Visual intention detection algorithm for wheelchair motion

Thierry Luhandjula

UNIVERSITÉ PARIS-EST

Tshwane University of Technology
We empower people

ÉCOLE DOCTORALE
«Mathématiques, Sciences et Techniques de l'Information et de la Communication»

Thèse de doctorat

Spécialité : Informatique

Thierry Kalonda Luhandjula

Algorithme de reconnaissance visuelle d'intentions. Application au pilotage automatique d'un fauteuil roulant

July 2012

COMPOSITION DU JURY

Michel VERLEYSEN, Professeur,	Université Catholique de Louvain, Belgique	Rapporteur
Tania S DOUGLAS, Professeur,	Université de Cape Town, RSA	Rapporteur
Jean-Luc ZARADER, Professeur,	Université Pierre et Marie Curie, France	Examinateur
Latifa OUKHELLOU, DR,	IFSTTAR, France	Examinateur
Yskander HAMAM, Professeur,	F'SATI, TUT, Pretoria, RSA	Examinateur
Karim DJOUANI, Professeur,	Université Paris-Est, Créteil, France	Examinateur
Barend J VAN WYK, Professeur,	F'SATI, TUT, Pretoria, RSA	Directeur de Thèse
Yacine AMIRAT, Professeur,	Université Paris-Est, Créteil, France	Directeur de Thèse

Declaration and Copyright

I hereby declare that the thesis submitted for the degree of Doctorate Technology: Engineering: Electrical, at the Tshwane University of Technology, is my own original work and has not previously been submitted to any other institution of higher education. I further declare that all sources cited or quoted are indicated and acknowledged by means of a comprehensive list of references.

K.T. LUHANDJULA

Dedication

My dedication goes to my Lord and Saviour whose grace has been vital

for the completion of this work

The dedication extends to Martina, Angelo and Anthony

And

To my parents and siblings

Acknowledgments

I would like to express my deepest gratitude to my supervisors, Prof. Barend Jacobus VAN WYK, Prof. Yskander HAMAM, Prof. Karim DJOUANI, Prof. Yacine AMIRAT and Dr. Quentin WILLIAMS for their encouragement, suggestions, guidance, invaluable counsel and friendship that highly contributed to the completion of this work. It has been a true privilege to be exposed to their expertise.

I would like to thank all the staff members of the CSIR Meraka Institute for a conducive, interactive and encouraging atmosphere during my research. My appreciation goes to Guillaume OLVRIN who welcomed me into the Meraka family. Special thanks are addressed to Quentin WILLIAMS and Hina PATEL for their constant support and for not giving up on me throughout my research that took longer than initially planned.

I am grateful to my parents, parents in law and siblings for their patience, love and prayers. Special thanks and congratulations go to my wife for her unfailing patience, love and daily words of encouragement, and to my two sons, I thank you for being with your mother the joy of my life and the great impetus that made the completion of my thesis possible. I am also grateful to all my friends and colleagues for their prayers and encouraging remarks.

I would like to acknowledge CSIR Meraka Institute for offering me a studentship programme and taking full responsibility to defray all my study fees. The Tshwane University of Technology, the University of Paris-Est are greatly appreciated for giving me the opportunity to pursue a PhD degree.

Finally, I am indebted to my Heavenly Father, for granting me abundant grace without measure and being faithful all my life.

Thierry LUHANDJULA

July 2012

Pretoria, South Africa

Abstract

In this thesis, a methodological and algorithmic approach is proposed, for visual intention recognition based on the rotation and the vertical motion of the head and the hand. The context for which this solution is intended is that of people with disabilities whose mobility is made possible by a wheelchair. The proposed system is an interesting alternative to classical interfaces such as joysticks and pneumatic switches. The video sequence comprising 10 frames is processed using different methods leading to the construction of what is referred to in this thesis as an “intention curve”. A decision rule is proposed to subsequently classify each intention curve.

For recognition based on head motions, a symmetry-based approach is proposed to estimate the direction intent indicated by a rotation and a Principal Component Analysis (PCA) is used to classify speed variation intents of the wheelchair indicated by a vertical motion. For recognition of the desired direction based on the rotation of the hand, an approach utilizing both a vertical symmetry-based approach and a machine learning algorithm (a neural network, a support vector machine or *k*-means clustering) results in a set of two intention curves subsequently used to detect the direction intent. Another approach based on the template matching of the finger region is also proposed. For recognition of the desired speed variation based on the vertical motion of the hand, two approaches are proposed. The first is also based on the template matching of the finger region, and the second is based on a mask in the shape of an ellipse used to estimate the vertical position of the hand.

The results obtained display good performance in terms of classification both for single pose in each frame and for intention curves. The proposed visual intention recognition approach yields in the majority of cases a better recognition rate than

most of the methods proposed in the literature. Moreover, this study shows that the head and the hand in rotation and in vertical motion are viable intent indicators.

Résumé

Dans cette thèse, nous proposons une approche méthodologique et algorithmique pour la reconnaissance visuelle d'intentions, basée sur la rotation et le mouvement vertical de la tête et de la main. Le contexte dans lequel cette solution s'inscrit est celui d'une personne handicapée, dont la mobilité est assurée par un fauteuil roulant. Le système proposé constitue une alternative intéressante aux interfaces classiques de type manette, boutons pneumatiques, etc. La séquence vidéo, composée de 10 images, est traitée en utilisant différentes méthodes pour construire ce qui dans cette thèse est désigné par « courbe d'intention ». Une base de règles est également proposée pour classifier chaque courbe d'intention.

Pour la reconnaissance basée sur les mouvements de la tête, une approche utilisant la symétrie du visage est proposée pour estimer la direction désirée à partir de la rotation de la tête. Une Analyse en Composantes Principales (ACP) est utilisée pour détecter l'intention de varier la vitesse de déplacement du fauteuil roulant, à partir du mouvement vertical de la tête. Pour la reconnaissance de la direction basée sur la rotation de la main, une approche utilisant à la fois la symétrie verticale de la main et un algorithme d'apprentissage (réseaux neuronaux, machines à vecteurs supports ou *k*-means), permet d'obtenir les courbes d'intentions exploitées par la suite pour la détection de la direction désirée. Une autre approche, s'appuyant sur l'appariement de gabarits de la région contenant les doigts, est également proposée. Pour la reconnaissance de la vitesse variable basée sur le mouvement vertical de la main, deux approches sont proposées. La première utilise également l'appariement de gabarits de la région contenant les doigts, et la deuxième se base sur un masque en forme d'ellipse, pour déterminer la position verticale de la main.

Les résultats obtenus montrent de bonnes performances en termes de classification aussi bien des positions individuelles dans chaque image, que des courbes d'intentions. L’approche de reconnaissance visuelle d’intentions proposée produit dans la très grande majorité des cas un meilleur taux de reconnaissance que la plupart des méthodes proposées dans la littérature. Par ailleurs, cette étude montre également que la tête et la main en rotation et en mouvement vertical constituent des indicateurs d'intention appropriés.

Table of Content

List of Figures

List of Tables

Glossary

HCI	Human Computer Interaction
CCD	Charged Couple Device (camera)
PCA	Principal Component Analysis
ADABOOST	Adaptive Boosting
CAMSHIFT	Continuously Adaptive Mean Shift
MLP	Multi Layer Perceptron (Neural Network)
SVM	Support Vector Machines
HOG	Histogram of Oriented Gradient
MMHCI	Multimodal Human Computer Interaction
CFG	Context Free Grammar
HMM	Hidden Markov Models
SST	Spatio-Spectral Tracking
AUTOS	Automated Understanding of Task and Operator State
AHMM	Abstract Hidden Markov Models
ANFIS	Adaptive Neuro-Fuzzy Inference System
BP	Back-Propagation
LMS	Least Mean Square
PHATT	Probabilistic Hostile Agent Task Tracker
DNF	Disjunctive Normal Form strategy
RFID	Radio Frequency Identification
HaWCoS	Hands-free Wheelchair Control System

VFH	Vector Field Histogram
IW	Intelligent Wheelchair
OMNI	Office wheelchair with high Manoeuvrability and Navigational Intelligence
VAHM	Véhicule Autonome pour Handicapé Moteur.
SIAMO	Integral System for Assisted Mobility (in Spanish)
LRF	Laser Range Finders
AAM	Active Appearance Model
LDA	Linear Discriminant Analysis
LGBP	Local Gabor Binary Patterns
RBF	Radial Basis Function
SVR	Support Vector Regression
KPCA	Kernel Principal Component Analysis
DOF	Degree Of freedom
FORMS	Flexible Object Recognition and Modelling System
IBL	Instance-Based Learning
ASL	American Sign Language
PDP	Parallel Distributed Processing
TSL	Taiwanese Sign Language
COG	Centre of Gravity
SIFT	Scale-Invariant Feature Transform descriptors

Chapter 1

Introduction

One of the main functions of an enabled environment is to provide a setting where people with disabilities and the aged can function independently, be active, and contribute to society. One of the challenges facing the task of realising such an environment is to develop systems that can assist them in performing the tasks they wish to carry out without other people's assistance.

Formally defined, an intention is a psychological concept, commonly understood as the determination to act in a certain manner [3]. Intention recognition, also known as plan recognition, refers to the problem of inferring a person's intentions from observations of that person's behaviour. Good performance in a team environment is heavily conditioned by awareness of the intentions of people within society [204]. As a result, it can be said that human machine interaction (HMI) where the machine plays a support role requires that the intention of the user is well understood by the machine. This intention recognition capability is central to the multidisciplinary area of HMI and for the more specific area of enabled environment.

There are many contexts in which intention recognition finds applications, and a common one is that of a person with a physical disability whose mobility is enabled by the use of a powered wheelchair. In many of these physical disabilities (tetraplegia, upper and lower limb disabilities, etc.), the motion of the head usually remains intact, and in other cases (lower limb disabilities) even the hand motion is still available.

This renders the head and the hand suitable intent indicators for the motion of a powered wheelchair. Such a wheelchair, whose mobility is made possible through an intent recognition solution that is easier to use than traditional means or in other cases that shares control of the motion with the user, becomes a robotic wheelchair.

