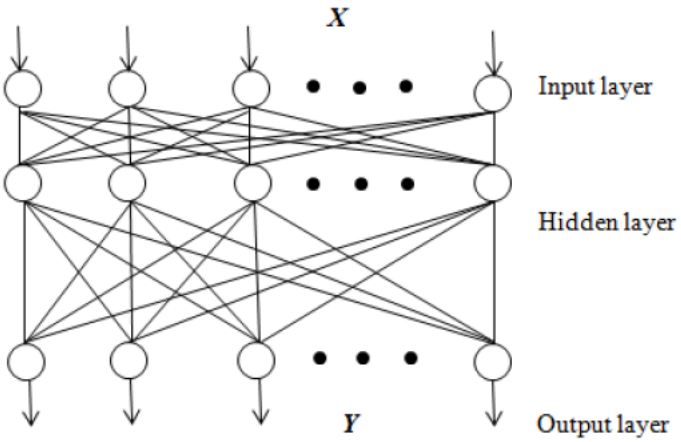

Figure 2.11: the structure of Neural Network (NN)

Nodes are the basic units in the neural network, the weight is the basic parameter between nodes (also called value vector). The above chart shows the simplest neural network. Assume the input are $X_1, X_2, , X_k$, the value vector between input layer and hidden layer is W_ih, the input from input layer to the hidden layer is:

$$Input = \sum_{i}^{h} W_{ih} X_i \tag{2.16}$$

The output from input layer to the hidden layer is:

$$W_i^m = f(\sum_{m-1}^{m} W_{ih} X_k + \theta_i^I), \quad i = i, 2, ... \tag{2.17}$$

Then the output of the hidden layer can be calculated as follows.

$$Output = \sum_{i}^{k} W_{im} X_k \tag{2.18}$$

If the output is not exactly the same with results, the system will turn to operate reversely, it means the error signals will be returned along the original link to repeatedly change the coefficients of all layers until the actual output is corresponding to the input.

E. Stergiopoulou et al. proposed neural network shape fitting technique to recognise hand gestures [42]. Authors use YC_bC_r colour space to segment the hand region. The *SGONG* network is applied to get the hand shape, then three features are obtained; Palm region, Palm center, and Hand slope. It calculates the Center Of Gravity (COG) of the segmented hand and the distance from the COG to the farthest point in the fingers, and extracted one binary signal (1D) to estimate the number of fingers in the hand region. While the characteristics of palm are extracted, combine with the computation, the hand gesture can be identified.
Simei et al. proposed a method of using boundary histograms and neural networks on static gestures [59]. The feature extraction method results in a good recognition performance and leads to the significant reduction of processing time. They use boundary histograms to reduce the effects of rotation and deformable shape of the gestures. In their work, authors illustrated a new fast search start point algorithm for the boundary, which has a rotational invariance property. In the experiments, this model was tested using 26 postures of American Sign Language. The recognition rate directly depends on the number of histograms and

histogram resolution. This method is only used for posture recognition.

2.4 Hand Gesture processing and Tracking

Colour is one of the basic physiological phenomenon of human vision, colour vision is the result of environment, interaction between light and the human brain. Colour perception is part of basic research of the visual system. It has close relation with cognitive science, physiology and information sciences. So colour science is a highly cross-defined disciplines. In the colour study area, biologists hope that through the study of visual information process to understand the process that information are indicated in human brain. On the other hand, in the field of computer vision, the researcher are trying to establish the representation and processing model of colour vision by using the human brain visual process model to finally do the visual information segmentation, tracking and recognition, and ultimately visual understanding. As can be seen, the colour representation and processing of information are fundamental problems among computer vision research. In the field of computer vision, colour information are stored digitally, in practice, most people use colour space to represent colour information and model. In theory, the colour space model is represented by a number of colour components. Some common colour spaces include RGB, HSV, HSL, HSI, YC_iC_b, *Lab*.

Colour is an important and common feature in image target detection. Selecting an appropriate colour space enables skin colour detection to show a better discrimination which can effectively detect exposed parts of the body. Colour model can generally be divided into three categories: The first category is heuristic rules, this approach directly uses a set of determination condition, generally give a direct

result of discrimination.

This mode is typically derived from direct observation of the surface of the problem, it may not follow a mathematical basis. However, it should be noted that although there are heuristics portion shoddy, there are many issues using such methods hit the nail on the head. In a long term and complicated practical, it may exhibit high performance. The second type is a histogram model, it needs to calculate various types of histograms that can achieve a very high image matching accuracy, but its drawback is a huge amount of real-time computation statistics. Meanwhile, the models tend to require a lot of training samples. The third category is the Gaussian Mixture Model, which uses fewer parameters to indicate a centralised distribution regions of different colour spaces. It can be trained offline, when it is on line, the model is able to determine results directly by its fast calculation method.

Depth refers to the distance from the object to image detecting devices. In recent years, with the popularity of various integrated imaging device, obtaining depth data of an image is easier, and thus the depth data is also widely used to a variety of object detection tasks.

2.4.1 Mean shift algorithm

R^d represents an arbitrary d -dimensional space. Assume several samples exist in the space represented by $x_i = 1, ..., n$, so the mean shift of any x in it can be defined as follows.

$$M_h(x) = \frac{1}{k} \sum_{x_i \in S_h} (x_i - x) \tag{2.19}$$

Wherein, s is Mean Shift constraint, which defined as the d-dimensional space in a high-dimensional sphere of radius region h. It's a collection of the points satisfied restrictions.

$$S_h = \{y : (y - x)^T (y - x) \leqslant h^2\} \tag{2.20}$$

k represents the quantity of samples in the High-dimensional sphere. The Mean Shift of sample point is $(x_i - x)$. The arithmetic average Mean shift of all sample points and the point x the final high-dimensional ball area is represented by $M_h(x)$. From the probability density function (PDF) modelling point of speaking, each sample point x can be derived from the PDF $f(x)$ sampling. According to the definition of gradient extended to PDF [60], the directions of PDF and PDF gradient fastest increase the value are the same direction, so calculating $M_h(x)$ can guarantee the directions of this value and PDF gradient are the same.

Mean shift algorithm in target tracking applications mostly applied by the following four steps.

1. Built target model. It is a process to build the tracking model for the target. Recently it mostly uses grey scale and colour histogram as modelling tools to describe the target. Usually, the center of target is setted to X_0, the tracked target can be defined as follows.

$$q = \{q_u\}, u \in [1, ..., m] q(u) = C \sum_{i=1}^{n} k(\| \frac{X_i - X_0}{H} \|^2) \delta(b(X_i) - u), i \in [1, ...n] \tag{2.21}$$

X_i is the ith point of the window, C is a normalisation constant to make sure

$\sum_u \in [1, ...m]q_u = 1$. $k(x)$ is a kernel function, H is the bandwidth vector. M is the quantity of eigenvalues, that is the grades of grey scale levels. Eigenvalue u is the corresponding grey scale level. δ function is a pulse function to ensure the pixel which has eigenvalue u making sense for probability of distributions. Thus k function can be regarded as a weighted frequency of grey scale u.

2. Establish a matching object model same with object model.

$$p_u(Y) = C_h \sum_{i=1}^{n_h} k(\| \frac{X_i - Y^2}{H_h})\delta(b(X_i) - u), \quad i \in [1, ..., n_h] \tag{2.22}$$

Wherein, Y is the center of a matching object, X_i is the ith vector within the matching window, H_h vector is the bandwidth vector of kernel functions, C is the normalisation constant of in matching window.

3. Define a similarity metric function. To define a similarity metric function usually uses Bhattacharyya function to measure the similarity degree between candidates and target model. This function is defined as follow.

$$\rho(p(Y), q) = \sum_{u=1}^{m} \sqrt{p_u(Y)q_u} \tag{2.23}$$

4. The matching process is to find the maximum value of a similar function. Mean shift uses a gradient descent method. First, doing the Taylor series expansion to $p(Y)$ near $p(Y_0)$. Get the first two as follows.

$$p \approx p(Y_0) + \frac{d_\rho}{d_p}(p(Y) - p(Y_0)) \tag{2.24}$$

Let $\rho_u(Y) = \sqrt{p_u(Y)q_u}$, so

$$\rho_u(Y) = \rho_u(Y_0) + \frac{q_u}{2\sqrt{p_u(Y_0)q_u}}(p_u(Y) - p_u(Y_0)) = \frac{1}{2}(\rho_u(Y_0) + \sqrt{\frac{p_u(Y)q_u}{p_u(Y_0)}}) \tag{2.25}$$

It can be achieved as follows.

$$\rho(Y) = \frac{1}{2}\sum_{u=1}^{m}\sqrt{p_u(Y_0)q_u} + \frac{1}{2}\sum_{u=1}^{m}p_u(Y)\sqrt{\frac{q_u}{p_u(Y_0)}} \tag{2.26}$$

To make $\rho(Y)$ go to the maxim value, Y must search alone the gradient direction. Gradient direction Y can be obtained by derivation.

$$\nabla\rho(Y_0) = \frac{C_h}{H_h^2}[\sum_{i=1}^{n_h}\omega_i g(\|\frac{Y_0 - X_i}{H_h}\|^2)][\frac{\sum_{i=1}^{n_h}X_i\omega_i g(\|\frac{Y_0-X_i}{Hh}\|^2)}{\sum_{i=1}^{n_h}\omega_i g(\|\frac{Y_0-X_i}{H_h}\|^2)} - Y_0] \tag{2.27}$$

Among it,

$$\omega_i = \sum_{u=1}^{m}\sqrt{\frac{q_u}{p_u(Y_0)}}\delta(b(X_i) - u) \tag{2.28}$$

is the weight.

If

$$Y_1 = \frac{\sum_{i=1}^{n_h}X_i\omega_i g(\|\frac{Y_0-X_i}{H_h}\|^2)}{\sum_{i=1}^{n_h}\omega_i g(\|\frac{Y_0-X_i}{H_h}\|^2)} \tag{2.29}$$

so $Y_1 - Y_0$ can be the same gradient direction.

2.4.2 Bayes Filter

Motion sequence tracking problem can be constructed as a state space model described in following figure. Based on this modelling approach, in order to simplify the process, usually considering tracking problems meet two assumptions. First is that the current state of the system is only related to the latest state and has no relation to any other states, meanwhile, the transfer between states obeys the first order Markov process. Second is that the observations at one single time is only related to the state of system and with no relation to the state of any other time [61].

In object tracking problem, the space model of a dynamic system can be described as follows.

$$x_k = f(x_{k-1}) + u_{k-1} \qquad y_k = h(x_k) + u_k \tag{2.30}$$

x_k, y_k, u_k, v_k represent system state, observation value, process noise and observation noise of the model respectively at k state. $f(x), h(x)$ respectively define the state transfer function and observation function. Usually, $X_k = x_{0:k} = \{x_0, x_1, ..., x_k\}$ and $Y_k = y_{0:k} = \{y_0, y_1, ..., y_k\}$ are used to represent all system states and observation values from 0 to k state.

Bayes filter solves non-linear systems state estimation problems by using prob-

ability distribution modelling [62]. Specifically, Bayes filter estimates the probability of system status using derivation procedure, that is using Bayes probability formulas calculate posterior probability $p(X_k|Y_k)$ or current probability density $p(x_k|Y_k)$ of the system to obtain the estimation of the target state. Bayes filter algorithm generally has two phrases, prediction and update. Prediction stage is using the priori probability of a known time to achieve priori probability of the prediction state of the next time. Updating phrase is based on the prediction process to obtain the priori probability of prediction state of the next time, and then use Bayes filter formula to obtain the posterior probability density.

Let $p(x_{k-1}|Y_{k-1})$ be the probability density function of $k-1$ state, hence the Bayes filter can be describe as follows.

1. Prediction phase, obtain $p(x_k|Y_{k-1})$ from $p(x_(k-1)|Y_(k-1))$:

$$p(x_k, x_{k-1}|Y_{k-1}) = p(x_k|x_{k-1}, Y_{k-1})p(x_{k-1}|Y_{k-1}) \tag{2.31}$$

So when $x_(k-1)$ is confirmed, state x_k and Y_{k-1} are independent. Hence following formulas can be obtained.

$$\rho(x_k, x_{k-1}|Y_{k-1}) = p(x_k|x_{k-1})p(x_{k-1}|Y_{k-1}) \tag{2.32}$$

Both ends of the above formula calculate integral to x_{k-1}, the Chapman Komolgorov formula can be achieved as follows.

$$p(x_k|Y_{k-1}) = \int p(X_k|x_{k-1})p(x_{k-1}|Y_(k-1))dx_{k-1} \tag{2.33}$$

2. Update phrase, obtain $p(x_k|Y_k)$ from $p(x_k|Y_{k-1})$ as follows. Posterior probability density can use Bayes function, in certain prediction phrase when $p(x_k|Y_{k-1})$ has been achieved, $p(x_k)|Y_k$ can be got as follows.

$$p(x_k|Y_k) = \frac{p(y_k|x_k, Y_{k-1})p(x_k|Y_{k-1})}{p(y_k|Y_{k-1})} \tag{2.34}$$

Based on the second assumption of the space model, Y_k is only related to x_k, hence,

$$p(y_k|x_k, Y_{k-1}) = p(y_k|x_k) \tag{2.35}$$

so,

$$p(x_k|Y_k) = \frac{p(y_k|x_k)p(x_k|Y_{k-1})}{p(y_k|Y_{k-1})} \tag{2.36}$$

Among it, $p(y_k|Y_{k-1})$ is normalisation constant.

$$p(y_k|Y_{k-1}) = \int p(y_k|x_k)p(x_k|Y_{k-1})dx_k \tag{2.37}$$

After the optimal solution of posterior probability density function was achieved by Bayes filter, generally, the achieved optimal solution needs to do maximisation based on Maximum A Posteriori (MAP) criterion or Minimum Mean Square Error (MMSE) criterion. finally, use this value as the estimation of the system state. It can be shown below.

$$\hat{x}_k^{MAP} = arg_{x_k}^{min} p(x_k|Y_k) \tag{2.38}$$

$$\hat{x}_k^{MMSE} = E[f(x_k)|Y_k] = \int f(x_k)p(x_k|Y_k)dx_k \tag{2.39}$$

According to the above calculation can be seen Bayes filter requires integral calculation. It is almost impossible for a general system which has no special properties (Gaussian or finite state discrete system) to get the posterior probability density resolution. Under more general cases, the approximate solutions of integration problem can only be achieved by numerical calculation. Among many methods to obtain approximate solutions, the method which based on particle filter algorithm of Monte Carlo simulation is widely used.