1.1 Problem Statement

This thesis proposes a vision-based solution for intention recognition of a person from the motions of the head and the hand.

This solution is intended to be applied in the context of wheelchair bound individuals whose intentions of interest are the wheelchair's direction and speed variation indicated by a rotation and a vertical motion respectively. Both head-based and hand-based solutions are proposed as an alternative to solutions using joysticks, pneumatic switches, etc.

The data used are video sequences of 576×768 image frames captured from a Charge-Coupled Device (CCD) camera (Hi-Resolution Dome Camera - 1/3" CCD, 470 TV lines, 0.8 lux, 3.6mm (F2.0) Lens) and a "25 frames per second" E-PICOLO-PRO-2 frame grabber.

As illustrated in Figure 1-1 the input to the solution is the head/hand motion of a subject accessed using the camera and the output is an inferred intention aimed to become the command for the wheelchair to move in a certain direction or to vary its speed. The subject's head/hand motion used as input is contained in a video sequence of 10 frames and the intention detection task consists of mapping these 10 frames to, as referred to in this work, an intention curve.

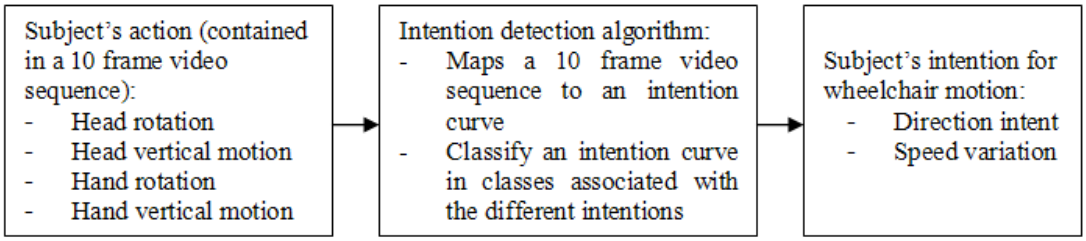

Figure 1-1: Intention detection system

For head rotation, a symmetry-based approach that maps a face image to a symmetry curve is adopted. From this symmetry-based approach four different methods are proposed according to the feature selection process (centre of gravity (COG) of the symmetry curve and *y*-intercept of the line approximating the symmetry curve) and the decision rule (based on the difference of means and the statistics in a Gaussian distribution) used for pose and intent recognition. For vertical motion of the head, Principal Component Analysis (PCA) is used for pose and intent recognition. A method proposed by Jia and Hu [66], [67] is also implemented for comparison. The approach uses adaboost for face detection and profile pose estimation, camshift for tracking, and nose template matching for vertical pose detection. These methods for head-based rotation and vertical motion detection are used to map the given 10-frame video sequences as input, to 10-point intention curves that are subsequently classified using appropriate decision rules.

For hand rotation, a variant of the symmetry-based approach used for the head is proposed where the symmetry curve is calculated vertically rather than horizontally. The statistics (mean and standard deviation) of the symmetry curves are used as two-dimensional (2D) data features and three different machine learning methods, two supervised (a neural network and a support vector machine) and one unsupervised (*k*-means clustering) are used for single pose classification. For intent recognition a

decision rule that makes use of two different intention curves is used – one comprising the output of the single pose classification step resulting from the machine learning approaches and the other containing the means of the vertically computed symmetry curves. Another method based on a normalised cross-correlation template matching is proposed. For the vertical motion of the hand, the geometric constraints on the hand's contour are considered, leading to the use of a mask in the form of an ellipse to determine the hand's vertical position. The other proposed approach is based on a normalised cross-correlation template matching. For comparison of the proposed methods for vertical motion of the hand, a feature selection found in the literature known as the Histogram of Oriented Gradient (HOG) is implemented. These methods for hand-based rotation and vertical motion detection are used to map the 10-frame video sequences to 10-point intention curves that are classified using appropriate decision rules.

1.2 Motivation and objectives

The motivation behind any solution aimed at an enabling environment is to "enable" people with disabilities and the aged to be more independent and furthermore to contribute to society. The solution proposed in this thesis is a contribution to the task of realising such an environment by providing an intent recognition algorithm intended to be applied in a robotic wheelchair application.

1.3 Sub-problems

The solution proposed in this thesis can be divided into two, namely a head-based solution and a hand-based one both aimed at the same intent indication (direction and speed variation intents). Both solutions recognise two different kinds of motion: rotation and vertical motion. The input video sequence is made of image frames with the head or the hand as the object of interest. A pre-processing step aimed at segmenting the head and the hand from each frame is performed before intent recognition. Note that for detection and tracking, the frontal view of the head (the face) is of interest as well as the dorsal view (as opposed to the palm) of the hand. Below is an enumeration of the sub-problems addressed in this thesis.

Sub-problem 1: Face detection and tracking

Determine the exact location of the face in the input image frame that will be used for further processing to achieve intent recognition.

Sub-problem 2: Head rotation recognition

Determine whether the head remains centred, moves to the right or to the left indicating the direction intent.

Sub-problem 3: Recognition of the head in vertical motion

Determine whether the head remains centred, moves up or down indicating the speed variation intent.

Sub-problem 4: Hand detection and tracking

Determine the exact location of the hand in the input image frame that will be used for further processing to achieve intent recognition.

Sub-problem 5: Hand rotation recognition

Determine whether the hand remains centred, moves to the right or to the left indicating the direction intent.

Sub-problem 6: Recognition of the hand in vertical motion

Determine whether the hand remains centred, moves up or down indicating the speed variation intent.

1.4 Assumptions

Assumption 1: A face viewed from the front is symmetric and presents separable patterns for the three different positions: centre, right and left.

Assumption 2: The disabilities targeted for the proposed solution are those where the head and/or the hand are still moving properly.

Assumption 3: The camera used to capture the motions of the head and the hand is assumed to be incorporated into the structure of the wheelchair next to these intent indicators rendering them close enough to be the only skin colour object within the field of view exempting occlusion problems.

Assumption 4: The camera used to capture the motions of the head and the hand are assumed to be incorporated on the wheelchair and therefore at a fixed distance from these indicators exempting any scaling considerations.

Assumption 5: For hand motion recognition, the hand is treated as a rigid object performing two types of motion: rotation and vertical motion.

1.5 Scope

Though intended for a wheelchair application, the algorithm has not been tested on actual people with disabilities. This work is limited to the implementation of intent recognition algorithms using recorded video sequences of subjects sitting on an office chair (to mimic a person with a physical disability sitting on a wheelchair) and performing the four types of motion of interest in this thesis.

Two motions of the head and the hand are defined as intent indicators, namely, rotation and vertical motion. For the head, rotation means the motion with respect to the vertical axis through the centre of the face and vertical motion means the motion with respect to the horizontal axis through the nose of the face. For the hand however, rotation consists of a motion relative to the horizontal axis parallel to the arm, and vertical motion consists of a motion relative to the vertical axis through the joint articulation linking the hand and the arm (the wrist).

The hand and the head are independent indicators for the same type of motions. No data fusion scheme is used to combine these two motion indicators.

1.6 Contribution

- An alternative visual solution for head and hand motion detection aimed at intent recognition, intended to be applied to assistive living is proposed. Its performance as Chapter 5 reveals is good when compared to those chosen from the literature and implemented in this work. This is an important

contribution because as shown in the literature, one of the most promising sensor technologies associated with assistive living application is machine vision and thus successful implementation of visual solutions is increasingly preferred.

- For head rotation: A symmetry-based approach is used for pose estimation and combined with a decision rule for direction intent recognition. This thesis therefore shows how the symmetry property of the head can be used for motion understanding. The other merit of the use of this symmetry-based approach is its simplicity as opposed to head pose estimation found in the literature that require sophisticated machine learning algorithm for recognition.
- For head in vertical motion: A decision rule is implemented for speed variation intent recognition using intention curves obtained from the PCA-based pose classification.
- For hand rotation: A variant of the symmetry-based approach (the symmetry is calculated vertically rather than horizontally) is combined with machine learning algorithms (Neural Networks, Support Vector Machines (SVM) and *k*-means clustering) for pose estimation and combined with a decision rule for direction intent recognition. An additional proposed approach is based on a normalised cross-correlation template matching resulting in the intention curve that is classified using an appropriate decision rule.
- For hand in vertical motion: An ellipse shaped mask is implemented for pose estimation and intention curves generation. An additional approach is based on a normalised cross-correlation template matching. A HOG descriptor is also

adapted for the same purpose. An appropriate decision rule is subsequently proposed to classify these intention curves.

- As the literature reveals, the solutions of head pose estimation and hand gesture recognition are used in many applications including wheelchair motion, allow symbolic commands based on the head or the hand posture. The solution proposed in this thesis on the other hand, recognises intents by classifying the motion contained in a specific number of frames (10 in this work) rather than the posture in a single frame. This contribution brings the advantage that even if the position of the head and the hand is only loosely detectable, that is, the exact pose cannot be quantified to determine which pose is left, right, up, down or centre and to which extent they are in these poses; the different kinds of motion can still be robustly detected. The other advantage is that the misdetection of a single frame is less costly on the overall performance.
- Gesture recognition solutions found in the literature are made possible looking at a change in the hand's contour shape and it is typically applied to sign language applications. The literature doesn't contain gesture recognition solutions where the motion of the hand is a micro-operation such as the rotation and vertical motion described in this thesis, for which the approaches found in the literature are typically invariant or unusable for robust classification.

1.7 Outline

The remainder of the thesis consists of the five following chapters:

Chapter 2

Chapter 2 presents some of the literature found in four areas of research relevant to the work presented in this thesis:

- Intention detection
- Robotics wheelchairs
- Head pose estimation
- Hand gesture recognition

Chapter 3

Chapter 3 describes the algorithms proposed for head-based pose estimation and intent recognition. For rotation, a symmetry-based approach is used to implement four different approaches: Two using the centre of gravity of the resulting symmetry curve and a decision rule-based on the difference of means and the statistics (means and standard deviation) in a Gaussian distribution. The two other approaches use the same decision rules on *y*-intercept of the line approximating the symmetry curve. For vertical motion, PCA and a decision rule are employed. For comparison, a method based on adaboost, camshift and template matching proposed by Jia and Hu [66], [67] is implemented.