2.4.3 Particle Filter

The general process of particle filter algorithm is to initial the system at time $k = 0$ state. The prior probability of the target can be obtained at this time, and then setting the initial value for each particle bases on the prior probability. When the system moves to the next time state, $k = k + 1$, with the transition of system state, all particles ensure the current state to be based on the state transition equation. Meanwhile, getting the observation value from the systematic observation, and making a similarity comparison of observation value and particle state to obtain the weight of each particle. The posterior probability was obtained from particle

weight. Finally, according to the weight of particles at this time, and then after re-sampling, system will make state transitions into the next cycle step [63] [64].

1. Importance sampling. Particle filter randomly take N independent distribution samples $x_k^{(i)}, i = 1, 2, ..., N$ from the posterior probability $p(x_k|Y_k)$, and then do the weighted sum for these samples to approximate the integral operation.

$$p(x_k|Y_k) \approx \frac{1}{N} \sum_{i=1}^{N} \delta(x_k - x_k^{(i)}) \tag{2.40}$$

Among this, x_k is continuous variable, $\delta(x - x_k)$ is a unit impulse function (Dirac function), it means when $\delta(x - x_k) = 0, x \neq x_k, \int \delta(x)dx = 1$. and x_k are discrete variables, the posterior probability distribution can be approximated to follows.

$$p(x_k|Y_k) \approx \frac{1}{N} \sum_{i=1}^{N} \delta(x_k - x_k^{(i)}) \tag{2.41}$$

Among it, $\delta(x - x_k^{(i)}) = 1, \ x = x_k^{(i)}, \ \delta(x - x_k^{(i)}) = 0, \ x \neq x_k^{(i)}$.

Assuming $x_k^{(i)}$ is the sample particle from the posterior probability function $p(x_k|Y_k)$, so the expected estimation of random function $f(x_k)$ can be approximated by summation mode as follows.

$$E[f(x_k)|Y_k] = \int f(x_k)p(x_k|Y_k)dX_k = \frac{1}{N} \sum_{i=1}^{N} f(x_k^{(i)}) \tag{2.42}$$

Since it is difficult to take samples directly from the posterior probability

distribution in the actual model [65], importance sampling method uses probability density function $q(x_k|Y_k)$ which is a relatively easy sampling method, and then getting samples based on this function to approximate the posterior probability density $p(x_k|Y_k)$ using the weight of these random samples. If $\{x_k^{(i)}, \omega_k^{(i)}, i = 1, ..., N\}$ represents the collection of random samples, the i -particle state in time k is $x_k^{(i)}$, the corresponding weighted value is $\omega_k^{(i)}$, therefore, the posterior probability density can be represented as follows.

$$p(x_k|Y_k) = \sum_{i=1}^{N} \omega_k^i \delta(x_k - x_k^{(i)}) \tag{2.43}$$

Among it, $\omega_k^{(i)} = \frac{p(x_k^{(i)}|Y_k)}{q(x^{(i)}|Y_k)}$. The larger number of random samples, the more accurate approximation probability density function can be achieved, so the expected estimate of random function can be described as follows.

$$E[f(x_k)|Y_k] = \frac{1}{N}\sum_{i=1}^{N} f(x_k^{(i)})\frac{p(x_k^{(i)}|Y_k)}{q(x_k^{(i)})|Y_k} = \frac{1}{N}\sum_{i=1}^{N} f(x_k^{(i)})\omega_k^{(i)} \tag{2.44}$$

2. The importance probability density function. The particle filter algorithm is affected by the chosen of importance probability density function [66]. In actual calculation, the state transition probability density function in the system $p(x_k|x_k - 1)$ is usually used as importance probability density function. So the weight of particle can be shown as follows.

$$\omega_k^{(i)} = \omega_{k-1}^{(i)} p(y_k|x_k^{(i)}) \tag{2.45}$$

There are many methods to improve the efficiency of particle sampling, such as, approximating $p(x_k|x_k - 1, Y_k)$ uses local linear method, uses gradi-

ent data, mean shift or Newton iteration. The advantages of these methods are making particles closer to the posterior probability density distribution of the system, also, they can decrease the number of particles that the system needs.

3. Particle filter algorithm steps. First, set a default weight, the priori probability of $k-1$ time can be represented by N -number of $x_k^{(i)}$ particles which weight is $\frac{1}{N}$. In time update phrase, the state of each particles $x_k^{(i)}$ at k time can be predicted by system state transition function. The third observation phrase, adjust the weight value $\omega_k^{(i)}$ of each particle based on the posterior probability density function. In the last sampling phrase, the big weight particle replaces the small weight particle, and the weight of particle is setted to $1/N$.

   The standard particle filter algorithm steps like follows.

   1. Initial the particle collection, set $k = 0$. In the time update phrase, for $i = 1, 2, ..., N$, the random sampling particle $\{x_0^{(i)}\}_{i=1}^N$.

   2. In observation phrase, for $i = 1, 2, ..., N$, the sampling particle $\{x_0^{(i)}\}_i^N = 1$ is produced from importance priori probability density function $p(x_0)$, and then calculate the particle weight $\omega_k^{(i)}$. In re-sampling phrase, doing the re-sampling to particles collection $\{x_k^{(i)}, \omega^(i)_k\}$, the collection of re-sampling particles is $\{x_k^{(i)}, 1/N\}$. Then get the result, the estimation of system state at k time, $x_k = \sum_{i=1}^N x_k^(i)\omega_k^(i)$.

2.4.4 Depth Based Hand Tracking

The $[0, L-1]$ range histogram can be represented as discrete function $h(r_k)$, r_k is k grade grey scale, N_k is the number of pixels that have r_k grey scale, $k = 0, 1, ..., L-1$, usually, Using the total number of pixels n divide each value can get normalised histogram. Hence, the normalised histogram was given by $P(r_k) = n_k/n$. $P(r_k)$ gives the probability estimation value of r_k grey scale. As we can see from the above definition, the summation of all parts of the histogram is equal to 1.

Relative depth histogram calculates the absolute depth value of each pixel compared to the minimum depth value in the image based on the absolute depth value of each pixel in the image, so the value range of relative depth in the image can be defined as $[0, D-1]$. Similar to the grey scale histogram, if use discrete function $h(r_k) = n_k$ to represent the relative depth histogram, r_k is the relative depth of k grade, n_k is the number of pixels in r_k grey scale level. $k = 0, 1, ..., D-1$, so the normalised histograms of each value can be achieved from the total number of pixels in the image n divides the each value.

The similarity of relative depth histogram was determined by the Bhattacharyya Distance [67]. The Bhattacharyya Distance was widely used to measure the similarity of two probability distribution. Under the discrete probability distribution circumstance, a domain X is designed as follows [68].

$$DB(p,q) = -In(BC(p,q)) \qquad BC(p,q) = \sum_{x \in X} \sqrt{p(x)q(x)} \tag{2.46}$$

Among it, $BC(p,q)$ is called Bhattacharyya Coefficient.

2.5 Hand Gesture Description and Recognition

Hand gesture recognition system needs a hand gesture database that the system can recognise. However, the gestures in library have to have close connection with the application functions. Very few people can remember the complex and unnatural hand gestures. Hand gesture database has to consider the different culture background and different age range of users. So the database built in this thesis is based on the ASL (Australian Sign Language) to do the research on the general gesture recognition [69].

Static hand gesture is consisted by the direction of hand palm and fingers included the toward direction excluded the motion information. Static hand gesture recognition involves features definition, extraction and classification phrase. The difficulty of features definition and extraction is that the recognition area has a relatively consistent colour, it's very hard to extract useful information from the skin texture, hence the features are mainly focused on edges, contours and hand structures in static hand gesture recognitions [70]. Support Vector Machine (SVM) attracts the most research in static hand gesture recognition field. The hot spot is how to make SVM classify many categories.

Dynamic hand gesture is a image sequence of hand movement within a period. The hand gesture trajectory is a main feature to classify gestures among dynamic gesture recognition. The trajectory is consisted by the center point of each frame in the sequence [71]. Most research project the hand gesture trajectory into two dimensional space to extract the feature. It makes the calculation simple, but meanwhile, it lost the depth information. Putting depth information into a 3 dimensional space is a trend to react the real hand gesture information.

2.5.1 Static gesture description

In this thesis, Hu moment invariants are used as hand gesture features. Hu moment invariants consist of 7 non-linear moment invariants. In feature extraction process, Hu moment invariants will not be affected by translation, rotation and scale [72]. It can be shown as follows (Fig. 2.12).

Letter 'A'	(a)	(b)	(c)
$\emptyset_1$	0.2165	0.2165	0.204
$\emptyset_2$	0.001936	0.001936	0.002161
$\emptyset_3$	3.6864e-005	3.6864e-005	3.6864e-005
$\emptyset_4$	1.6384e-005	1.6384e-005	1.6384e-005
$\emptyset_5$	-4.0265e-010	-4.0265e-010	-4.0265e-010
$\emptyset_6$	7.209e-007	7.209e-007	7.209e-007
$\emptyset_7$	0	0	0

(a) (b) (c)

Figure 2.12: Hu moment invariants theory, as can be seen, no matter what kinds of translation, rotation or scale happen to the letter A, the moment invariants of it keeps stable.

Fourier Descriptor (FD) is usually applied to extract the contour feature of gesture [74]. The basic algorithm step is to build the hand contour curve using one dimensional points sequence, and then doing the Fourier transformation to these points, finally, transfer the achieved Fourier to the hand contour features [75].

FD has a series of advantages, such as, simple calculation principle, it does't need

to set a large number of parameters, Just several Fourier coefficients can describe the features of system sequence. The more coefficients the system has, the more accurate features the system gets.

The collection of points $\{(x_k, y_k)\}$ in surface plane X, Y built a close contour. If regard X, Y as a complex plane, the sequence of points can be regarded as a one dimensional complex sequence, among it, $c_k = x_k + j \cdot y_k$, do Fourier transform as follows.

$$C(u) = \frac{1}{N} \sum_{k=0}^{N-1} c(k) \cdot e^{-j2\pi uk/N}, \qquad u = 0, 1, 2, ..., N-1 \tag{2.47}$$

Wherein complex coefficients $C(u)$ is called the Fourier descriptor of hand contour. The first Fourier descriptor is the mean of all points on the contour curve in X, Y plane. It's the centroid of contour to offer the position information. The second Fourier descriptor gives the radius of circle that can cover as much as contour points. Hence, it is possible to rebuild a circle using the first two descriptors.

Correspondingly, Inverse transform of Fourier descriptors $C(u)$ can rebuild the former contour curve, it is shown below.

$$c(k) = \sum_{u=0}^{N-1} C(u) \cdot e^{j2\pi uk/N}, \qquad k = 0, 1, 2, ..., N-1 \tag{2.48}$$

The sequence of Fourier descriptor $C(u)$ responds the shape feature of original curve. Meanwhile, Fourier transform is energy concentration, so a few number of Fourier descriptors is able to rebuild the original curve. The high frequency coefficients of Fourier descriptors are mainly used to describe the contour detail, low frequency coefficients are mainly used to describe the whole shape of the contour [76].

The Fourier descriptors are related to the shape, direction and start, finish position of curve. The number of fingers is the most widely used hand gesture feature, generally, the number of fingers are obtained using penetration method.

1. Using the left side as the start point, the entire image is scanned every other column in intervals of 10 pixels.
2. The total number of points $k_i (1 \leq k_i \leq 10)$ recorded in every single column on contour curve means k_i is the number of points in i -column on the contour curve.
3. Without loss of generality, set the maxim of k_i is k_max, obviously, the number of fingers can be defined as $k_max/2$.

The ratio of perimeter and area of a hand is another widely used hand feature. Assume $f(x, y)$ is the points on hand gesture contour, the perimeter means the summation number of pixels on the hand contour. It can be shown as follows.

$$perimeter = \sum_{x} \sum_{y} f(x, y) \tag{2.49}$$

The area of hand contour is the summation of all pixels on the hand contour, it can be shown as follows.

$$area = \sum f(x, y) \tag{2.50}$$

Hence, the ratio of perimeter and area of and contour is like follows.

$$pa = \frac{perimeter}{area} \tag{2.51}$$

2.5.2 Dynamic gesture description

Dynamic gesture trajectory is a space model consisted by centroid points. Generally speaking, there are two location features, the first is set to $L1$ to represent the distance from any point to the centroid point [77]. It can be defined as follows.

$$L_1 = \sqrt{(x_{t+1} - C_x)^2 + (y_{t+1} - C_y)^2} \tag{2.52}$$

$$(C_x,\ C_y) = \frac{1}{n}(\ \sum_{t=1}^{n} x_t,\ \sum_{t=1}^{n} y_t\ ) \tag{2.53}$$

the second location feature is set to L2 to represent the distance from the start point to the current point in the dynamic gesture trajectory. It can be defined as follows.

$$L_2 = \sqrt{(x_{t+1} - x_1)^2 + (y_{t+1} - y_1)^2} \tag{2.54}$$

Among it, $t = 1, 2, ..., T-1, T$ represents the length of dynamic gesture trajectory.

Dynamic gesture recognition has three basic elements, location, orientation and velocity. Orientation features represent the hand direction at any time. Features are based on displacement vector in the trajectory, which usually includes three orientation features [78].

The first feature is the centre direction of dynamic gesture trajectory. It is defined below.

$$\theta_{1t} = tan^{-1}(\frac{y_{t+1} - C_y}{x_{t+1} - C_x}) \tag{2.55}$$

The second feature is the direction of two continuous points in the trajectory. It is defined as below.

$$\theta_{2t} = tan^{-1}(\frac{y_{t+1} - y_t}{x_{t+1} - x_t}) \tag{2.56}$$

The third feature is the direction from the start point to current point. It is defined as below.

$$\theta_{3t} = tan^{-1}(\frac{y_{t+1} - y_1}{x_{t+1} - x_1}) \tag{2.57}$$

The orientation feature is in a 3-dimensional dynamic hand gesture trajectory that also uses time as the fourth dimensional, so the orientation feature is defined as follows.

$$\theta_{1t} = tan^{-1}(\frac{z_{t+1} - C_z}{\sqrt{(y_{t+1} - C_y)^2 + (x_{t+1} - C_x)^2}}) \tag{2.58}$$

$$\theta_{2t} = tan^{-1}(\frac{z_{t+1} - z_t}{\sqrt{(y_{t+1} - y_t)^2 + (x_{t+1} - x_t)^2}}) \tag{2.59}$$

$$\theta_{3t} = tan^{-1}(\frac{z_{t+1} - z_1}{\sqrt{(y_{t+1} - y_1)^2 + (x_{t+1} - x_1)^2}}) \tag{2.60}$$

Generally, the hand moving speed at corner of dynamic gesture will decrease, the hand moving speed in a stable line will increase, sometimes, the speed will also change unpredictably. hence, the velocity feature is defined by the ratio of

Euclidean distance and time of corresponding points in consecutive frames, the time is usually instead by number of frames. It is defined as follows.

$$V_t = \sqrt{\left(\frac{x_{t+1} - x_t}{t}\right)^2 + \left(\frac{y_{t+1-y_t}}{t}\right)^2} \tag{2.61}$$

The velocity feature is in a 3-dimensional dynamic hand gesture trajectory that also uses time as the fourth dimensional, so the velocity feature can also be defined as follows.

$$V_t = \sqrt{\left(\frac{x_{t+1} - x_t}{t}\right)^2 + \left(\frac{y_{t+1-y_t}}{t}\right)^2 + \left(\frac{z_{t+1} - z_t}{t}\right)^2} \tag{2.62}$$

Dynamic hand gesture trajectory is usually described by using the different combinations of above three features.

2.5.3 Support Vector Machines

In 1995, Cortes and Vapnik first proposed the concept of Support Vector Machine (SVM) [79]. Then Boser, Guyon and Vapnik introduced kernel function to solve the non-linear SVM problemm. SVM has many unique advantages on solving problems that have few samples, non-linear low-dimensional space and high-dimensional space recognition. Meanwhile, it can also be apply to function fitting and other areas. SVM algorithm needs relatively less samples than other algorithms. SVM uses slack variants and kernel function to solve sample data linearly inseparable problem. SVM is a two class classifier, the basic model is to assign the maximum interval to linear classifier in feature space, and also use kernel function to be a non-linear classifier[80].

The basic model of SVM has three categories, linearly separable SVM, linear SVM and non-linear SVM. If the training data is linearly separable, it can produce a linear classifier by learn the maxim interval, the system is linearly separable SVM. If the training data is not totally linearly separable, just a part of data is linearly separable, it can also produce a linear classifier by learn the maxim interval, the classifier is linear SVM. If the training data is totally not linearly separable, the kernel function and maxim interval are introduced, the produced classifier is non-linear SVM.

The collection of training data in feature space can be represented as follows.

$$T = (x_1, y_1), \ (x_2, y_2) \ , ..., \ (x_N, y_N) \tag{2.63}$$

Among it, $x_i \in R^n$, $y_i \in +1, -1, \ \ i = 1, 2, ..., N$, x_i is the i -feature vectors in the training set, y_i is the class mark of x_i. It means if $y_i = +1$, feature vector x_i is positive case. If $y_i = -1$, feature vector x_i is negative case. (x_i, y_i) is called sample point.

The object of SVM is to train the data set, which means looking for a separating hyperplane to classify the features to two different categorises in the feature space, one side is negative cases, another side is positive cases. in R^n space, the separating hyperplane can be described as $\omega \cdot x + b = 0$, wherein ω represents the normals of this plane, b is intercept.

Generally speaking, the distance between the separating hyperplane and the point can be used to describe the accuracy of prediction of classification. In a certain $\omega \cdot x + b = 0$ separating hyperplane, the distance between x_i and separating hy-

perplane can be shown as follows.

$$d = \frac{|\omega \cdot x_i + b|}{||\omega||} \tag{2.64}$$

Wherein $||\omega||$ is the L_2 norm of ω. But in SVM, the distance from feature vectors x_i to the hyperplane is represented by $|\omega \cdot x_i + b|$.

If in a certain training set T and hyperplane (ω, b), the function interval of samples (x_i, y_i) in hyperplane (ω, b) can be defined as $\gamma_i = y_i(\omega \cdot x_i + b)$. Furthermore, the minimum function interval value of sample points (x_i, y_i) is equal to the function interval of training set T in hyperplane (ω, b). It is $\gamma = min_{i=1,\ldots,N}\gamma_i$.

As we can see, whether the symbol of $\omega \cdot x_i + b$ and the symbol of class mark y_i are the same, it means the classification of y_i is correct or not. So the function interval has the able to describe the classification correct. According to the above formulas, function interval $|\omega \cdot x_i + b|$ is changing with ω, b, usually, it will need to be normalised to achieve the certain interval, and it makes the function interval change to geometric interval, it can be defined as follows [81].

$$\gamma_i = \frac{\omega}{||\omega||} \cdot x_i + \frac{b}{||\omega||} \tag{2.65}$$

Similarly, the geometric interval of training set T in hyperplane (ω, b) is the minimum function interval of all sample points (x_i, y_i), that is $\gamma = min_{i=1,\ldots,N}\gamma_i$.

At this time, if the parameters ω, b of hyperplane (ω, b) proportionally change, the function interval will change proportionally correspondingly, but the geometric interval will not change, the relation between function interval and geometric interval can show as follows.

$$\gamma_i = \frac{\gamma_i^1}{||\omega||}, \quad \gamma = \frac{\gamma^1}{||\omega||} \tag{2.66}$$

If the reality data set is linearly separable, it only needs to find the biggest separating hyperplane of the collection interval of the training set. If the data set is not linearly separable, it needs kernel function and slack variables.

2.6 Summary

In summary, this thesis differs from past works in the following manners:

1. Past works on the hand gesture recognition problem ensure all gestures are monitored by sophisticated algorithms. The key limitation is that the response time of the system is restricted by redundant codes, and recognition accuracy of recognition model was not able to reach the expectation.

   On the other hand, the conventional algorithms mainly focus on detection methodology and recognition logic. In authors opinion, the combination of depth and colour information is a better way. To fulfil these gaps, this thesis considers upgrade the algorithm from logical to code with an objective to maximise the recognition rate and minimise the response speed using fusion method of depth information and colour information.

2. Past thresholding strategies assume the input gesture has a fixed colour background or a colour marker. However, this is not valid in practice because the user have different input environment, and users also may

have different skin colours which have high probability to interface the recognition rate.

This thesis is the first to propose a simple but efficient algorithm for complete background isolation using the depth information onto the colour information.

3. Similar to the response speed problem, past works on the hand gesture recognition has no chance to build a big dataset, because the bigger dataset the system has, the slow response speed the system get. To fulfil this gap, this thesis proposes a novel high recognition rate hand gesture recognition model.

Chapter 3

Robust Hand Gesture Detection by Fusion of Depth and Colour Information using Kinect

This chapter considers the gesture segmentation problem in the context of hand gesture recognition process. The goal is to maximise the recognition rate whilst ensuring all other environment factors are still the same, such as light, background colour and computer. As mentioned in Chapter 2, past works to solve the segmentation problem do not consider using the combination of depth data and colour information with new algorithms. Moreover, existing works on coverage for hand gesture recognition have only focused on maximising the recognition rate by using more restrictions or more advanced devices. In comparison to the past works, this chapter aims to ensure complete logic upgraded fusion model

and afford new efficient algorithm.

3.1 Depth Cameras Overview

As a major method in human communication such as in sign language, hand gesture recognition has been applied to the Human- Computer Interaction (HCI) area for a long time. Compared to the traditional inputs such as keyboard and mice, hand gestures are more natural and flexible. They have a great potential in the area of digital control, real-time input and communication among disabled [8]. In recent years, research has focused on vision-based hand gesture recognition and control as the contact-type devices strongly diminish the flexibility of hand movements. As shown in Fig. 3.1

Figure 3.1: contact type device and vision based device

The tremendous development of computational ability of many hardware devices, such as smart phones, has allowed computer vision the opportunity to play an important role in human computer interaction [82]. Although, the research on vision-based hand gesture analyses have made rapid progress, the reliability and practicality of the hand gesture recognition systems are still problematic. The biggest challenge in advancing the field lies on reliably tracking hand gestures

in order to recognise dynamic gestures under complex lighting conditions and background clutter [83][84].

The recognition systems under background clutter usually require the background to be free of skin tones. This usually requires the people to wear colour markers or use fixed colour background, then machines analyse and segment hand gestures by detecting the marked colour [30]. These contact type devices such as colour gloves or markers worn on hand limits the flexibility and reaction time of the system. However, these methods have advantages in terms of accuracy. The skin colour detection is based on the characteristics of the spatial distribution of the skin colour in colour space to convert the image to the corresponding colour space to do the threshold segmentation [85]. The skin colour detection can directly isolate the skin colour area from the image. But there are some disadvantages in current technology, such as the gesture and skin colour area cannot overlap, if not, the segmentation will be affected by clutter background [86] [4].

Nowadays, with the development of cameras, there are two kinds of the most adopted devices for hand gesture detection and recognition, one is Time of Flight (TOF) cameras [87] shown in Fig. 2.1, and another one is Microsoft Kinect (shown in Fig. 3.2).

Both devices can produce the depth image which is also known as the range image. The information of this kind of image records the distance of each point of the space between the object surface and the camera, the grey scale of each pixel on depth image is only related to the distance of each point between the object and camera, so depth data have the 3D characterises of an object in the space, which the grey scale image and colour image does not have. It can be used to accurately extract the foreground from background in computer vision.

Figure 3.2: Kinect and Kinect 2
Source: http://www.xbox.com/

The TOF cameras record the depth data by measuring the flying time of the light between the object (the hand) and the sensor. In fact, it calculates the time elapsed between the sent pulse and the reflection of it off the object when received by the receiving sensor [88]. Compared to the traditional 2D camera, the TOF cameras can easily extract foreground from background, so it has advantages of object tracking and analysis. However, the disadvantage is that the TOF cameras are expensive and lower resolution [89].

In 2010, Microsoft launched a 3D camera peripheral somatosensory for the Xbox360. The Kinect uses structured light coding techniques to obtain depth information of captured images. The Kinect camera includes an RGB camera, an infrared camera and an infrared emitter. The Infrared emitters can emit near-infrared laser, when the laser irradiate rough object, it will produce a high degree of randomness diffraction spots, called laser speckle. Laser speckle will vary patterns according to the distance of objects. When the laser speckle irradiate to the entire space, it means that the space has been marked. Infrared camera is used to receive

space markers and pass the makers to the core chip of the Kinect. The processor produces the depth image by analysing the laser speckle pattern. Two Kinect cameras are shown in Fig3.2

Compared to the TOF cameras, Kinect is much cheaper with higher resolution. Besides, the Kinect has an additional graphic processor, so there is no extra computation for the computer. It achieves the real- time gesture analysis under a comparative low configuration [90].

These features make Kinect become a popular tool in the domain of movement recognition. The hand gesture recognitions based on it usually use the depth information, which was produced by Kinect. There are two common ways for using the depth information [91]. First is that using depth information instead of colour information, which means transferring the depth information of the hand area to 2D image [92], and then applies the traditional recognition methods to the 2D image, this kind of methods actually take advantage of the depth information to get a relative robust recognition. However, it wastes of depth data while converting 3D depth information to 2D information, and it will easily effected by the finger occlusion problems [93].