Chapter 4

Chapter 4 describes the algorithms proposed for hand-based pose estimation and intent recognition. For rotation a vertical symmetry-based approach is employed in combination with three different methods based on three different machine learning algorithms (Neural Network, Support Vector Machines and *k*-means) and combined

with a decision rule. For vertical motion two methods are proposed, one based on a normalized cross-correlation template matching and the other, on an ellipse shaped mask. For comparison, a HOG descriptor proposed in the literature is also implemented and compared to the proposed method.

Chapter 5

Chapter 5 furnishes the experimental results of the proposed methods. Three sets of results are reported for each proposed method: the first set portrays the performance for single frame pose classification, the second set illustrates the performance for intent recognition through the classification of intention curves and the third set of results indicates the performance of each method when fewer frames are used for recognition within a 10-frame video sequence.

Chapter 6

Chapter 6 furnishes a summary of the work proposed in this thesis, some concluding remarks, and some suggestions for future work.

Chapter 2

Literature Survey

2.1 Introduction

The area of Human Computer Interaction (HCI) is essential to enable efficient and effortless communication between humans and computers [1]. It spans through many areas of research such as psychology, artificial intelligence and computer vision [2], and through four categories of techniques, namely manual, speech, tele-operation and vision [1]. An important trend in recent work on HCI is to consider it as a kind of collaboration [4] where the computer or machine's aim is to increase the performance of the human user by providing assistance [5], or to perform a task that the user can not carry out on his / her own. Three areas of investigation are of interest in HCI:

- The understanding of the user who interacts with the computer.
- The understanding of the system (the computer technology and its usability).
- The understanding of the interaction between the user and the system.

A more wide-ranging variant of HCI is the Multimodal Human-Computer Interaction MMHCI, which similarly to HCI is a multidisciplinary area lying at the crossroads of several research areas where psychology and cognitive science are needed to understand the user's perceptual, cognitive, and problem solving skills. Sociology is used to understand the wider context of interaction, ergonomics provides an understanding of the user's physical capabilities, graphic design is required to produce

effective interface presentation and computer science and engineering are used to build the necessary technology [6]. Unlike in traditional HCI applications, however, typically consisting of a single user facing a computer and interacting with it via a medium such as a mouse or a keyboard, in MMHCI applications, interactions do not always consist of explicit commands, and often involve multiple users (e.g., intelligent homes, remote collaboration, arts, etc.). This was made possible by the remarkable progress in the last few years in computer processor speed, memory, and storage capabilities, matched by the availability of many new input and output devices such as phones, embedded systems, laptops, wall size displays, and many others. This wide range of computing devices being available, with differing computational power and input/output capabilities, enables new ways of interaction through visual methods that include large-scale body movements, gestures and head pose, eye blinks or gaze [7], and other methods such us speech, haptics and glove mounted devices.

Vision-based HCI interfaces usually focus on head tracking [8], face and facial expression recognition, eye tracking, gaze analysis, gesture recognition, human motion analysis and lower arm movement detection, where the recognition methods are classified using a human-centred approach using one of the following indicators:

- Large-scale body movements
- Hand gestures
- Gaze.

Large-scale body movement solutions result from articulated motion analysis where three important issues must be addressed:

- The representation: Joint angles or motion of all the sub-parts
- The computational paradigms: They can be deterministic or probabilistic
- Computation reduction

A Previous work describes certain methods that use geometric primitives to model different components and others that use feature representations based on appearance (appearance-based methods). In the first approach, external markers are often used to estimate body posture and relevant parameters. While markers can be accurate, they place restrictions on clothing and require calibration. They are therefore not desirable in many applications. Moreover, the attempt to fit geometric shapes to body parts can be computationally expensive and these methods are often not suitable for real-time processing. Appearance-based methods, on the other hand, do not require markers, but require training [2].

Gesture recognition (which refers exclusively to hand gesture recognition within the computer vision community) plays an essential role in HCI and MMHCI to remote collaboration applications. Most of the gesture-based HCI systems allow only symbolic commands based on hand posture or 3D pointing. This is due to the complexity associated with gesture analysis and the desire to build real-time interfaces. Also, most of the systems accommodate only single-hand gestures. Gaze detection systems essentially consisting of an eye tracking solution can be grouped into wearable or non-wearable, and infrared-based or appearance-based solutions. Infrared systems are more accurate than those that are appearance-based; however, there are concerns over the safety of prolonged exposure to infrared lights. Appearance-based systems usually capture both eyes using two cameras to predict gaze direction. Due to the computational cost of processing two streams simultaneously, the resolution of the image of each eye is often small making such systems less accurate. As an alternative, the use of a single high-resolution image of one eye is proposed to improve accuracy. On the other hand, infrared-based systems usually use only one camera, although the use of two cameras is proposed to further

increase accuracy. Wearable eye trackers have also been investigated mostly for desktop applications. The main issues in developing gaze tracking systems are intrusiveness, speed, robustness, and accuracy. Gaze analysis can be performed at three different levels [2]:

- Highly detailed low-level micro-events
- Low-level intentional events
- Coarse-level goal-based events

In general, vision-based human motion analysis systems used for HCI and MMHCI can be thought of as having mainly four stages:

- Motion segmentation
- Object classification
- Tracking
- Interpretation

The literature also makes a distinction between command (actions can be used to explicitly execute commands: select menus, etc.) and non-command interfaces (actions or events used to indirectly tune the system to the user's needs) [9].

Human-Computer collaboration provides a practical and useful application for plan recognition techniques. Plan recognition also known as intention detection is a central component in many applications among which assistant systems for elders [5]. Intent recognition solutions are very useful for mobile robots (among which are robotic wheelchairs) as they can help human users on a variety of tasks, such as, material handling, transport, surveillance, demining, assistance to people with disabilities and housekeeping, provided that it understands the intent of the user [10].

The solution proposed in this thesis has some of the characteristics of an HCI solution as it is vision-based, non-intrusive, human centred, and it involves the four stage mentioned earlier namely segmentation, classification, tracking and interpretation or recognition. The rest of this chapter presents some of the solutions found in the literature for areas such as intention detection, robotic wheelchairs, head pose estimation, hand gesture and pose recognition as they are relevant in addressing the problem at hand.

2.2 Intention detection

Schmidt *et al.* [11] first identify the problem of plan recognition also known in some cases as intention detection or intent recognition. Since then it is applied to a diversity of areas, including natural language understanding and generation [12], [13], dynamic traffic monitoring [14], story understanding [11], [15], [16], adventure game [17], network intrusion detection [18], multi-agent coordination [19] and multi-agent team monitoring [20]. Kautz and Allen [21] present the first formal theory of plan recognition where they define it as “identifying a minimal set of top-level actions sufficient to explain the observed actions, and use minimal covering set as a principle for disambiguation”.

Research in plan recognition has taken several different directions, the most popular being the development of logic theories to provide algebras through which to reason about plans from observed agent actions. In [22], the authors introduce the theoretical concept of plan knowledge graphs, along with a new formalism, to simplify the process of plan recognition. In [23], the authors interpret the assignment of intentions to a sequence of incoming behaviours or activities indicated by body

trajectories as a pattern recognition problem. The solution proposed is a formalism known as Context Free Grammar (CFG). In [24], three distinct behaviours of a rat namely the walking behaviours of exploratory locomotion (EL), the grooming (GR) and behavioural stillness (BS) are recognised using a visual approach based on a supervised neural network, demonstrating the feasibility for automated machine learning of behaviour at some level. In [25] Pynadath and Wellman propose a probabilistic method based on parsing. Their approach employs probabilistic state-dependent grammars (PSDGs) to represent an agent's plan. The PSDG representation, together with inference algorithms supports efficient answering of restricted plan recognition queries. The work in [26] addresses the problem of inferring high-level intentions from a global positioning system (GPS) using Bayesian networks to predict the position and velocity of a traveller in an urban setting, using auto, bus and foot travel as the means of locomotion. The vision-based solution described in [27] also makes use of Bayesian networks and model-based object recognition to identify multi-person actions in the real world indicated by large-scale body movements.

In [28] a vision-based technique is presented for interpreting the near-term intention of an agent performing a task in real-time by inferring the behavioural context of the observed agent defined as the path followed by a vehicle in a military application (a tank). A hierarchical, template-based reasoning technique is used as the basis for intention recognition, where there is a one-to-one correspondence between templates and behavioural contexts or sub-contexts. In this approach, the total weight associated with each template is critical to the correct selection of a template that identifies the agent's current intention. A template's total weight is based on the contributions of individual weighted attributes describing the agent's state and its surrounding environment. The work described develops and implements a novel

means of learning these weight assignments by observing actual human performance. It accomplishes this by using back-propagation neural networks and fuzzy sets. In [29], two mathematical methods are proposed for creating a model to characterise a non-rigid motion and its dynamics. The work is based on the observation that every activity has an associated structure characterised by a non-rigid shape. In one method the activity is modelled using the polygonal shape formed by joining the locations of these point masses at any time, using Kendall's statistical shape theory. A nonlinear dynamic model is used to characterise the variations in the 2D or 3D shapes being observed. The second method consists of modelling the trajectories of each moving object in 3D space. In [30], a system is proposed with a perception level made of a sensor fusion system. The system processes the sensing data first, and then gets the physical information about the environment, including the large-scale body movements of humans. The recognition level is a translator from the crisp data processed at perception level to the qualitative expression that contains vague time scales by means of fuzzy logic. The intention inference level has groups of fuzzy rules using qualitative expression to infer the human's intention for the simple specific cooperative task.