The second way is totally using depth information, which means transferring the depth information to the 3D pixel cloud. Then simulate the gesture motion in virtual space, and calculate the 3D information of each point. This kind of method is more accurate than the former method, but the drawbacks are obvious, the computation is too large for a normal computer, if count the hand gesture recognition system model, it cannot be a real-time method under the usual current hardware [94].

In the recent years, there is one recognition model which is a fusion by depth image

and colour image. The common way is that apply the depth information to the colour image to use the depth information to isolate the hand gesture from colour image. Then put the processed colour image to the traditional recognition model. This kind of methods has more advantages than the former methods, it uses the accurate position information to get the gesture, and then apply the traditional system to the colour image to save the resources. It uses the depth information one time during gesture detection phase, although, it save the computation time, because the detection phase only use the depth data, the object, which has the same distance away from the camera with your hand, will highly effect the recognition results [95].

Hand detection is one of the most important phases of hand gesture recognition. It highly affected the recognition accuracy. So in this thesis, a novel hand gesture detection method is proposed, which is a combination of depth data and colour information, which uses both colour information and depth information during hand gesture detection. The key idea of this method is the multiple threshold settings to isolate the useful information from the depth image or the colour information alternately to achieve a better hand gesture.

This section is organised as follows, It will discusses the basic characteristics of the camera and the calibration of the Kinect. then it will present the method to set the threshold and do the first phase segmentation. The detailed procedure is given to describe how to further remove some useless part. The next part will illustrate the method to apply the threshold to hand detection to achieve a further clear hand. Then it will analyse the methods and meaning to do the region growing and corrosion, its not the necessary step for general hand detection, but for detailed high quality hand detection, its quite suitable. The last step of the system, the Kalman filter is chosen due to its high toleration for the sudden noise, and high performance for a continuous recognition.

3.2 Characteristics of Kinect camera and calibration

The Kinect sensor can achieve depth data and RGB colour image at the same time. It can also track object movement. The left lens is an infrared emitter with a common RGB colour camera in the middle and a 3D depth sensor is on the right. Kinect has focus tracking function with the base motor can rotate Kinect by around 270 degree. It also has an array of microphones. This allows Kinect to capture a colour image, 3D depth image and audio as shown in Fig. 3.3 [92].

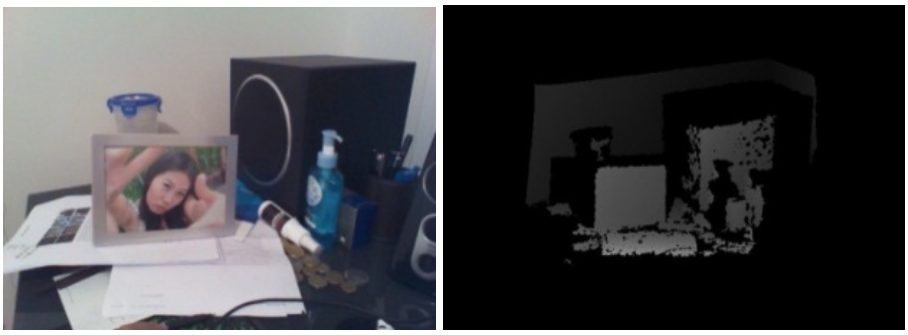

Figure 3.3: a colour image and the corresponding depth image

Compared to an ordinary camera, the Kinect has a CMOS infrared sensor, which is used to estimate the environment by using black and white spectrum. The pure black is on behalf of infinity faraway, pure white means infinity close, the grey area between black and white is corresponding to the distance between the point and camera. It collects every point in the space to form a comprehensive depth image of the surrounding environment. The sensor generates a depth image at 30 frames per second to rebuild the surrounding environment [96].

Compared with traditional cameras, the Kinect has many advantages, it work in real-time, and the depth data, which is sent to the next step process without additional computation. Besides, the depth data from Kinect will not be affected

by the light condition and clutter in the background. The depth camera generates depth data even at low lighting conditions [97]. Compared to the traditional hand detection methods, Kinect does not require the colour markers or fixed colour background. Even with overlap of two skin colour areas, it will not affect the detection result.

To use the Kinect camera to produce depth data, there are two usual methods; first is using the Microsoft SDK to achieve the data and another is to use Microsoft virtual studio to input the libraries of OpenCV and OpenNi, and then achieve the depth data from Kinect. Before producing the depth image and colour image at the same time, the camera should be calibrated. Because there is some distance between the depth camera and colour camera. The depth-Generator Get-Alternative View-Point-Cap sentence can use to adjust the view of two cameras to achieve the same image in virtual studio as shown in Fig. 3.3.

3.3 First time thresholding

Threshold of hand detection phase is very important to divide the points into different groups by their different characteristics. Because of the drawbacks of depth image and colour image in the hand detection phase, the threshold only applied to colour image or depth image may lead to different kinds of inaccuracies. In this thesis, thresholding will apply to the colour image and depth image for multiple times in order to achieve a clear hand gesture.

The first step is to apply the threshold to the depth data as the grey scale of each pixel on depth image is only related to the distance, so that the point which is closer to the camera is much brighter than a distant point. This step is used to

exclude the obvious background in depth data as visible in Fig.3.4. Any skin-tone object in the background will not affect the hand gesture which is in the foreground when using this depth thresholding simplifying age-old background separation problem in computer vision.

Figure 3.4: original depth image and the depth image after the first time thresholding, the background had been removed

A proper threshold can lead to a good segmentation result. In this research, the wide spread Ostu method was applied to look for a proper threshold by using grey level histograms [40]. The main idea is to select a threshold from the histogram which was derived from discriminant analysis point of view. The optimum threshold is determined by the discriminant criterion which maximises the discriminant measurement of separability of the resultant. A threshold, T, is setted. All the points, which their grey scale values are lower than T, will be dropped. In this phase, it will reduce the majority noisy signal in the image to achieve a relative clear and smaller area.

3.4 Overlapping depth image on colour image to remove the background

After segmentation using depth information, the foreground is identified in the depth information. This information can be utilised to threshold the colour image to remove the background. It is a process very similar to Logic AND operation where the foreground image which contain the hand region will preserve the area in the colour image. Everything else will be discarded in the colour image thereby obtaining the hand gesture in full colour.

In this phase, the image should be converted to HSV format, which is more convenient for image analysis. The above process can be mathematically described as follows: Assuming the total number of pixels in this phase is N. The H, S, and V represent hue, saturation and brightness respectively. The system should require three constraints to achieve the colour thresholding. First, by choosing the skin colour area and second requiring the saturation to be not white, and thirdly, the skin colour area should be bright in case of choosing the other object which has a similar colour with skin. So the constrains can be summarised as follows:

$$y = \begin{cases} -10 < H_y < 10 \\ S_y > ths \qquad y \in N \\ V_y > thv \end{cases} \tag{3.1}$$

And then, the threshold should be set again to isolate the skin colour area. After this process, only the skin colour area is kept in the image as shown in Fig. 3.6.

3.5 Total thresholding

After the above steps, the elements in image are clear, but there are still some unnecessary artefacts in the image, such as another arm. Final segmentation can remove this as shown in Fig. 3.5.

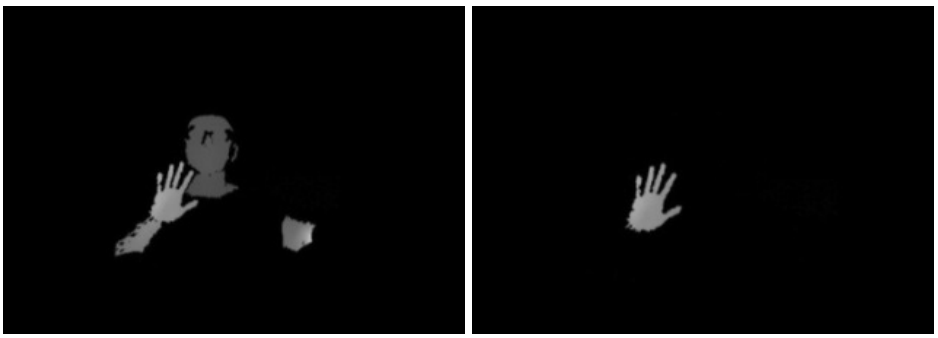

Figure 3.5: Left image shows the result of fusion of depth and colour information after applying the threshold to image, only the skin colour area is kept, right image shows what the detected hand looks like after the final phase processing.

These constraints can be mathematically expressed using the d_{min} referring to the shortest distance, d_{max} referring to the longest distance, y is a point in the image, G_y referring to the grey scale value, d_y is the distance between y and camera.

$$y = \begin{cases} d_{min} < D_y < d_{max} \\ G_y > \lambda \\ y \in N \end{cases} \tag{3.2}$$

3.6 Morphological filtering for smooth edges

In order to remove the jagged area, it needs a certain morphological filtering to

achieve a smooth edges for effective hand gesture recognition. A process known as region growing is shown in Fig. 3.6, which is very effective at producing a smooth edge. Before the process, a seed pixel of the image must be settled as a start point, and then the seed pixel will absorb a set of pixels [98], which have the similar characteristics with the seed pixel, in the neighbouring regions. The pixel of the image, which has no similar characteristics with any other pixels, will be settled as the new seed pixel. This process can repeat until there is no pixel to absorb, it means the system finish the region growing phase.

In mathematics, image expansion is that doing the convolution between image (called A) and core B. The core B can be any shape or size, it has an additional reference point. Generally, a core is a solid square or a disc with a reference point. Expansion is a method, which is used to get the local maximum value, and then assign the value to the reference point to make the highlight area of the image gradually increase [99].

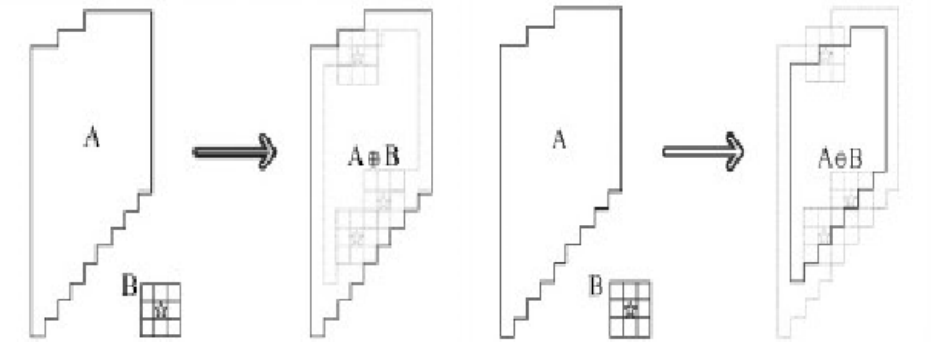

Figure 3.6: Left image shows the image expansion process from A to B,right image shows the image erosion process from A to B

Erosion phase of an image is on the contrary of expansion phase as shown in Fig. 3.6. It is used to record the minimum value of the pixel in the core region. The system will calculate the minimum value of a pixel in the area covered by B while the core B is doing the convolution with the image, and then place the value on

the reference point [100].

For this image expansion procedure, the set of seeds should be founded, assuming that the G_x is the grey scale value of the point x, the new threshold is represented by λ_{new}, the set of seed is S, the point in this area, which will be absorbed, could be summarised as follows:

$$X = \begin{cases} G_x > \lambda_n ew \\ \qquad\qquad\qquad x \in S \\ |x - S| > 1 \end{cases} \tag{3.3}$$

For this hand detection system, all these procedure are used to achieve a smooth contour, reliable hand shape to improve the precision of detection rate. The corrosion phase is mainly used for removing some fixed useless point of the image, because the final image is much smaller than before, so the noise filter is very important, and it will be easily affected the detection results.

3.7 Hand detection

In the OpenCV library, there is one filter which is widely used, because it has many advantages than other methods, such as good tracking ability, edge detection and so on. It is the Kalman filter. The main task of it is to track the value of a variable. The tracking is based on the equation of motion of the system to make a prediction. The prediction may have error, so that the Kalman filter uses another measuring instrument to measure the value of the variable, the measurement may also have the error. But these two values have different weight ratio, the Kalman filter is based on these two values to do a series of iterations to track the target [101].

In this paper, this detection and tracking system is attempted by two fundamental formulas. First, we need to introduce a system of discrete control process. The system can be described by a linear stochastic differential equation:

$$X(k) = AX(k-1) + BU(K) + W(k) \tag{3.4}$$

And the measurement from the system:

$$Z(k) = HX(k-1) + V(k) \tag{3.5}$$

In the above two equations, $X(k)$ is the system state at the k time, $U(k)$ is the control amount at the k time. A and B are two system parameters. $Z(k)$ is the measurement value at k time, H is the parameter of the measurement system. $W(k)$ and $V(k)$ are noises [102].

In this system, a fusion of colour and depth information is collected for Kalman filter, which keeps the detection system working well under the different lighting conditions and background clutter as shown in Fig.3.7. Even there are other people in the background, the system was not affected [103]. With use of the Kalman filter, the system will keep work accurately when some short term gestures appear in the scene, these useless gesture will not affect the detection and detection accuracy unless these gestures keep in the scene for a longer time than the main user.

The specific model chosen for the hand detection depends on the specific application and circumstance, for example, one hand or multiple hands, static or dynamic gesture, indoor or outdoor, different lighting condition and so on. In this

Figure 3.7: The four images show the performance of system

thesis, the Kalman filter is chosen for the study due to its good tracking ability and high tolerance of sudden noise, such as, a gesture suddenly appearing in the image and minimal computational requirements.