To deal with uncertainty inherent in plan inference, Charniak and Goldman [15], [16] built the first probabilistic model of plan recognition based on Bayesian reasoning. Their visual system supports automatic generation of a belief network from observed human actions indicated by his motion trajectory according to some network construction rules. The constructed belief network is then used for actions understanding. As a powerful tool for time series prediction problems, many solutions make use of Hidden Markov Models (HMM) given the temporal nature of human actions. In [31], a visual system is proposed for intent recognition that is robust to

illumination changes, background clutter, and occlusion. The system uses a Spatio-Spectral Tracking module (SST) to determine human motion trajectory and track them in the video sequence where the observer robot is assumed to be static. The tracking module is composed of three components:

- Appearance modelling
- Correspondence matching
- Model update

The activity modelling approach uses HMMs in a Bayesian framework that uses context to improve the system's performance. In [4], The AUTOS (Automated Understanding of Task and Operator State) model is proposed for team intent inference, where the activities of each team member as well as those of the team overall indicated by their motion trajectories, are observed. The underlying principles stem from the information-on-need paradigm that is viewed as being vital to contemporary C2 operations. AUTOS accepts as input speech and text, and calls for interfaces that can track progress through tasks and can facilitate those tasks, aiding operators without interrupting their work. Three components collectively make up an AUTOS system:

- Direct observation mechanisms
- Indirect observation mechanisms
- Task models

In [32], a general principle of understanding intentions is proposed, which states that people have a mechanism for representing, predicting and interpreting each other's actions. Using a novel formulation of HMMs, the proposed visual solution models the interactions of several people with the world indicating their intent to performing various activities: following, meeting, passing by, picking up an object, and dropping

off an object. The distinguishing feature in the HMMs is that they model not only transitions between discrete states, but also the way in which parameters encoding the goals of an activity change during its performance. This novel formulation of the HMM representation allows for recognition of the agent's intent well before the underlying actions indicated by the angle and the distance of each agent in the scene, are finalised. In [33], an HMM is used as a representation of simple events indicated by the shape of a hand which are recognised by computing the probability that the model produces the visual observation sequence. This solution is applied to sign language recognition. Visual systems based on parameterised-HMM [34] and coupled-HMM [35] are introduced in order to recognise more complex events such as interactions of two mobile objects by observing their large scale movements. In [36], a vision-based stochastic context-free grammar parsing algorithm is used to compute the probability of a temporally consistent sequence of primitive actions recognised by HMMs. The actions of interest include bending over and entering a secure area. More recently, Bui *et al*. [37], [38], propose an online probabilistic policy recognition method for the recognition of group behaviour, based on the Abstract Hidden Markov Model (AHMM) and the extension of AHMM allowing for policies with memories. In their frameworks, scalability in policy recognition in the models is achieved using an approximate inference scheme called the Rao-Blackwellised Particle Filter. In [39], the intention of interest is the change in mode of transportation (e.g., walk, driving a car, get on a bus, etc.) and a Hierarchical Markov Model and particle filtering are used to predict a user's changes through spatial information and body motion.

It must also be noted that following the earlier definition of plan recognition, most systems infer a hypothesised plan based on observed actions. Therefore automatic human activity recognition usually constitutes the processing component for intent

recognition [29]. In computer vision, it involves detecting and tracking mobile objects from a video sequence, after which the activities are recognised from the characteristics of these tracked objects. The interesting task is therefore to map these tracked object characteristics to a specific activity description. This leads to a range of approaches, which interprets this task as a matching process between a sequence of image features to a set of activity models. The best matched models are then selected based on some criteria and their matching degree. The differences among these approaches are:

- Whether image features are computed automatically and independently of input image sequences
- Whether the activity representation is generic and expressive enough to model a variety of activities but yet powerful enough to discriminate between similar activities (e.g. sitting and squatting)
- Whether the matching is performed optimally

The authors in [40] propose a human behaviour detection and activity support in a vivid room environment. The behaviour detection in the vivid room is performed using the ID4-based learning algorithm that builds decision trees incrementally, and three kinds of sensors embedded in the room namely:

- Magnet sensors: for doors/drawers
- Micro-switches: for chairs
- ID-tags: for humans

The information from these sensors is collected by a sensor server via RF-tag system and LAN. The human activity support system takes into account the human behaviours in the room using sound and voice.

Other vision-based works include solving the problem of capture intention in an indoor environment of camcorder users for home video content analysis [41], [42], where visual and temporal features are used on a support vector machines. The visual solution in [43] tracks a person's region of interest by both recovering the 3D trajectory in the indoor environment and estimating the head pose indicating the attention direction. First, a nonlinear graph embedding method is used to robustly estimate the head yaw angle under 0^o~360^o in low resolution images. Second, the person's trajectory is recovered in an affinely-equal manner with uncalibrated cameras. In [44], a non-visual approach is presented where sEMG signals are used as an indication of hand movements for hand prosthesis control. The approach is based on the Adaptive Neuro-Fuzzy Inference System (ANFIS) integrated with a real-time learning scheme to identify hand motion commands. The fuzzy system is trained by a hybrid method consisting of Back-Propagation (BP) and Least Mean Square (LMS). In [45], a boosting-based approach is used for classification of a driver's lane change intent through a computational framework referred to as "mind-tracking architecture". The system simulates a set of possible driver intentions and their resulting behaviours using an approximation of a rigorous and validated model of driver behaviour. The recognition of the plans of the elderly in relatively unconstrained environments is achieved in [46] using a plan/intent recognition framework based on a probabilistic model known as PHATT (Probabilistic Hostile Agent Task Tracker), where the trajectory of the people in the scene is the intent indicator. Some vision-based probabilistic approaches use the Dempster-Shafer (D-S) theory [13], [47], [48] to recognise the preferences of the person in the field of view, that is his/her repeated behavioural pattern indicated by large-scale body movements. In [13], the author

relies on this evidential reasoning to support rational default inference about the user's plan. D-S theory is also used to represent user preferences (i.e., user's repeated behavioural patterns) and facilitate the selection of competing hypotheses. Weiss *et al.* [49] show how to generalise traditional discrete decision trees used for classification to regression trees used for functional estimation. This visual approach is used for two-dimensional gesture recognition applied to the monitoring of a continuous movement stream of a pointing device such as a computer mouse. Like decision trees, regression trees perform partitioning based on a Disjunctive Normal Form (DNF) strategy, which has the advantages of clarity of knowledge organisation and traceability to features.

As surveyed in this section, many methods that are used for plan recognition and intention detection are proposed in the literature. They include grammar parsing, Kalman filters, linear models, supervised neural network, fuzzy logic, decision tree, Bayesian networks, HMM, parameterised-HMM, coupled-HMM, AHMM combined with the Rao-Blackwellised particle filter, a template-based reasoning technique, Kendall's statistical shape theory, spatio-spectral tracking. They vary in the way in which intentions or plans are defined as they usually relate to observed actions such as large-scale body movements, trajectory, spatial position, speech, handwriting, hand gestures, and even American Sign Language (ASL). Some of the application areas include pedestrian and transportation safety [50], surveillance [51], crime prevention, HCI, and even interpreting sign language. Many sensors can be used, including GPS, Radio Frequency Identification (RFID) tags, digital cameras, ultrasound sensors, infrared sensors, light sensors, physiological sensors, accelerometers, and motion sensors [52].

This thesis addresses the problem of detecting a micro-operation of a user involving the motion of the head and the hand rather than explicit and well defined action. Alternative vision-based approaches are proposed for such a purpose, which recognise motions of the head and the hand of a subject in a well defined setting. They make use principally of a symmetry-based approach, a normalised cross-correlation template matching and machine learning algorithms such as principal component analysis (PCA), neural networks, support vector machines and *k*-means clustering. The application of interest in this work is an interaction between a human and a robotic wheelchair. The next section discusses some robotic and powered wheelchair solutions.

2.3 Robotic wheelchairs

Mobile robots can help humans with a variety of tasks, such as, material handling, transport, surveillance, demining, assisting people with disabilities and housekeeping. Mobile robot architecture can be classified according to the relationship between sensing, planning and acting components inside the architecture. There are therefore three types of architecture:

- Deliberative architecture
- Reactive architecture
- Hybrid (deliberative/reactive) architecture

In deliberative architectures, there is a planning step between sensing and acting. When we compare deliberative architectures with reactive architectures we observe that deliberative architectures work in a more predictable way, have a high

dependency of a precise and complete model of the world, and can generate optimised trajectories for the robot. On the other hand, reactive architectures have a faster response to dynamic changes in the environment; they can work without a model of the world and are computationally much simpler [10].

One important subset of mobile robots can be found in the context of a person with a physical disability whose mobility is enabled by a wheelchair. Various tools that increase the mobility of the physically impaired, such as (powered) wheelchairs, walkers, or robotic manipulators, are commercially available. Many people who suffer from chronic mobility impairments, such as spinal cord injuries or multiple sclerosis, use powered wheelchairs to move around their environment [53]. Unfortunately, many of the common every-day-life manoeuvres such as docking at a table or driving through a door are experienced as difficult, time-consuming or annoying. Severe accidents such as falling down stairs or ramps, collisions with other chairs or persons, and getting blocked in corridors or elevators regularly occur. For these reasons, several existing mobility tools were equipped with sensors and a computerised controller to aid the physically impaired with everyday-life manoeuvring. Not only in wheelchairs [54], [55], but also in walkers [56], robotic guide canes for the visually impaired [57], and robotic manipulators [58].

Traditionally, powered wheelchairs have been driven with a joystick, which has proven to be an intuitive solution. Unfortunately to drive both efficiently and safely requires the user to have steady hand-control and good reactions, which can be impeded due to a variety of physical, perceptive or cognitive impairments [59] and by factors such as fatigue, degeneration of their condition and sensory impairments. Consequently, alternative methods of interaction are being investigated. Work has been carried out in the fields of speech, vision (gesture and gaze-direction

recognition) and brain-actuated control and the most popular set of approaches are vision-based [60], [61], [62].