3.8 Conclusion

In this section, a novel technique was proposed to remove the background and skin-tone regions in the background for effective hand gesture recognition using fusion of depth information and colour image. The main advantage of this system is that this detection system has a strong tolerance of noise, complex lighting conditions and background clutter. Due to the low cost and easy use of Kinect, this system is also inexpensive to implement. Static gestures as well as dynamic gestures can be tracked and analysed using Kalman filter, which can highly improve the robustness of hand gesture recognition.

Chapter 4

Dynamic gesture recognition method of Kinect fusion fast entropy SVM

This chapter considers a fast recognition method using Kinect. Gesture recognition is an important and challenging task in the field of computer vision. Starting from the 3D shape of coding gestures, it puts forward a new kind of gesture recognition framework based on depth image. It extracts the space characteristics of a variety of 3D point cloud based on Kinect, including local principal components analysis on point cloud to get the histogram of main component, gradient direction histogram based on local depth difference and depth distribution histogram of local point cloud. Principal component histogram and gradient direction histogram effectively coding the local shape of gestures, depth distribution histogram

compensates the loss of the shaping descriptor information. Through preliminary training of random forest classifier to filter the characteristics, and characteristics with less influence on classification results are removed, thus the computational costs are reduced. The filtered characteristics are used for training of random forest classifier again to classify gestures. Experiment is carried on two large-scale gesture data sets, for more difficult ASL dataset [69], this method has improved the recognition rate of 3.6% than the best previous algorithm.

4.1 Problem and Motivation

Gestures are a natural and intuitive way of human communication, with the popularity of computing technology, gesture recognition based on computer vision has become an important research subject in the field of human-computer interaction. On the other hand, the commercial gesture recognition system, such as Leap Motion [104] etc. has become an alternative way of human-computer interaction in recent years. However, the development of this field is very rapid, but gesture recognition is still a very difficult problem. This is caused by the inherent flexibility and complexity of gesture itself. In recent years, with the emerging depth sensor such as Kinect [43], gesture recognition task has become more and more convenient. First, due to the depth image is not sensitive to light condition, gesture segmentation method based on depth threshold [105] is more simple and has more robustness than traditional gesture segmentation based on skin colour; second, compared with the traditional colour image the depth image provides additional distance information, which converts the gesture recognition from 2D image recognition problem into a 3D object recognition problem. Third, the depth image does not contain the information of colour and material of the object, thus it expresses the geometric shape of the object more purely, so it

is more convenient for the researchers to extract the characteristic based on shape.

4.2 Multiple spatial characteristics

4.2.1 Gestures character description

A given depth image containing gestures $d = I(x, y)$, where x and y is the position coordinates of the pixels in the image, d is the corresponding depth, with the range from 0 to 255, all depth values equal to 255 pixels are regarded as the background pixels [106]. This image has been made standardisation on the center of the gesture and the main direction of gesture, and only contains the extracted gesture part after gesture segmentation. Preprocessing part will be introduced in the experiment.

Set the size of the image as $M \times N$, and evenly divided into n_b image blocks of $\triangle I$, and $n_b = n_x \times n_y$. n_x and n_y denote the number of image block at x direction and y direction respectively. Set $\triangle x = M/n_x$, $\triangle y = N/n_y$, so the size of each image block is $\triangle x \times \triangle y$. This thesis extracts three different characteristics based on spatial information, finally put all the characteristics of the image blocks combined into a long vector to be the characteristics of the whole image, as shown in Fig. 4.1.

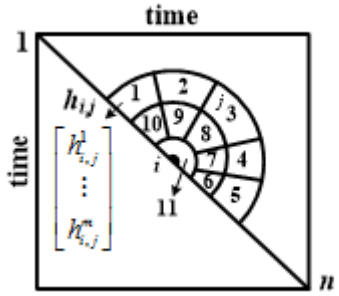

Figure 4.1: Local characteristic description

4.2.2 Principal component histogram

Principal component histogram was first put forward by Hossein et.al [107], used for gesture recognition. In order to describe three-dimensional shape of gestures, first convert depth image into point cloud in 3D space. For convenience, set $z = 255 - d$. So 3D point cloud Ω composed by all foreground pixels of this depth image can be represented as [108] [109]:

$$\Omega(x, y, z)|z \neq 0, \tag{4.1}$$

For any point $p \in \Omega$ in this point cloud, we define its local space as Ω_p and satisfies

$$\Omega_p = [q| \, ||q - p|| \, \leq r] \tag{4.2}$$

Where, p and q transform to (x, y, λ_z), λ converted to the proportion of conversion parameters of depth and plane coordinate, r is the distance parameter, they need to be debugged in the experiment[108].

Points in Ω_p has certain descriptive power on gestures surrounding, so it conducts principal component analysis on Ω_p.

Set n_p as the number of points in Ω_p, then covariance matrix C of point in Ω_p can be expressed as,

$$C = \frac{1}{n_p} \sum q \in \Omega_p (q - \mu)(q - \mu)^\gamma \tag{4.3}$$

where

$$\mu = \frac{1}{n_p} \sum q \in \Omega_p q \tag{4.4}$$

Make characteristic decomposition on C, then we have

$$CV = EV \tag{4.5}$$

E is the diagonal matrix, includes three characteristic values $\lambda_1 \geq \lambda_2 \geq \lambda_3$. V contains characteristic vectors $[v_1, v_2, v_3]$ of three characteristic values [110]. v_1 indicates the direction of the maximum variance, v_3 indicates the normal vector of the surface of 3D point cloud. They contain the local shape information of 3D point cloud. In order to carries on the quantification and coding on it, this thesis defined two projection methods, so as to avoid 180°ambiguity existed in characteristic vector, we regulate the component of characteristic vector on Z axis must be non-negative, otherwise each dimension will be inverted before projection.

4.2.3 Depth distribution histogram

Principal component histogram and gradient direction histogram are shape descriptor, for they do not have robustness on depth value changes of the same shape, and for the image block $\triangle I$, its depth value distribution also contains local information of the gesture. Depth distribution histogram solved the problem that shape descriptor is sensitive to the depth changes, and added the depth distribution information of gestures [111] [112].

Select minimum depth of all foreground pixels $d_{min} > 0$ and maximum depth d_{max}.

And $d_{max} - d_{min}$ is divided evenly into N_d segmentsand the size of each segment is

$$\triangle d = (d_{max} - d_{min})/N_d \tag{4.6}$$

For all foreground pixels in image block $\triangle I$, we determine its segment num according to its depth value, and construct depth distribution histogram H_{dd} based on this.

$$num = \left[\frac{I(x, y)}{\triangle d} \right] \tag{4.7}$$

$$H_i(num) = H_i(num) + 1, \quad \forall (x, y) \in \triangle i \tag{4.8}$$

Finally, the characteristic of depth distribution histogram of the whole image is

$$H_d d \ = \ [\ H_1, H_2, ..., H_n b\ ] \ \in \ R_{N_d n_b} \tag{4.9}$$

4.3 Kinect data gesture recognition steps

4.3.1 data acquisition

After extraction of various space characteristics, this paper fuses them into a long vector as global characteristics of the whole image. Due to characteristics included Histogram of Oriented Gradient (HOG) dense operator [113] can lead to character

dimension too high, and ”dimension disaster” caused expensive computational cost, so it needs dimension reduction by characteristics filtering. This thesis adopts the method of preliminary training of random forest to measure the importance of characteristics, so as to select a discriminant characteristic.

Because of the large noise in Kinect depth data, hands can be positioned as the most reliable marks in such 3D gesture model, based on 2D+3D algorithm [114], it can detect fingertips of 3D gesture under different expressions and gestures. However, in this case, the 3D data is high resolution, this thesis assumes that the tip position has been detected approximately. Because the fingertip only need face clipping and rough alignment, therefore, as long as the detected point is close enough to the real position, system can work normally.

4.3.2 Model building

Given the tip of the finger position, 3D gesture cutting can be done easily, this algorithm uses a sphere with radius of 8 cm to cut gestures, first of all to finger point cloud into the origin, then remove those points away from the origin more than 8 cm, thus can get only face 6D point cloud surface area.

Iterative closest point (ICP) algorithm [115] is based on a precise alignment technique, and its computational cost is very large, because different objects have different face shapes, reference gesture model must be the reliable expression of general 3D gesture, and can not be constructed with high noise level of the Kinect 3D data [116]. The proposed algorithm, therefore, through alignment scanning, and resampling on uniform grid, then take their average to build reference gestures, the reference gesture shall be 64 points between the center of two eyes,

and points on the ligature from lips center to the eyes are also 64, the complete gestures have 128×128 points, reference gesture trajectories are used as shown in Fig. 4.2, all gestures including training data and query gestures use six ICP iteration to get the reference gestures.

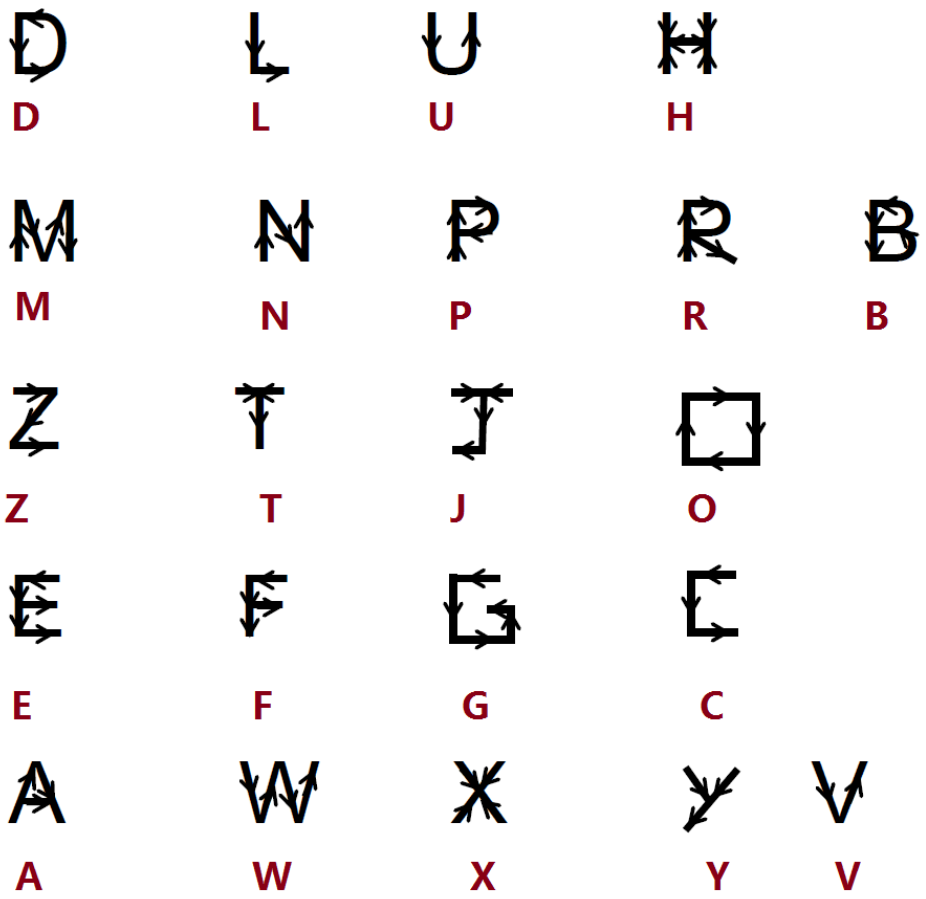

Figure 4.2: Reference letters

The dataset was captured by two kinds of hand gesture expression methods, the first method is using a single camera to film different gestures presented by one person before a fixed white colour wall, the light in the environment is sufficient. The second method is using a single camera to film different gestures presented by one person through a hole of one white cotton frame. All these two methods are filmed more than 20 times for one single gestures under different light conditions

and from different angles to robust the dataset, also, all these two methods are using fixed colour background to make the dataset reliable. It is shown in Fig 4.3.