A method for wheelchair obstacle avoidance is presented in [63] using the Canesta 3D time-of-flight infrared laser range sensor. The collision avoidance solution is presented for powered wheelchairs used by people with cognitive disabilities (such as Alzheimer's disease and dementia) therefore increasing their mobility and feeling of independence. An integration of the sensor system with global mapping and localisation methods as well as control methods using partially observable Markov decision processes is performed. In [64], a system called Hands-free Wheelchair Control System (HaWCoS), which allows people with severe disabilities to reliably navigate an electrical wheelchair without the need to use the hands, is presented. The system monitors a specific bio-signal which is the time series of a certain bodily function of the user (such as brain-waves, muscular activity, or eye posture) and reacts appropriately to the particular detected pattern in the monitored signal. The detection of intentional muscle contractions involves a piezo-based sensor, which is almost insensitive to external electro-magnetic interference.

As stated earlier perhaps the most promising sensor technology (among ultrasonic acoustic range finder (i.e., sonar), infrared (IR) range finder, laser range finders (LRFs), laser striper, etc.) associated with these robotic wheelchairs is machine vision. Cameras are much smaller than LRFs and, thus, much easier to mount in multiple locations on a wheelchair, they can also provide much greater sensor coverage, the cost of machine vision hardware has fallen significantly, and machine vision software continues to improve. Thus successful implementation of a robotic wheelchair based on computer vision is increasingly preferred. There are smart and robotic wheelchairs in the literature [55] that use computer vision for landmark

detection (e.g., Rolland, MAid, Computer-Controlled Power Wheelchair Navigation System) where the visual indicators are the head and eyes.

Among the vision-based solution found in the literature, the authors in [65] propose a Bayesian approach to robotic assistance for wheelchair driving, which can be adapted to a specific user. The proposed framework is able to model and estimate even complex manoeuvres, and explicitly takes the uncertainty of the user's intent into account. In [66] and [67], the authors propose an integrated approach to real-time detection, tracking and direction recognition of human faces, which is intended to be used as a human-robot interaction interface for a robotic wheelchair. Adaboost face detection is applied inside the comparatively small window, which is slightly larger than the camshift tracking window, so that the precise position, size and frontal, profile left or profile right direction of the face can be obtained rapidly. If the frontal face is detected, template matching is used to tell the nose position. In [68], [69] and [70] the authors present the NavChair assistive navigation system based on a modelling approach to monitoring human control behaviour in real-time. The NavChair takes advantage of the capabilities of both the user and the machine by allowing them to share the control of the system output. The Vector Field Histogram method, which is an effective sonar-based obstacle avoidance for mobile robots, is adapted for use in human-machine systems; the shortcomings of the wheelchair platform in this regard have been overcome. The sensory system comprises an array of 12 Polaroid ultrasonic transducers, a joystick and sonar sensors. In [71], the authors propose an intelligent wheelchair (IW) control system for people with various disabilities. The proposed system involves the use of face-inclination to determine the wheelchair direction and mouth-shape information to determine whether the wheelchair must proceed or stop. In the detector, the facial region is first obtained

using adaboost; thereafter the mouth region is detected based on edge information. The extracted features are sent to the recogniser, which recognises the face inclination and mouth shape using statistical analysis and *k*-means clustering respectively.

In [72], many other platforms that help people in their daily manoeuvring tasks are surveyed: OMNI, Bremen autonomous wheelchair, RobChair, Senario, Drive Assistant, VAHM, Tin man, Wheelesley (stereo-vision guided), and NavChair (sonar guided). These systems are based on "shared control" [73], [74], [75] where the control of the wheelchair or any other assistive device is shared with the user. Often the developed architectures consist of different algorithms that each realise specific assistance behaviour, such as "drive through door", "follow corridor" or "avoid collision". The presence of multiple operating modes creates the need to choose from them, and therefore makes the user responsible for selecting the appropriate mode, which in some instances might be an inconvenience. Several powered wheelchairs are available with modular architecture [76], of which the SIAMO project (Spanish acronym for Integral System for Assisted Mobility) is an example [77]. The goal of this modular architecture is to easily configure the wheelchair to suit the needs of a high variety of users with different disabilities. This modular architecture also makes it easy to adapt new functionalities to the wheelchair [78]. For severely disabled persons one way of controlling a wheelchair is by means of head movements. Currently, such devices do exist, such as those called head controlled joystick or head movement interface, mechanical, camera-based [79], accelerometer-based [80] and based on infrared light [81], where active components are attached to the head of the user.

This section shows how rich the field of electrically powered and robotic wheelchairs is, where the system sensoring component includes ultrasonic acoustic

range finder (i.e., sonar), infrared (IR) range finder, laser range finders (LRFs), laser striper, piezo-sensors for (bio-signal), and vision-based. Typical indicators for motion control include joysticks, gesture, gaze direction, head poses, brain signals, speech, etc. Of interest in this thesis are visual solutions focusing on the pose of the head and the free hand of the user where several alternative approaches are proposed for intent recognition. The next two sections present some of the solutions found in the literature for head pose estimation and hand gesture recognition.

2.4 Head pose estimation

In surveillance systems the knowledge of head poses provides an important cue for higher level behavioural analysis and the focus of an individual's attention often indicates their desired destination [60]. In addition to contributing to the task of robust face recognition for multi-view analysis which is still a difficult task under pose variation [82], pose estimation can also be considered as a sub problem of the general area of intention detection as it is useful for inference of nonverbal signals related to attention and intention. This makes head pose estimation solutions a key component for HCI [83]. Existing head pose estimation methods can be grouped into:

- Model-based methods (within which we also classify Active Appearance-based methods) [84]
- Appearance-based methods [85]
- Feature-based approaches [86], [87], [88], [89] within which we also classify appearance-based subspace methods.

Appearance-based techniques use the whole sub-image containing the face while model-based approaches use a geometric model.

2.4.1 Model-based solutions

Several works on head pose measurement in low resolution video involve the use of labelled training examples which are used to train various types of classifiers such as neural networks [90], [86], [91], support vector machines [92] or nearest neighbour and tree-based classifiers [93], [94], [95]. Other approaches model the head as an ellipsoid and either learn a texture from training data [96] or fit a re-projected head image to find a relative rotation [97]. In [98], a 2D ellipse is used to approximate the head position in the image. The head position is obtained using colour histogram or image gradients. However, light changes and different skin colours result in tracking failures. Another drawback with such an approach is the inability to report head orientation. In [99], partial orientation information, such as tilt or yaw is available. However, the accuracy of those systems is low (up to 15 degree error in estimating rotation).

Recently, model-based approaches like the bunch graph approach, PCA, Eigen faces and Active Appearance Models (AAMs), have received considerable interest. AAMs [100] are nonlinear parametric models derived from linear transformations of a shape model and an appearance model. A neural network can also be trained to distinguish between different persons or to make a distinction between poses of one person's face [101]. The system proposed in [102], uses neural networks on each camera view to estimate head orientation in either direction. For the fusion of the multiple views, a Bayesian filter is applied to both diffuse prior estimates (temporal propagation) as well as search for the most coherent match of overlapping single view hypotheses over all the included sensors.

Unsupervised approaches such us eigenfaces [103], [104] learn the subspace for recognition via the Principle Component Analysis (PCA) [105] of the face manifold, while supervised approaches like Fisherfaces [106] learn the metric for recognition from labelled data via the Linear Discriminant Analysis (LDA). Linear approaches in head pose estimation are found in [107], [108], [109]. It must be noted that the PCA/LDA approaches for head pose estimation are limited because of the non-linearity of the underlying manifold structure, and richness in local variations. In recent years, non-linear methods for high dimensional non-linear data modelling, Locally Linear Embedding (LLE) [110] and Graph Laplacian [111], perform very well in finding manifold structure through embedding a graph structure of the data derived from the affinity modelling. When the problem space is large, a kernel method [112], [113] is employed and in other cases where complexity is an issue, a piece-wise linear subspace/metric learning method [114] is developed to map out the global nonlinear structure for head pose estimation. Template matching is another popular method used to estimate head pose where the best template can be found via a nearest-neighbour algorithm and where the pose associated with this template is selected as the best pose. Advanced template matching can be performed using Gabor Wavelets and Principle Components Analysis (PCA) or Support Vector Machines, but these approaches tend to be sensitive to alignment and are dependent on the identity of the person [82].

More accurate systems use 3D geometrical models to represent the head as a rigid body. In [115], Yang and Zhang use a rough triangular mesh with semantic information. Stereo is used to obtain 3D information, which is matched against the known model. A major shortcoming of this method is the amount of time one needs to spend to create a precise model. Recent approaches use a cylinder to approximate

both the underlying head geometry and texture [116], [117]. Since a cylinder is only a rough approximation of head geometry, those methods suffer from inaccuracies in estimating rotation, and have difficulties differentiating between small rotations and translations. In [118], an approach for 3D head pose estimation from a monocular sequence is proposed. To estimate the head pose accurately and simply, an algorithm is used based on the geometry information of the individual face and projective model without the need of any 3D face model and any special markers on the user's face. Another 3D solution to head pose estimation is presented in [119] where the system relies on a novel 3D sensor that generates a dense range image of the scene. In [82], a novel discriminative feature is introduced which is efficient for pose estimation. The representation is based on the Local Gabor Binary Pattern (LGBP) and encodes the orientation information of the multi-view face images into an enhanced feature histogram. A Radial Basis Function (RBF) kernel Support Vector Machines (SVM) classifier is used to estimate poses. The aim of the work in [83] is to develop a new vision-based method which can estimate the 3D head pose with high accuracy with an adaptive control of diffusion factors in a motion model of a user's head used in particle filtering.

2.4.2 Appearance and feature-based techniques

Appearance-based approaches use filtering and image segmentation techniques to extract information from the image. Some typical appearance-based techniques include optical flow algorithms as well as edge detectors such as Gabor wavelets [120]. Filtering and segmentation resulting from appearance-based methods play a significant role in head pose estimation, but it must be noted that few head pose estimation algorithms are known to be exclusively appearance-based as they require

the step of recognising the pose. Among the exceptions are the Gabor head pose estimation described in [87] where the weights of a Gabor wavelet network directly represent the orientation of the face. Its disadvantage, however, is the computational effort involved, which is very user specific. Brolly *et al.* [121] used Nurbs surface with texture to synthesise both appearance and pose, but could not report pose accuracy since ground truth was unavailable. Several researchers [122], [123] introduced the notion of extended super quadric surface, or Fourier synthesised representation of a surface, which possesses a high degree of flexibility to encompass the face structure. They use model-induced optical flow to define pose error function. The usage of a parameterised surface enables them to resolve ambiguities caused by self occlusion.

The majority of feature-based algorithms use the eyes as features since they are easy to detect due to their prominent appearance. The nostrils are also features that are used; however, they become invisible as soon as the user tilts his head downwards. The mouth is also easy to find except when covered by a moustache or a beard. Several authors use a set of these features to estimate a 3D head orientation. In [62], the authors address the problem of estimating head pose over a wide range of angles from low-resolution images. Faces are detected using chrominance-based features. Grey-level normalised face images serve as input for linear auto-associative memory. One memory is computed for each pose using a Widrow-Hoff learning rule. Head pose is classified with a winner-takes-all process. Fitzpatrick [124] demonstrates a feature-based approach to head pose estimation without manual initialisation. For feature detection and tracking the cheapest paths across the face region is found, whereby the cost of a path depends on the darkness of crossed pixels. The paths will therefore avoid dark regions and a pair of avoided regions is assumed to be the pair of

eyes. The algorithm is thus dependant on the visibility of the eyes. Head pose is then determined based mainly on the head outline and the eye position. Gorodnichy [125] demonstrates a way to track the tip of the nose by using the resemblance of the tip of the nose with a sphere with diffuse reflection. This template is then searched in the image. This approach does not estimate the head pose, but simply tracks the nose tip across the video images and therefore a pose recognition task has to be added. In [61], a novel approach to estimate head pose from monocular images, which roughly classifies the pose as frontal, left profile, or right profile is presented. Subsequently, classifiers trained with adaboost using Haar-like features, detect distinctive facial features such as the nose tip and the eyes. Based on the positions of these features, a neural network finally estimates the three continuous rotation angles used to model the head pose.

Appearance-based subspace methods that treat the whole face as a feature vector in some statistic subspace has recently become popular. They avoid the difficulties of local face feature detection and face modelling. However, in the subspace, the distribution of face appearances under variable pose and illumination is always a highly non-linear, non-convex and maybe twisted manifold, which is very difficult to analyse directly [126]. Murase and Nayar [127] make a parametric description of this nonlinear manifold to estimate pose in a single PCA subspace. Pentland *et al.* [128] construct the view-based subspaces to detect face and estimate pose. The same idea is used in [85] to estimate head poses in the Independent Subspace Analysis (ISA) subspace. Some approaches solve this problem by kernel-based methods such as Support Vector Regression (SVR) [129] and Kernel Principal Component Analysis (KPCA) [113].

As shown in this section, the area of head pose estimation is rich and at the same time opens to interesting new avenues of investigation. The solutions proposed in this work however use two model-based approaches, namely, the template matching and PCA and a symmetry-based approach, which can be classified among appearance-based approaches. The solutions presented in this section focus mostly on single frame head pose estimation but not on the way in which these positions vary. It is, however, an important component of the intent recognition solution proposed in this work. The next section focuses on hand gesture recognition.

2.5 Hand gesture recognition

Hand gesture recognition from video images is of considerable interest as a means of providing simple and intuitive man-machine interfaces. Possible applications range from replacing the mouse as a pointing device to virtual reality, communication with the deaf and to Human-Computer Interaction (HCI). M.W. Krueger [130] proposed gesture-based interaction as a new form of HCI in the middle of the 1970s initially, which has since witnessed a growing interest in aiming at making HCI as natural as possible [131]. Much human visual behaviour can be understood in terms of the global motion of the hands. Such behaviours include most communicative gestures [132], [133] as well as movements performed in order to control and manipulate physical or virtual objects [134], [135], [136], [137]. Hand gestures and poses are not only extensively employed in human non-verbal communication [138], but are also used to complement verbal communication as they are co-expressive and complementary channels of a single human language system [139], [140], [141]. The primary goal of any automated gesture recognition system is to create an interface that

is natural for humans to operate or communicate with a computerised device [142], [143]. There are three main categories of hand gesture analysis approaches [144]:

- Glove-based analysis [145]
- Vision-based analysis that can be divided into model-based [146] and state-based [147], and analysis of drawing gestures
- There are also solutions that approach the problem from a neuroscience point of view [148]

Glove-based approaches have several drawbacks including the fact that they hinder the ease and natural way with which the user can interact with the computer-controlled environment; they also require long calibration and setup procedures [143]. The non-intrusive property of vision-based approaches makes them more suitable, thus rendering them probably the most natural way of constructing a human-computer gesture interface as they do not require any additional devices (e.g. gloves) and can be implemented with off-the shelf devices (e.g. webcams) [149]. Yet it is also the most difficult type of approach to implement in a satisfactory manner.

There are two main approaches in hand pose estimation. The first approach is the full Degree Of Freedom (DOF) hand pose estimation that targets all the kinematic parameters (i.e., joint angles, hand position or orientation) of the skeleton of the hand, leading to a full reconstruction of hand motion [143]. The second one consists of ‘‘partial pose estimation’’ methods that can be viewed as extensions of appearance-based systems that capture the 3D motion of specific parts of the hand such as the fingertip(s) or the palm. These systems rely on appearance-specific 2D image analysis to enable simple, low DOF tasks such as pointing or navigation. 3D hand models offer a way of more elaborate modelling of hand gestures but lead to computational hurdles that have not been overcome given the real-time requirements of HCI. Appearance-

based models lead to computationally efficient “purposive” approaches that work well under constrained situations but seem to lack the generality desirable for HCI [150].

There are an increasing number of vision-based gesture recognition methods in the literature. Baudel and Beaudouin-Lafom [151], Cipolla *et al.* [152], and Davis and Shah [153] all describe systems based on the use of a passive “data glove” with markers that can be tracked relatively easily between frames. A 3D structure from image sequences is recovered in [152] but does not attempt to classify gestures. David and Shah [154] propose a model-based approach by using a finite state machine to model four qualitatively distinct phases of a generic gesture. Hand shapes are described by a list of vectors and then matched with the stored vector models. Darrell and Pentland [155] propose a space-time gesture recognition method. Signs are represented using sets of view models, and then matched to stored gesture patterns using Dynamic Time Warping (DTW). Cui and Weng [156] developed a system based on a segmentation scheme which can recognise 28 different gestures in front of complex backgrounds. In [157] Ohknishi and Nishikawa propose a new technique for the description and recognition of human gestures. The proposed method is based on the rate of change of gesture motion direction that is estimated using optical flow from monocular motion images. Nagaya *et al.* [158] propose a method to recognise gestures using an approximate shape of gesture trajectories in a pattern space defined by the inner-product between patterns on continuous frame images. Heap and Hogg [159] present a method for tracking a hand using a deformable model, which also works in the presence of complex backgrounds. The deformable model describes one hand posture and certain variations of it and is not aimed at recognising different postures. Zhu and Yuille [160] developed a statistical framework using PCA and stochastic shape grammars to represent and recognise the shapes of animated objects.

It is called Flexible Object Recognition and Modelling System (FORMS). Rehg and Kanade [161] describe a system that does not require special markers. They use a 3D articulated hand model that they fit to stereo data but do not attempt gesture recognition. Blake *et al.* [162] describe a tracking system based on a real-time "snake" that can deal with arbitrary pose, but treats the hand as a rigid object.

An important application of hand gesture recognition is sign language understanding [132]. In [163], a large set of isolated signs from a real sign language is recognised with some success using a low-end instrumented glove using two machine learning techniques:

- Instance-Based Learning (IBL)
- Decision-tree learning.

Simple features were extracted from the instrumented gloves, namely the distance, energy and time of each sign. They have several advantages among which the most important are cost, processing power and the fact that the data extracted from a glove are concise and accurate. On the other hand, gloves are an encumbrance to the user and today's most convenient solutions require the property of being non-intrusive. In addition to instrumented gloves, early approaches to the hand gesture recognition problem in a robot control context involved the use of markers on the finger tips [164]. Again, the inconvenience of placing markers on the user's hand makes this solution less suited in practice. Liang *et al.* [165] developed a gesture recognition system for TSL using Data-Glove to capture the flexion of 10 finger joints, the roll of palm and other 3D motion information. In [166], [167] and [168], two visual HMM-based systems are presented for recognising sentence-level continuous American Sign Language (ASL) using a single camera to track the user's unadorned hands. To segment each hand initially, the algorithm scans the image until it finds a pixel of the

appropriate colour, determined by an a priori model of skin colour. Given this pixel as a seed, the region is grown by checking the eight nearest neighbours for the appropriate colour. Each pixel checked is considered to be part of the hand. The tracking stage of the system does not attempt a fine description of hand shape, instead, concentrating on the evolution of the gesture through time. In [169], a gesture recognition method for Japanese sign language is presented making use of the computational model called Parallel Distributed Processing (PDP) and a recurrent neural network for recognition. Huang *et al.* [170] use a 3D neural network method to develop a Taiwanese Sign Language (TSL) recognition system to identify 15 different gestures. Lockton *et al.* [171] propose a real-time gesture recognition system, which can recognize 46 ASL letter spelling alphabet and digits. The gestures consist of "static gestures" where the hand does not move.

More solutions include a fast algorithm proposed in [164] for the automatic recognition of a limited set of gestures from hand images for a robot control application. The approach contains steps to segment the hand region based on skin colour statistics and size constraints, locating the fingers by finding the Centre Of Gravity (COG) of the hand region as well as the farthest point from the COG, and finally classifying the gesture by constructing a circle centred at the COG that intersects all the fingers that are active in the count and subsequently extracting a 1D binary signal by following the circle. The algorithm is invariant to translation, rotation, and scale of the hand and does not require the storage of a hand gesture database. In [138], a robust hand gesture detection and recognition system for dynamic environments is proposed. The system is based on the use of a cascade of boosted classifiers for detection of hands and gesture recognition, together with the use of skin colour segmentation and hand tracking procedures. The authors in [142]

present a system that performs automatic gesture recognition using a unified technique for segmentation and tracking of faces and hands through a skin colour detection algorithm and a static and dynamic gesture recognition system based on PCA. An HMM-based gesture recognition algorithm is presented in [172] where the system uses a threshold model that calculates the threshold likelihood given an input pattern. For gesture segmentation, it detects the reliable end point of a gesture and finds the start point by back-tracking the Viterbi path from the end point. A visual hand gesture recognition technique that uses the fusion of a static and a dynamic recognition technique is proposed in [144].