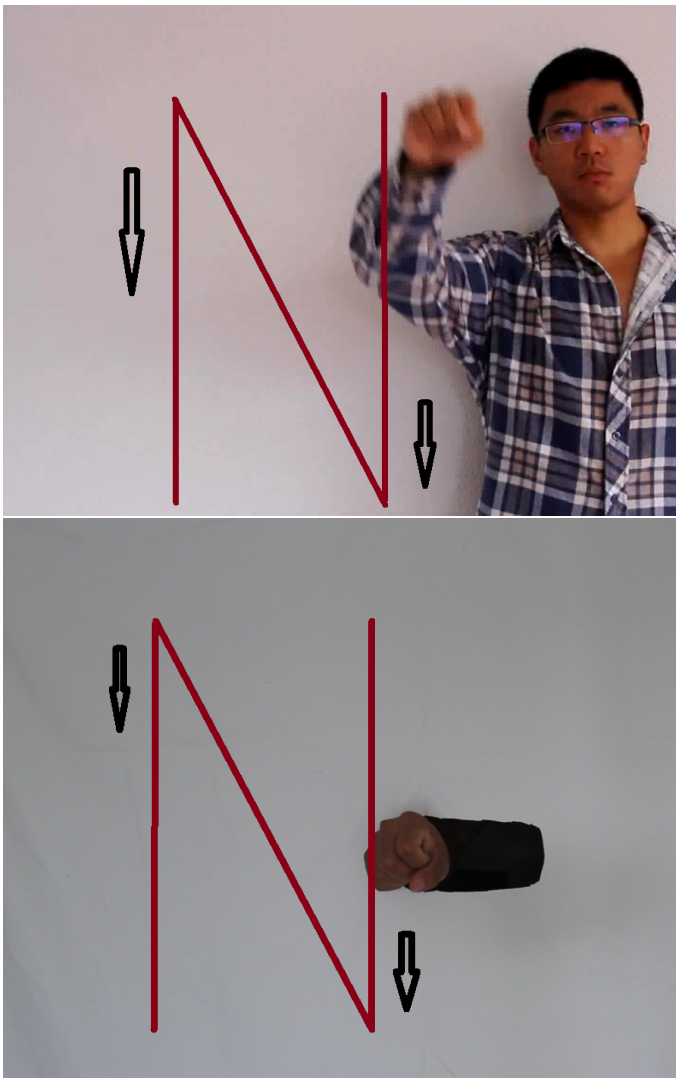

Figure 4.3: Two different hand gestures capture methods

After gesture correction, through X value of the original point cloud replaced with opposite ($-X$) to create a mirror point cloud. However, not all mirror points are useful, because the purpose of this study was to fill the missing data. In ideal condition, the positive gestures don't need to add points, and all points should be reflected in a profile view. For this, each mirror point, this paper calculates Euclidean distance of the nearest point at the origin cloud (XY value only), if the

distance is less than threshold , so mirror points are removed, when, and only when there is no neighbourhood points on a location to add mirror points. One shall note that, do not use Z when calculating the distance, because the difference of Z is usually caused by palm symmetry instead of missing data. Then to merge the rest of the mirror points and origin cloud.

4.3.3 Smoothing

Threshold σ can be spatial resolution based on sensor or point cloud itself, this value can be user defined. Depending on the initial sample density, too high σ value will produce a noise surface, while too low value is useless for symmetrical filling. Experiment has shown that different σ value taken from 1 to 5 mm had less influence on performance, when σ = 2mm a good balance can be achieved [117] [118].

Resampling has three main objectives:

1. It can remove the noise surface generated by Kinect sensor smoothly, and symmetric filling.

2. It may fill loopholes still existed after symmetric filling.

3. It reduces the influence of gesture alignment error on 2D grid caused by ICP registration. For this purpose, the algorithm will fit smooth surface to point cloud (XYZ), the algorithm will use similar instead of interpolation to fit curved surface to point, using a smoothing factor (or fitting) to perform

curved surface fitting, surface sudden bending is not allowed, so as to alleviate the influences of noise and outliers.

For every gesture, 128×128 points were uniform resampling, from the minimum X and Y to the maximum X and Y, the advantage of resampling from the minimum to the maximum is that, it can align the faces on 2D grid. There is no smooth texture, because it's not noisy, smooth will only make it become blurred. After resampling, X and Y grid will be discarded, and depth and four 128×128 matrices can be obtained, in order for further processing, they continue to the next sampling as 32×32 size.

4.4 Experiment

4.4.1 Data Set

This thesis carried on experiments on two gesture data sets of MMC data set [114] and ASL dataset [69]. Two data sets were collected from the depth gesture image of Microsoft Kinect, ASL dataset contained colour gesture image, but this paper did not use it.

MMC data set contained 1000 images, including 10 different types of gestures (from 0 to 9). Image acquisition from 10 individuals, that was, collected 10 images of each gesture on each person. The original image contained person and background, through gestures segmentation, gestures contained in the data set as shown in Fig. 4.4 .

ASL data set contained 60000 pieces of divided gesture images, contained 24 let-

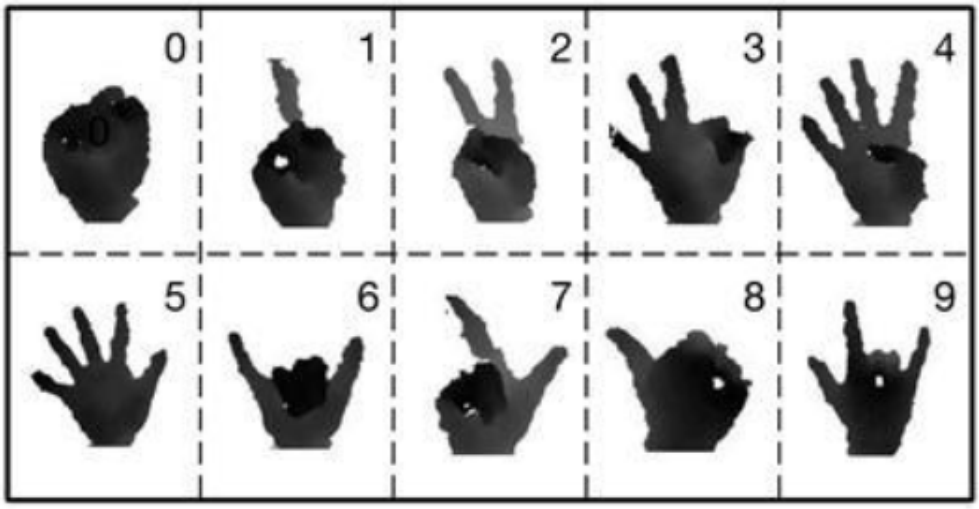

Figure 4.4: MMC Hand Digits Dataset

ters of sign language(from a to z, remove two dynamic hand gestures of j and z), collected from five people. Compared with MMC data set, the difference between gestures in this data set was smaller, and intra-class difference was bigger, which made the classification more difficult. The gestures of the data set as shown in Fig 4.4.

4.4.2 Parameter Setting

In order to compare with the current algorithm, this thesis adopted the same cross validation strategy of literature [119] and [120], namely, independence between objects and co-dependent between objects. For the samples collected from N individuals, independence between objects indicated that, with $N-1$ individuals as the training sample set, 1 individual as a test set, repeated for N times to make the training set and test set covering all situations, and then take the average accuracy; co-dependent between objects indicated that, all N individuals randomly and evenly divided into two parts, one part as the training set, the other part as the test set. Also take average accuracy after repeated N times.

For the original depth image, the required preprocessing steps, including gesture segmentation, image scale standardisation, the main direction standardisation of gestures. This thesis adopted the method of limited depth threshold for gesture segmentation: hands were regarded as the object most close to the depth camera, and selected a certain depth range of pixels as point cloud of gestures, and mapped the depth value to 0 to 255 of gray space, to generate gesture image.

For MMC data set, we also made more accurate gesture segmentation by calculating the palm range. Due to the image size was differ, the standardisation of image size was required, after experiment, we selected the best image size to be pixels of 120 height, pixels of 100 width. The standardisation of main direction of gestures can reduce inner class difference caused by the in-plane rotation, this paper set the direction of the principal component of foreground pixels found by PCA as the main direction of gestures, and rotating the image make the y axis as the main direction [121].

In the experiment, the size of the selected image block was 10×10, so the number of the image blocks was 120. For principal component histogram, positive icosahedron projection had 7200D, three plane projection had 6480D.Gradient direction histogram had 960D, depth distribution histogram had 1200D. After the characteristics filtering, we selected 2000 characteristics with high importance.

4.4.3 Experiments Results

Through systematic contrast experiments, the recognition rate of proposed method and the current gesture recognition algorithm on the two data sets as shown in

table 4.1 and table 4.2. As can be seen from the table that, on the two data sets, the proposed method has obtained better recognition rate than that of current method.

method	independent between objects	co-dependent between objects
Ren	0.939	N/A
HOG	0.931	0.964
H3DF	0.955	0.992
Our method 1	0.972	0.994
Our method 2	0.963	0.992

Table 4.1: Recognition rate of each method used on MMC data set

The proposed method 1 indicates positive icosahedron projection, the proposed method 2 refers to the three-plane projection. In addition, the experiments of two data sets, the recognition rate of two projection methods are similar, the effect of positive icosahedron projection is better than that of three-plane projection.

method	independent between objects	co-dependent between objects
Ren method	N/A	N/A
Bowden method	0.480	N/A
HOG	0.634	0.970
H3DF method	0.713	0.979
This method No.1	0.757	0.977
This method No.2	0.759	0.972

Table 4.2: Recognition rate of each method used on ASL data set

This thesis regarded the gestures as a 3D object to extract characteristics, without depending on the particular perspective and gestures, so there would not appear the condition of some gesture not supported. Method most close to the proposed method was [119], which also encoded the normal vector of 3D object surface.

Figure 4.5: Finger Spelling Dataset

	dissymmetry			symmetry		
gesture	D	T	fusion	D	T	fusion
positive	100	100	100	100	100	100
rotate ±30°	49.5	98.1	93.6	88.3	99.8	99.4
rotate ±60°	14.9	80.4	55.1	87.0	97.4	98.2
rotate ±90°	1.0	39.4	14.4	74.0	83.7	84.6
Tilt ±60°	77.2	91.3	90.0	81.6	89.1	92.8
average	46.2	87.6	77.0	85.4	95.0	96.3

Table 4.3: recognition rate of posture and gesture changes(%)

This thesis effectively expressed the 3D shape through local principal components analysis, and integrated the characteristics of more identifying information, thus the classification accuracy was improved. Table 4.2 provided the confusion matrix on ASL data set, data from Fig.4.5, it reflected the percentage relationship of real

category of sample and predicted category. It can be seen from the figure that, even though the proposed method improved the recognition rate, but for some gestures with similar appearance, such as the letter M and N, P and Q gestures, recognition error rate was still high.

4.5 Conclusion

This thesis proposed a new gesture recognition method based on multiple spatial characteristics Kinect sensor data. The principal component histogram and gradient direction histogram described the shape of gestures in different scales, and depth distribution histogram embodied the depth distribution of gestures. On this basis, this thesis calculated characteristics importance through the preliminary training of random forests and filtered characteristics. Experiment was carried on two large-scale gesture data sets, the results showed that, compared with the present popular gesture recognition algorithm, the proposed method can effectively improve the recognition effect. i will give consideration on how to extract characteristics of more discriminant information or using convolution neural network method to learn characteristics of gestures image automatically in the future.

Chapter 5

Conclusion

Hand extraction is a key technology in many interaction applications relied on hand gesture recognition. It ensures the input gestures will not interface by background. Up to now, most existing works only use information or depth information. After Kinect released, some studies consider using both depth and colour information in each step to improve the accuracy, this means the processing time of the system is limited by the huge coding logic and algorithm. Furthermore, past works on hand gesture recognition only focused on the different combination of filter and recognition model. In other words, they do not maximise the database and upgrade the thresholding logic to decrease the system reaction time. Another key assumption of past works is that the system has already been put into a relatively pure background or sufficient light condition. However, in reality, these system do not guarantee the recognition rate.

To fill in the gap, this thesis address the following research questions:

1. How to deploy a simple and effective logic to complete the gestures isolation process based on depth and colour information?

2. How to build a robust system system which can work in tough conditions, such as clutter background or skin colour area?

3. How to build a novel hand gesture recognition system that has a high recognition rate?

To address the first question, Chapter 3.3, Chapter 3.4 and Chapter 3.5 shows a new solution, three step isolation was applied to the gesture extraction stage. The first step is using the colour information to primarily remove the obvious noises, and then using the depth information to isolate the gesture range from the whole input image, finally, the system applies colour information to deep clean the noise in input image. It means the depth information is only deployed to the second step. Because of its special characteristic, the depth information is only related to distance, it means depth information will not be affected by the colour or light noise. On the other hand, the depth information is a ideal filter to remove these colour and light related noises.

Chapter 3.6 and Chapter 3.7 address the second question. To build a robust system, the system should work well from gestures acquiring, gesture isolation, features extraction to the final recognition stage. In this thesis, a Kinect camera is used to collect the hand gestures, because it has the depth camera, so it can also produce depth information with the colour information. It is a key requirement to protect the system from serious interface of clutter background. For the feature extraction stage, the Kalman filter was applied to hand detection stage, it is widely used because its good tracking and edge detection ability. All these advantages make the whole system robust.

The third research question can be found the answer in Chapter 4.2, Chapter 4.3 and Chapter 4.4. In this thesis, a novel hand gesture model was proposed to achieve a high recognition rate. So the fast entropy SVM is considered to deployed to the system. From dataset to recognition algorithm, this model has many good characteristics to improve the recognition accuracy, it has a huge training database that includes more than 50 high definition images for one single gestures. It originally applied to recognise some letters, but to compare its performance to the others, there are two database were built to do experiment tests. One is for the ASL, another is for the MMC which is a recognition model has very high recognition rate. The experiment shows model in this thesis achieved a higher rate recognition rate on average.

A key future research direction for hand gesture recognition is to make the hand gesture recognition technologies apply to many different applications whereby they are able to do the specific requirement for users. For example, because hand gesture recognition technology does not need the user contact device. So hand gesture can use to unlock smart phones screen, it is very convenient when hands are wet, such as, cooking or doing sports, it will also not make your phone dirty. Hand gestures can also use in engineering area. For example, using hand gestures protect workers from the dangerous working place, such as steel-making furnace workshop. Hand gestures have a great prospect, especially under the background of hardware revolution right now.

Bibliography

[1] S. Yang, P. Premaratne, and P. Vial, "Hand gesture recognition: An overview," in *Broadband Network Multimedia Technology (IC-BNMT), 2013 5th IEEE International Conference on*, pp. 63–69, Nov 2013.

[2] B. A. Myers, "A brief history of human-computer interaction technology," *interactions*, vol. 5, pp. 44–54, Mar. 1998.