The hand gestures can be divided into static hand gestures, which are represented by a single image of the hand, and dynamic hand gestures, which are represented by a sequence of images, each one corresponding to a hand posture within the gesture (hand movement). The static signature uses the local orientation histograms in order to classify the hand gestures. For the dynamic gesture recognition algorithm each gesture is represented by a sequence of images. The dynamic signature used for classification is the superposition of all hand region skeletons for each image within the sequence. The recognition is performed by comparing this signature with the ones from a model of the gestures, using Baddeley's distance as a measure of dissimilarities between model parameters. The pre-processing steps consist of the following operations:

- Binary image computation
- Binary image enhancement
- Hand region extraction.

The authors in [173] address the problem of high computation cost to solve the finger inverse kinematics in conventional model-based hand gesture analysis systems. They propose a fast hand model fitting method for the tracking of hand motion by finding the closed-form inverse kinematics solution for the finger fitting process, and defining the alignment measure for the wrist fitting process. Their proposed method however requires markers. In [174], a robust method for hand tracking in a complex environment using mean-shift analysis and Kalman filter in conjunction with a 3D depth map is proposed. Mean-shift analysis uses the gradient of Bhattacharyya coefficient as a similarity function to derive the candidate of the hand that is most similar to a given hand target model and Kalman filter is used to estimate the position of the hand target. A real-time vision system is presented in [175], which uses a fast segmentation process to obtain the moving hand from the whole image, which is able to deal with a large number of hand shapes against different backgrounds and lighting conditions. The recognition process identifies the hand posture from the temporal sequence of segmented hands through a robust shape comparison carried out through a Hausdorff distance approach operating on edge maps. The system‘s visual memory stores all the recognisable postures, their distance transform, their edge map and morphologic information. In [176], the author presents a real-time stereo vision hand tracking system that can be used for interaction purposes. The system can track the 3D position and 2D orientation of the thumb and index finger of each hand without the use of special markers or gloves. The method includes a background subtraction, skin colour segmentation, a region extraction, and a contour-based feature extraction. In [177], a novel method for hand gesture recognition is presented based on Gabor filter and SVMs. Gabor filters are first convolved with images to acquire desirable hand gesture features. PCA is then used to reduce the dimensionality of the feature

space. With the reduced Gabor features, SVM is trained and exploited to perform the hand gesture recognition task. Other methods use optical flow where the position of the moving hand is estimated and segmented into motion blobs. Gestures are recognised using a rule-based technique based on characteristics of the motion blobs such as relative motion and size [178]. A histogram of local orientation [179] is also used as a feature vector for gesture classification and interpolation.

As discussed in [143], the main difficulties encountered in the design of hand pose estimation systems include:

- High-dimensional problem: the hand is an articulated object with more than 20 DOF.
- Self-occlusions: Since the hand is an articulated object, its projection results in a large variety of shapes with many self-occlusions, makes it difficult to segment different parts of the hand and extract high level features.
- Processing speed: With the current hardware technology, some existing algorithms require expensive, dedicated hardware, and possibly parallel processing capabilities to operate in real-time.
- Uncontrolled environments: For widespread use, many HCI systems would be expected to operate under non-restricted backgrounds and a wide range of lighting conditions.
- Rapid hand motion: The combination of high speed hand motion and low sampling rates introduces additional difficulties for tracking algorithms.

Since it is difficult to satisfy all the issues listed above simultaneously, some studies apply restrictions on the user and the environment. In this thesis, the problem is defined in such a manner that both the user and the environment have some restrictions. The environment is restricted as the solution is intended to be used in a

wheelchair with one camera facing the user's right hand from its dorsal view and another camera in front of the user's face. For the user, the motions of interest are rotation with respect to a horizontal axis and vertical motion of a relatively rigid hand solving the issue of the multiplicity of DOF. No self occlusion is therefore anticipated, and the environment is more or less controlled as the camera is already facing the objects of interest.

It is also evident that the types of motion of interest in this thesis are less explicit and pronounced than the gestures found in the literature for sign language and human-robot interaction. Furthermore, some of these solutions are invariant to rotations [142], [164], and therefore may not all be able to detect these hand motions as defined in this thesis.

2.6 Conclusion

As the literature reveals, intention detection aimed for HCI and collaboration is a fairly rich field of investigation where many published work exist, and where there is still room for new contributions. Furthermore, many robotic and intelligent power wheelchairs that share control with users or help them perform the tasks that they cannot carry out on their own, were developed where computerised intent awareness of the user constitutes an essential component. One useful speed and direction intent indicator for these power wheelchairs is the motion of the head as it remains available for many physical disabilities. Hand gesture has also been shown to be an important means of non-verbal communication, a complement to verbal communication and a means for interaction with virtual and physical reality.

The solutions found in the literature for head pose estimation applied to wheelchair motion, allow symbolic commands based on the head is posture. The disadvantage is that the intention is therefore indicated by a single frame and therefore more vulnerable to misdetection. Hand gesture recognition solutions found in the literature focus more on the change in the hand's contour shape and are typically applied to sign language applications. The literature doesn't contain hand gesture recognition solutions where the motion of the hand is a micro-operation such as the rotation and vertical motion described in this thesis, for which the approaches found in the literature are typically invariant or unusable for robust classification. The next two chapters describe the methods proposed in this thesis.

Chapter 3

Head-Based Intent Recognition

3.1 Introduction

One of the visual intent indicators used in this work is the frontal view of the head (face) in motion. The motivation behind this choice is its availability and flexibility for a wide range of disabilities. Moreover, the head in motion is a useful intent indicator as it presents separable patterns for different poses. The solution consists of a camera with the head as the object of interest in its field of view. The head performs two types of motion: Rotation and vertical motion to indicate an intention in direction and speed variation respectively. Head rotation in a particular direction (right or left) is selected to indicate the chosen direction the subject intends to take. Vertical head motion (up or down) is chosen to indicate the subject's speed variation intent where the head going up is chosen to indicate a decrease in speed, and the head going down is chosen to indicate an increase.

The visual solution proposed in this thesis accepts a video sequence as the input with the head in rotation and vertical motion as object of interest, and gives direction and speed variation intent respectively as output. Intent recognition is achieved by analysing the motion of the head through the video sequence rather than looking at a single frame. In this work 10 frames are required as input to the proposed algorithm that maps them into a vector referred to in this work as the "intention curve".

This chapter provides a detailed description of the head-based intent recognition methods proposed in this thesis. The pre-processing steps of detection and tracking of the frontal view of the face within a video sequence are implemented using skin colour detection and PCA on the detected skin colour region resulting in a smaller image frame containing the only face in the field of view. Note that in this work, tracking only consists of repeating the detection task on a smaller region that is slightly larger than the head detected region of the previous frame. For recognition of the head in rotation, a symmetry-based approach is used on each face detected frame resulting in a symmetry curve where the centre of gravity (COG) and the *y*-intercept of the line approximating that symmetry curve are used to construct the intention curves. These intention curves are subsequently classified using a decision rule based on their increasing, decreasing or constant tendency. For recognition of the head in vertical motion the intention curves are constructed using a PCA-based approach on each frame of the input sequence, and are classified using a decision rule also based on their increasing, decreasing and constant propensity. Furthermore, a method by Jia and Hu [66], [67] based on adaboost, camshift and nose template matching is also implemented for the comparison of results.

To distinguish between the two different sets of motions, rotation detection of the head for direction intent recognition is performed first and if no significant change in position (rotation wise) is observed, detection of the vertical motion of the head for speed variation intent recognition is then performed.

3.2 Pre-processing steps: face detection and tracking

The topic of face detection is a very rich area of research in the literature, providing solutions to determine if a face is present in an image frame as well as the exact location of that face [180]. The methods for face localisation in a single frame can be divided into four categories:

- Knowledge-based methods: They encode human knowledge of what constitutes a typical face: e.g., the relationships between facial features.
- Feature invariant approaches: They find structural features of the face that exist even when the pose, viewpoint, or lighting conditions vary.
- Template matching methods: where the correlations between an input image and the stored templates are computed for detection.
- Appearance-based methods: where in contrast to template matching, templates are learned from a set of training images, which should capture the representative variability of facial appearance.

The present solution makes use of skin colour detection, which is a very popular feature invariant approach. It was shown that colour is the most powerful means of discerning object appearance. So it is better than greyscale processing leading to the detection through facial features such as the eyes and the mouth. Another merit that may be attributed to skin colour detection over the detection of other facial feature is its diversity of application including hand detection that is of interest in this thesis (refer to Chapter 4). Two solutions are implemented to model the skin colour: The first makes use of a colour histogram in the HSV colour space [181], while the second

makes use of a variant of adaboost to learn the skin colour in the YCrCb colour space [182].

3.2.1 Histogram-based skin colour detection

Typically skin colour is modelled using a histogram, a single Gaussian distribution or a mixture of Gaussians, although other approaches can also be found. However, among those three principal skin colour models, the authors in [183] have demonstrated that the histogram model is superior to the others, easier to implement and computationally efficient. The different colour spaces used in skin colour detection include HSV, normalised RGB, YCrCb, YIQ and CIELAB. According to [184], HSV yields the best performance for skin colour pixel detection. In this colour space, H stands for the Hue component, which describes the shade of the colour, S stands for the Saturation component, which describes how pure the Hue (colour) component is, while V stands for the Value component, which describes the brightness. The removal of the V component takes care of varying lighting conditions. H varies from 0 to 1 on a circular scale, that is, the colours represented by H=0 and H=1 are the same. S varies from 0 to 1, 1 representing 100 percent purity of the colour. H and S scales are partitioned into 100 levels and the colour histogram is formed using H and S.