[3] N. Berci and P. Szolgay, "Vision based human-machine interface via hand gestures," in *Circuit Theory and Design, 2007. ECCTD 2007. 18th European Conference on*, pp. 496–499, Aug 2007.

[4] P. Premaratne, Q. Nguyen, and M. Premaratne, *Advanced Intelligent Computing Theories and Applications: 6th International Conference on Intelligent Computing, ICIC 2010, Changsha, China, August 18-21, 2010. Proceedings*, ch. Human Computer Interaction Using Hand Gestures, pp. 381–386. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010.

[5] J. J. LaViola, Jr., "A survey of hand posture and gesture recognition techniques and technology," tech. rep., Providence, RI, USA, 1999.

[6] A. R. Sarkar, G. Sanyal, and S. Majumder, "Hand gesture recognition systems: a survey," *International Journal of Computer Applications*, vol. 71, no. 15, 2013.

[7] K. R. Shastry, M. Ravindran, M. Srikanth, N. Lakshmikhanth, *et al.*, "Survey on various gesture recognition techniques for interfacing machines based on ambient intelligence," *arXiv preprint arXiv:1012.0084*, 2010.

[8] A. R. Sarkar, G. Sanyal, and S. Majumder, "Article: Hand gesture recognition systems: A survey," *International Journal of Computer Applications*, vol. 71, pp. 25–37, June 2013. Full text available.

[9] W. Jingqiu and Z. Ting, "An arm-based embedded gesture recognition system using a data glove," in *Control and Decision Conference (2014 CCDC), The 26th Chinese*, pp. 1580–1584, May 2014.

[10] Y. Yao and Y. Fu, "Contour model-based hand-gesture recognition using the kinect sensor," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, pp. 1935–1944, Nov 2014.

[11] R. R. Yan, K. P. K. P. Tee, Y. Y. Chua, H. H. Li, and H. H. Tang, "Gesture recognition based on localist attractor networks with application to robot control [application notes]," *IEEE Computational Intelligence Magazine*, vol. 7, pp. 64–74, Feb 2012.

[12] S. H. Lee, M. K. Sohn, D. J. Kim, B. Kim, and H. Kim, "Smart tv interaction system using face and hand gesture recognition," in *Consumer Electronics (ICCE), 2013 IEEE International Conference on*, pp. 173–174, Jan 2013.

[13] B. W. Miners, O. A. Basir, and M. S. Kamel, "Understanding hand gestures using approximate graph matching," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 35, pp. 239–248, March 2005.

[14] J. Lee, Y. Lee, E. Lee, and S. Hong, "Hand region extraction and gesture recognition from video stream with complex background through entropy analysis," in *Engineering in Medicine and Biology Society, 2004. IEMBS '04.*

26th Annual International Conference of the IEEE, vol. 1, pp. 1513–1516, Sept 2004.

[15] D. Chai and K. N. Ngan, "Face segmentation using skin-color map in videophone applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, pp. 551–564, Jun 1999.

[16] N. Habili, C. C. Lim, and A. Moini, "Segmentation of the face and hands in sign language video sequences using color and motion cues," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, pp. 1086–1097, Aug 2004.

[17] S. Kim, G. Park, S. Yim, S. Choi, and S. Choi, "Gesture-recognizing handheld interface with vibrotactile feedback for 3d interaction," *IEEE Transactions on Consumer Electronics*, vol. 55, pp. 1169–1177, August 2009.

[18] S. Y. Lin, Y. C. Lai, L. W. Chan, and Y. P. Hung, "Real-time 3d model-based gesture tracking for multimedia control," in *Pattern Recognition (ICPR), 2010 20th International Conference on*, pp. 3822–3825, Aug 2010.

[19] Y. Azoz, L. Devi, and R. Sharma, "Reliable tracking of human arm dynamics by multiple cue integration and constraint fusion," in *Computer Vision and Pattern Recognition, 1998. Proceedings. 1998 IEEE Computer Society Conference on*, pp. 905–910, Jun 1998.

[20] C. A. Pickering, K. J. Burnham, and M. J. Richardson, "A research study of hand gesture recognition technologies and applications for human vehicle interaction," Citeseer.

[21] H. Cheng, L. Yang, and Z. Liu, "A survey on 3d hand gesture recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. PP, no. 99, pp. 1–1, 2015.

[22] F. M. Ciaramello and S. S. Hemami, "A computational intelligibility model for assessment and compression of american sign language video," *IEEE Transactions on Image Processing*, vol. 20, pp. 3014–3027, Nov 2011.

[23] R. C. Luo and Y. C. Wu, "Hand gesture recognition for human-robot interaction for service robot," in *Multisensor Fusion and Integration for Intelligent Systems (MFI), 2012 IEEE Conference on*, pp. 318–323, Sept 2012.

[24] H. K. Yun and B. H. Song, "Dynamic characteristic analysis of users' motions for human smartphone interface," in *Computing and Networking Technology (ICCNT), 2012 8th International Conference on*, pp. 395–398, Aug 2012.

[25] K. Dabre and S. Dholay, "Machine learning model for sign language interpretation using webcam images," in *Circuits, Systems, Communication and Information Technology Applications (CSCITA), 2014 International Conference on*, pp. 317–321, April 2014.

[26] M. S. Lew, N. Sebe, C. Djeraba, and R. Jain, "Content-based multimedia information retrieval: State of the art and challenges," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 2, no. 1, pp. 1–19, 2006.

[27] A. Chaudhary, J. L. Raheja, K. Das, and S. Raheja, "Intelligent approaches to interact with machines using hand gesture recognition in natural way: a survey," *arXiv preprint arXiv:1303.2292*, 2013.

[28] L. Dipietro, A. M. Sabatini, and P. Dario, "A survey of glove-based systems and their applications," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, pp. 461–482, July 2008.

[29] R. Xu, S. Zhou, and W. J. Li, "Mems accelerometer based nonspecific-user hand gesture recognition," *IEEE Sensors Journal*, vol. 12, pp. 1166–1173, May 2012.

[30] S. G. Wysoski, M. V. Lamar, S. Kuroyanagi, and A. Iwata, "A rotation invariant approach on static-gesture recognition using boundary histograms and neural networks," in *Neural Information Processing, 2002. ICONIP '02. Proceedings of the 9th International Conference on*, vol. 4, pp. 2137–2141 vol.4, Nov 2002.

[31] H.-C. Lee, C.-Y. Shih, and T.-M. Lin, *Advances in Intelligent Systems and Applications - Volume 2: Proceedings of the International Computer Symposium ICS 2012 Held at Hualien, Taiwan, December 12–14, 2012*, ch. Computer-Vision Based Hand Gesture Recognition and Its Application in Iphone, pp. 487–497. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013.

[32] M. M. Hasan and P. K. Misra, "Gesture recognition using modified hsv segmentation," in *Communication Systems and Network Technologies (CSNT), 2011 International Conference on*, pp. 328–332, June 2011.

[33] C. W. Chang and C. H. Chang, "A two-hand multi-point gesture recognition system based on adaptive skin color model," in *Consumer Electronics, Communications and Networks (CECNet), 2011 International Conference on*, pp. 2901–2904, April 2011.

[34] T. Kirishima, K. Sato, and K. Chihara, "Real-time gesture recognition by learning and selective control of visual interest points," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, pp. 351–364, March 2005.

[35] D. Chai and A. Bouzerdoum, "A bayesian approach to skin color classification in ycbcr color space," in *TENCON 2000. Proceedings*, vol. 2, pp. 421–424 vol.2, 2000.

[36] D. Osorio and M. Vorobyev, "A review of the evolution of animal colour vision and visual communication signals," *Vision research*, vol. 48, no. 20, pp. 2042–2051, 2008.

[37] H. Cooper, E.-J. Ong, N. Pugeault, and R. Bowden, "Sign language recognition using sub-units," *J. Mach. Learn. Res.*, vol. 13, pp. 2205–2231, July 2012.

[38] V. Vezhnevets, V. Sazonov, and A. Andreeva, "A survey on pixel-based skin color detection techniques," in *IN PROC. GRAPHICON-2003*, pp. 85–92, 2003.

[39] S. Zhao, W. Tan, S. Wen, and Y. Liu, *Intelligent Robotics and Applications: First International Conference, ICIRA 2008, Wuhan, China, October 15-17, 2008, Proceedings, Part I*, ch. An Improved Algorithm of Hand Gesture Recognition under Intricate Background, pp. 786–794. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008.

[40] "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, pp. 62–66, Jan 1979.

[41] L. Lamberti and F. Camastra, *Image Analysis and Processing – ICIAP 2011: 16th International Conference, Ravenna, Italy, September 14-16, 2011, Proceedings, Part I*, ch. Real-Time Hand Gesture Recognition Using a Color Glove, pp. 365–373. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011.

[42] E. Stergiopoulou and N. Papamarkos, "Hand gesture recognition using a neural network shape fitting technique," *Eng. Appl. Artif. Intell.*, vol. 22, pp. 1141–1158, Dec. 2009.

[43] J. Han, L. Shao, D. Xu, and J. Shotton, "Enhanced computer vision with microsoft kinect sensor: A review," *IEEE Transactions on Cybernetics*, vol. 43, pp. 1318–1334, Oct 2013.

[44] W. L. Chen, C. H. Wu, and C. H. Lin, "Depth-based hand gesture recognition using hand movements and defects," in *Next-Generation Electronics (ISNE), 2015 International Symposium on*, pp. 1–4, May 2015.

[45] Y. Wang, C. Yang, X. Wu, S. Xu, and H. Li, "Kinect based dynamic hand gesture recognition algorithm research," in *Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2012 4th International Conference on*, vol. 1, pp. 274–279, Aug 2012.

[46] Y. Chen, B. Luo, Y. L. Chen, G. Liang, and X. Wu, "A real-time dynamic hand gesture recognition system using kinect sensor," in *2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 2026–2030, Dec 2015.

[47] K. K. Biswas and S. K. Basu, "Gesture recognition using microsoft kinect.," in *ICARA*, pp. 100–103, IEEE, 2011.

[48] S. Hafiane, Y. Salih, and A. S. Malik, "3d hand recognition for telerobotics," in *Computers Informatics (ISCI), 2013 IEEE Symposium on*, pp. 132–137, April 2013.

[49] K. Y. Fok, N. Ganganath, C. T. Cheng, and C. K. Tse, "A real-time asl recognition system using leap motion sensors," in *Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2015 International Conference on*, pp. 411–414, Sept 2015.

[50] F. Schubert and K. Mikolajczyk, "Performance evaluation of image filtering for classification and retrieval," in *ICPRAM 2013-Proceedings of the 2nd International Conference on Pattern Recognition Applications and Methods*, pp. 485–491, 2013.

[51] H. A. Jalab, "Image retrieval system based on color layout descriptor and gabor filters," in *Open Systems (ICOS), 2011 IEEE Conference on*, pp. 32–36, Sept 2011.

[52] A. Malima, E. Ozgur, and M. Cetin, "A fast algorithm for vision-based hand gesture recognition for robot control," in *Signal Processing and Communications Applications, 2006 IEEE 14th*, pp. 1–4, April 2006.

[53] P. Premaratne and M. Premaratne, *Emerging Intelligent Computing Technology and Applications: 8th International Conference, ICIC 2012, Huangshan, China, July 25-29, 2012. Proceedings*, ch. New Structural Similarity Measure for Image Comparison, pp. 292–297. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012.

[54] P. Premaratne and M. Premaratne, "Image matching using moment invariants," *Neurocomputing*, vol. 137, pp. 65 – 70, 2014. Advanced Intelligent Computing Theories and MethodologiesSelected papers from the 2012 Eighth International Conference on Intelligent Computing (ICIC 2012).

[55] P. Premaratne, *Examples of Invariant Properties of Hu Moments*. Springer, 2014.

[56] P. Premaratne, F. Safaei, and Q. Nguyen, *Intelligent Computing in Signal Processing and Pattern Recognition: International Conference on Intelligent Computing, ICIC 2006 Kunming, China, August 16–19, 2006*, ch. Moment Invariant Based Control System Using Hand Gestures, pp. 322–333. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006.

[57] L. Li, D. Jia, X. Chen, and L. Sun, "A fast discrete moment invariant algorithm and its application on pattern recognition," in *Intelligent Control and Automation, 2006. WCICA 2006. The Sixth World Congress on*, vol. 2, pp. 9773–9777, 2006.

[58] B.-W. Min, H.-S. Yoon, J. Soh, Y.-M. Yang, and T. Ejima, "Hand gesture recognition using hidden markov models," in *Systems, Man, and Cybernetics, 1997. Computational Cybernetics and Simulation., 1997 IEEE International Conference on*, vol. 5, pp. 4232–4235 vol.5, Oct 1997.

[59] S. G. Wysoski, M. V. Lamar, S. Kuroyanagi, and A. Iwata, "A rotation invariant approach on static-gesture recognition using boundary histograms and neural networks," in *Neural Information Processing, 2002. ICONIP '02.*

Proceedings of the 9th International Conference on, vol. 4, pp. 2137–2141 vol.4, Nov 2002.

[60] J. Li, H. Chen, G. Li, B. He, Y. Zhang, and X. Tao, "Salient object detection based on meanshift filtering and fusion of colour information," *IET Image Processing*, vol. 9, no. 11, pp. 977–985, 2015.

[61] M. R. Basheer and S. Jagannathan, "Localization and tracking of objects using cross-correlation of shadow fading noise," *IEEE Transactions on Mobile Computing*, vol. 13, pp. 2293–2305, Oct 2014.

[62] H. Fei and I. Reid, *Computer Vision - ECCV 2004: 8th European Conference on Computer Vision, Prague, Czech Republic, May 11-14, 2004. Proceedings, Part III*, ch. Joint Bayes Filter: A Hybrid Tracker for Non-rigid Hand Motion Recognition, pp. 497–508. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004.

[63] Y. W. Lee, *Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence: 4th International Conference on Intelligent Computing, ICIC 2008 Shanghai, China, September 15-18, 2008 Proceedings*, ch. Application of the Particle Filter for Simple Gesture Recognition, pp. 534–540. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008.

[64] C.-B. Park and S.-W. Lee, "Real-time 3d pointing gesture recognition for mobile robots with cascade {HMM} and particle filter," *Image and Vision Computing*, vol. 29, no. 1, pp. 51 – 63, 2011.

[65] L. Bretzner, I. Laptev, and T. Lindeberg, "Hand gesture recognition using multi-scale colour features, hierarchical models and particle filtering," in *Automatic Face and Gesture Recognition, 2002. Proceedings. Fifth IEEE International Conference on*, pp. 423–428, May 2002.

[66] S. Sedai, M. Bennamoun, and D. Q. Huynh, "A gaussian process guided

particle filter for tracking 3d human pose in video," *IEEE Transactions on Image Processing*, vol. 22, pp. 4286–4300, Nov 2013.

[67] D. Miao, J. Fu, Y. Lu, S. Li, and C. W. Chen, "Layered compression for high-precision depth data," *IEEE Transactions on Image Processing*, vol. 24, pp. 5492–5504, Dec 2015.

[68] Z. Yang, L. Zicheng, and C. Hong, "Rgb-depth feature for 3d human activity recognition," *China Communications*, vol. 10, pp. 93–103, July 2013.

[69] P. Premaratne, S. Yang, Z. Zou, and P. Vial, *Intelligent Computing Theories and Technology: 9th International Conference, ICIC 2013, Nanning, China, July 28-31, 2013. Proceedings*, ch. Australian Sign Language Recognition Using Moment Invariants, pp. 509–514. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013.

[70] G. Plouffe and A. M. Cretu, "Static and dynamic hand gesture recognition in depth data using dynamic time warping," *IEEE Transactions on Instrumentation and Measurement*, vol. 65, pp. 305–316, Feb 2016.

[71] J. Sell and P. O'Connor, "The xbox one system on a chip and kinect sensor," *IEEE Micro*, vol. 34, pp. 44–53, Mar 2014.

[72] T. H. Reiss, "The revised fundamental theorem of moment invariants," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, pp. 830–834, Aug 1991.

[73] J. Flusser, "On the independence of rotation moment invariants," *Pattern Recognition*, vol. 33, no. 9, pp. 1405 – 1410, 2000.

[74] P. D. Vecchio and A. Salvini, "Neural network and fourier descriptor macromodeling dynamic hysteresis," *IEEE Transactions on Magnetics*, vol. 36, pp. 1246–1249, Jul 2000.

[75] H. T. Sheu and M. F. Wu, "Fourier descriptor based technique for reconstructing 3d contours from stereo images," *IEE Proceedings - Vision, Image and Signal Processing*, vol. 142, pp. 95–104, Apr 1995.

[76] P. R. G. Harding and T. Ellis, "Recognizing hand gesture using fourier descriptors," in *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, vol. 3, pp. 286–289 Vol.3, Aug 2004.

[77] S. S. Rautaray and A. Agrawal, "Real time gesture recognition system for interaction in dynamic environment," *Procedia Technology*, vol. 4, pp. 595 – 599, 2012. 2nd International Conference on Computer, Communication, Control and Information Technology(C3IT-2012) on February 25 - 26, 2012.

[78] X. Shen, G. Hua, L. Williams, and Y. Wu, "Dynamic hand gesture recognition: An exemplar-based approach from motion divergence fields," *Image and Vision Computing*, vol. 30, no. 3, pp. 227 – 235, 2012. Best of Automatic Face and Gesture Recognition 2011.

[79] V. N. Vapnik, *The Nature of Statistical Learning Theory*. New York, NY, USA: Springer-Verlag New York, Inc., 1995.

[80] V. Cherkassky, "The nature of statistical learning theory ," *IEEE Transactions on Neural Networks*, vol. 8, pp. 1564–1564, Nov 1997.

[81] V. N. Vapnik, "An overview of statistical learning theory," *IEEE Transactions on Neural Networks*, vol. 10, pp. 988–999, Sep 1999.

[82] P. Premaratne, S. Ajaz, and M. Premaratne, *Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence: 7th International Conference, ICIC 2011, Zhengzhou, China, August 11-14, 2011, Revised Selected Papers*, ch. Hand Gesture Tracking and Recognition System for Control of Consumer Electronics, pp. 588–593. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012.

[83] P. Doliotis, A. Stefan, C. McMurrough, D. Eckhard, and V. Athitsos, "Comparing gesture recognition accuracy using color and depth information," in *Proceedings of the 4th International Conference on PErvasive Technologies Related to Assistive Environments*, PETRA '11, (New York, NY, USA), pp. 20:1–20:7, ACM, 2011.

[84] P. Premaratne, S. Ajaz, and M. Premaratne, "Hand gesture tracking and recognition system using lucaskanade algorithms for control of consumer electronics," *Neurocomputing*, vol. 116, pp. 242 – 249, 2013. Advanced Theory and Methodology in Intelligent ComputingSelected Papers from the Seventh International Conference on Intelligent Computing (ICIC 2011).

[85] N. A. Ibraheem, R. Z. Khan, and M. M. Hasan, "Article: Comparative study of skin color based segmentation techniques," *International Journal of Applied Information Systems*, vol. 5, pp. 24–34, August 2013. Published by Foundation of Computer Science, New York, USA.

[86] S. Oprisescu, C. Rasche, and B. Su, "Automatic static hand gesture recognition using tof cameras," in *Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European*, pp. 2748–2751, Aug 2012.

[87] K. Lai, J. Konrad, and P. Ishwar, "A gesture-driven computer interface using kinect," in *Image Analysis and Interpretation (SSIAI), 2012 IEEE Southwest Symposium on*, pp. 185–188, April 2012.

[88] J. Suarez and R. R. Murphy, "Hand gesture recognition with depth images: A review," in *RO-MAN, 2012 IEEE*, pp. 411–417, Sept 2012.

[89] A. Kolb, E. Barth, R. Koch, and R. Larsen, "Time-of-Flight Cameras in Computer Graphics," *Computer Graphics Forum*, 2010.

[90] Z. Ren, J. Yuan, J. Meng, and Z. Zhang, "Robust part-based hand gesture

recognition using kinect sensor," *IEEE Transactions on Multimedia*, vol. 15, pp. 1110–1120, Aug 2013.

[91] Z. Ren, J. Meng, J. Yuan, and Z. Zhang, "Robust hand gesture recognition with kinect sensor," in *Proceedings of the 19th ACM International Conference on Multimedia*, MM '11, (New York, NY, USA), pp. 759–760, ACM, 2011.

[92] Y. Li, "Hand gesture recognition using kinect," in *Software Engineering and Service Science (ICSESS), 2012 IEEE 3rd International Conference on*, pp. 196–199, June 2012.

[93] Y. Kuno, M. Sakamoto, K. Sakata, and Y. Shirai, "Vision-based human interface with user-centered frame," in *Intelligent Robots and Systems '94. 'Advanced Robotic Systems and the Real World', IROS '94. Proceedings of the IEEE/RSJ/GI International Conference on*, vol. 3, pp. 2023–2029 vol.3, Sep 1994.

[94] P. Premaratne and Q. Nguyen, "Consumer electronics control system based on hand gesture moment invariants," *IET Computer Vision*, vol. 1, pp. 35–41, March 2007.

[95] Z. Zou, P. Premaratne, R. Monaragala, N. Bandara, and M. Premaratne, "Dynamic hand gesture recognition system using moment invariants," in *Information and Automation for Sustainability (ICIAFs), 2010 5th International Conference on*, pp. 108–113, Dec 2010.

[96] M. Tang, "Recognizing hand gestures with microsofts kinect,"

[97] J. Suarez and R. R. Murphy, "Hand gesture recognition with depth images: A review," in *RO-MAN, 2012 IEEE*, pp. 411–417, Sept 2012.

[98] Y.-L. Chang and X. Li, "Adaptive image region-growing," *IEEE Transactions on Image Processing*, vol. 3, pp. 868–872, Nov 1994.

[99] N. Ikonomatakis, K. N. Plataniotis, M. Zervakis, and A. N. Venetsanopoulos, "Region growing and region merging image segmentation," in *Digital Signal*

Processing Proceedings, 1997. DSP 97., 1997 13th International Conference on, vol. 1, pp. 299–302 vol.1, Jul 1997.

[100] M. Tabb and N. Ahuja, "Multiscale image segmentation by integrated edge and region detection," *IEEE Transactions on Image Processing*, vol. 6, pp. 642–655, May 1997.

[101] R. Chen and J. S. Liu, "Mixture kalman filters," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 62, no. 3, pp. 493–508, 2000.

[102] R. Chen and J. S. Liu, "Mixture kalman filters," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 62, no. 3, pp. 493–508, 2000.

[103] R. Faragher *et al.*, "Understanding the basis of the kalman filter via a simple and intuitive derivation," 2012.

[104] G. Marin, F. Dominio, and P. Zanuttigh, "Hand gesture recognition with leap motion and kinect devices," in *Image Processing (ICIP), 2014 IEEE International Conference on*, pp. 1565–1569, Oct 2014.

[105] G. Marin, F. Dominio, and P. Zanuttigh, "Hand gesture recognition with leap motion and kinect devices," in *Image Processing (ICIP), 2014 IEEE International Conference on*, pp. 1565–1569, Oct 2014.

[106] M. K. Bhuyan, D. Ghosh, and P. K. Bora, "Feature extraction from 2d gesture trajectory in dynamic hand gesture recognition," in *Cybernetics and Intelligent Systems, 2006 IEEE Conference on*, pp. 1–6, June 2006.

[107] H. Rahmani, A. Mahmood, D. Q Huynh, and A. Mian, *Computer Vision – ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part II*, ch. HOPC: Histogram of Oriented Principal Components of 3D Pointclouds for Action Recognition, pp. 742–757. Cham: Springer International Publishing, 2014.

[108] T. Jombart, S. Devillard, and F. Balloux, "Discriminant analysis of principal components: a new method for the analysis of genetically structured populations," *BMC genetics*, vol. 11, no. 1, p. 94, 2010.

[109] P. Nagabhushan and R. Pradeep Kumar, *Advances in Neural Networks – ISNN 2007: 4th International Symposium on Neural Networks, ISNN 2007, Nanjing, China, June 3-7, 2007, Proceedings, Part II*, ch. Histogram PCA, pp. 1012–1021. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007.

[110] O. Rodrıguez and A. Pacheco, "Applications of histogram principal components analysis,"

[111] H. Mousavi and C. Zaniolo, "Fast and accurate computation of equi-depth histograms over data streams," in *Proceedings of the 14th International Conference on Extending Database Technology*, pp. 69–80, ACM, 2011.

[112] M. Kisacanin, BranislavandGelautz, *AdvancesinEmbeddedComputerVision*. ChamHeidelbergNewYorkDordrechtLondon: SpringerInternationalPublishingSwitzerland, 2014.

[113] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, pp. 886–893 vol. 1, June 2005.

[114] X. Li, Z. Lv, J. Hu, B. Zhang, L. Shi, and S. Feng, "Xearth: A 3d gis platform for managing massive city information," in *Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA), 2015 IEEE International Conference on*, pp. 1–6, June 2015.

[115] X. Zhang, C. Glennie, and A. Kusari, "Change detection from differential airborne lidar using a weighted anisotropic iterative closest point algorithm," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 8, pp. 3338–3346, July 2015.

[116] C. Li, J. Xue, S. Du, and N. Zheng, "A fast multi-resolution iterative closest point algorithm," in *Pattern Recognition (CCPR), 2010 Chinese Conference on*, pp. 1–5, Oct 2010.

[117] H. Huang, Z. Ju, and H. Liu, "Real-time hand gesture feature extraction using depth data," in *Machine Learning and Cybernetics (ICMLC), 2014 International Conference on*, vol. 1, pp. 206–213, July 2014.

[118] C. Yu, X. Wang, H. Huang, J. Shen, and K. Wu, "Vision-based hand gesture recognition using combinational features," in *Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), 2010 Sixth International Conference on*, pp. 543–546, Oct 2010.

[119] G. Bao, L. Mi, Y. Geng, K. Pahlavan, *et al.*, "A computer vision based speed estimation technique for localiz ing the wireless capsule endoscope inside small intestine,"

[120] X. Song and Y. Geng, "Distributed community detection optimization algorithm for complex networks," *Journal of Networks*, vol. 9, no. 10, 2014.

[121] J. M. Kim and M. K. Song, "Three dimensional gesture recognition using pca of stereo images and modified matching algorithm," in *Fuzzy Systems and Knowledge Discovery, 2008. FSKD '08. Fifth International Conference on*, vol. 4, pp. 116–120, Oct 2008.