For skin colour training 69 601 pixels are used spanning 10 different subjects with different skin colours, to form a separate colour histogram for each component H and S: For each pixel, H and S values are found and the bin corresponding to these H and S values in the histogram is incremented by 1. Figure 3-1 depicts the histogram for the H and the S components separately. To classify a new pixel as skin colour or not (background), a common threshold λ = 2000 for both components H and S is

chosen empirically by trial and error according to the height of the bins where the skin colour is sufficiently frequent: Let p be a pixel to be classified as skin colour or non-skin colour in the input image (refer to Figure 3-2 (Part a) for examples of 576×768 input images), bin_H and bin_S the bins in the histogram corresponding to the H and S component values associated with the pixel p. The classification task is performed by the decision rule h given below (refer to Equation 3-1), and the resulting skin colour detection is depicted in Figure 3-2 (Part b):

$$h(p) = \begin{cases} skin\,colour & , \quad bin_H \geq \lambda \wedge bin_S \geq \lambda \\ non-skin\ \ colour & , \quad else \end{cases} \tag{3-1}$$

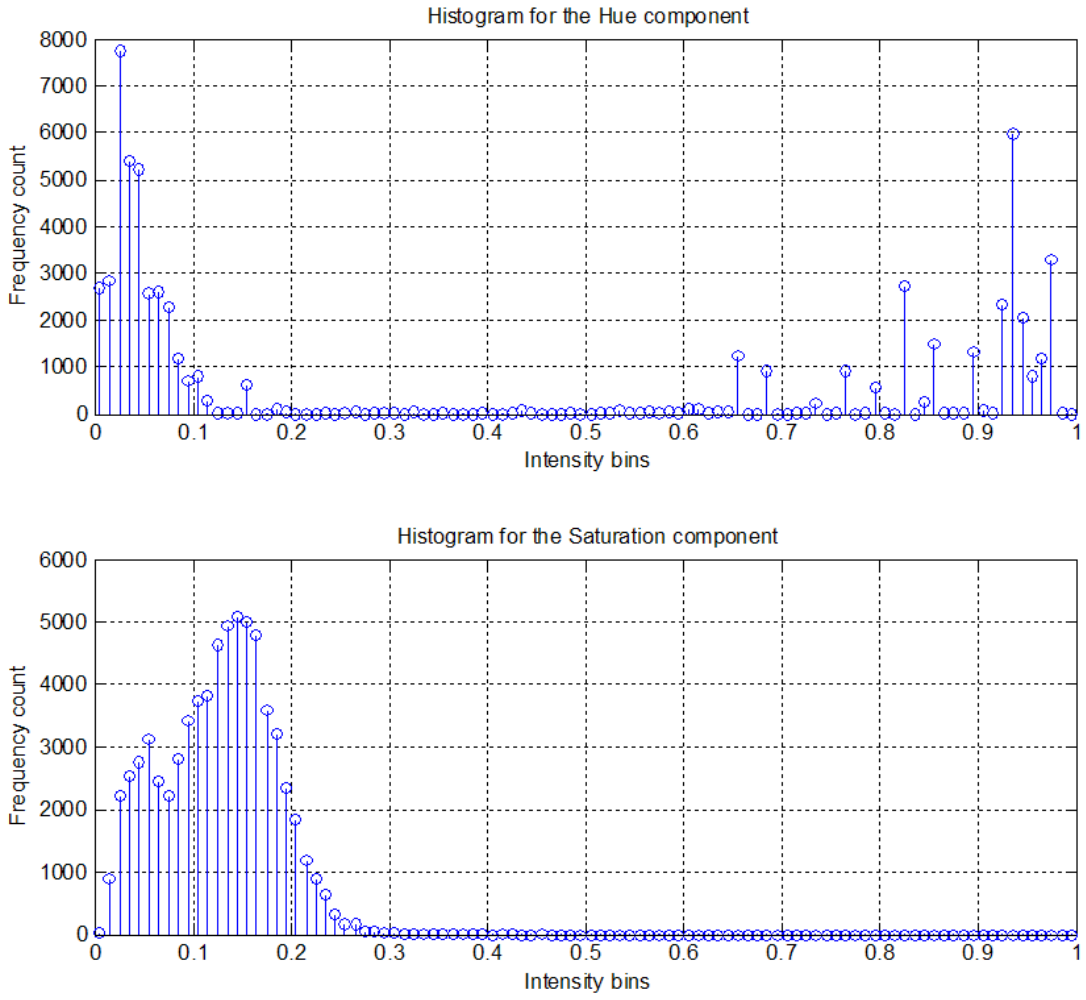

Figure 3-1: Skin colour histograms in the HSV colour space

3.2.2 Adaboost-based skin colour detection

The foundational colour space is RGB while the colour space that displays the best performance according to [184] is HSV. In [182] however, it is asserted that the choice of the YCrCb colour space as opposed to the two colour space mentioned previously is justified by its hardware-oriented advantages. Y represents the luminance component, while Cr and Cb represent the chrominance components of an image. The approach to learn skin colour pixel is based on the adaboost learning algorithm where the key factor for identifying skin colour is intensity. The task of colour segmentation is therefore based on the fact that colour distributions at different intensities have different centres of gravity, different means, and different standard deviations, that is, skin colour has different statistical features with different intensities. This method, based on skin colour training using adaboost, performs more robustly than the traditional threshold technique [183] for skin colour extraction, especially under poor or strong lighting conditions.

Boosting consists of the addition of a new weak classifier (refer to Equation 3-2), until the error is decreased to a specific threshold. The weak classifier is designed to select a circularity, which can contain as much skin colour pixels as possible. This circularity is required to round more than only 50% of the points, as a weak classifier for adaboost needs to be only a little better than random guess. A weak classifier $h_j(p)$ consists of the centre $c(cr,cb)$ and the radius r and is given as:

$$h_j(p) = \begin{cases} 1 & if \quad (p.Cr - c.Cr)^2 + (p.Cb - c.Cb)^2 <= r^2 \\ 0 & otherwise \end{cases} \tag{3-2}$$

where p is a pixel in the image. Given example pixels $p_1(Y,Cr,Cb)...p_n(Y,Cr,Cb)$, which are all skin pixels (all positive examples) from images spanning 10 different subjects in the training set with different skin colours, the algorithm for training skin colour is given below in Table 3-1.

Table 3-1: Adaboost algorithm for skin colour learning

- Divide these examples into m intervals: $it_1...it_m$, according to specific value ranges of Y.
- For each interval it_k:
  - Initialise weights $w_i = \frac{1}{n}$, where n is the number it_k examples.
  - For $t = 1, ..., T$
    - Normalise the weights

      $w_{t,i} = \frac{w_{t,i}}{\sum_{j=1}^{n} w_{t,j}}$, so that $w_{t,i}$ is a probability distribution.
    - Train a circularity

      $c.Cr = \sum_{j=1}^{n} p.Cr \times w_{t,j}$

      $c.Cb = \sum_{j=1}^{n} p.Cb \times w_{t,j}$

      $r = E(r) + \delta(r)$, with

      $E(r) = \sum_{j=1}^{n} p_j l \times w_{t,j}$ and $\delta(r) = (\sum_{j=1}^{n} (p_j l - E(r))^2 \times w_{t,j})^{\frac{1}{2}}$

      where $p_j l$ is distance of p_j to the centre. The error is evaluated as $\varepsilon_j = \sum_i w_{t,i} \left| h_j(p) - 1 \right|$
    - Update the weights

      $$w_{t+1,i} = \begin{cases} w_{t,i} \times \frac{\varepsilon_j}{1-\varepsilon_j} & if\ h_j(p) = 0 \\ w_{t,i} & otherwise \end{cases}$$
    - For interval it_k, the final strong classifier is

      $$s_k(p) = \begin{cases} 1 & if\ \sum_{t=1}^{T} \alpha_t h_t(p) \geq \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\ 0 & otherwise \end{cases},$$

      where $\alpha_t = log_2 \frac{1-\varepsilon_t}{\varepsilon_t}$

For skin colour classification of each pixel p in an input image (refer to Figure 3-3 (Part a) for examples of 576×768 input images), the first step is to determine the interval it_k among the m intervals, to which p belongs. If the pixel doesn't belong to any interval it is discarded as a non-skin colour pixel, otherwise the appropriate strong classifier s_k is subsequently used to determine if the pixel in interval it_k is a skin-colour pixel or not (refer to Equation 3-3). The resulting skin colour detection is depicted in Figure 3-3 (Part b).

$$map(p.Y) = \begin{cases} non-skin\ colour, & \forall k = \{1,...,m\},\ p.Y \notin it_k \\ s_k(p), & \exists k = \{1,...,m\},\ p.Y \in it_k \end{cases} \tag{3-3}$$

3.2.3 Face detection and localisation

After classifying skin colour pixels and non-skin colour pixels in a given image such as those portrayed in Figures 3-2 (Part a) and 3-3 (Part a), a higher level processing is required to determine whether a face is present as well as the exact location of the face. Unlike [181] and [182] that made use of some geometric constraints through connectivity analysis and identification of the connected region with the shape of a typical face through the well known golden ratio: $r = \frac{1+\sqrt{5}}{2} \pm \tau$ (where τ is a tolerance) and a cascade of neural networks with the Boltzmann factor respectively, the proposed method makes use of morphological image processing operations namely erosion and dilation, connected component labelling and Principal Component Analysis (PCA).

3.2.3.1 Erosion

Erosion is a morphological image processing operation that removes the extraneous pixels on object boundaries. It was originally defined for binary images, later being extended to greyscale images, and subsequently to complete lattices [186]. It is used in this work to remove noise from the binary image resulting from the skin colour detection step. The erosion uses a specified neighbourhood. The state of any given pixel in the output image is determined by applying a rule to the neighbourhood of the corresponding pixel in the input image such that if every pixel in the input pixel's neighbourhood is a skin colour pixel, then the output pixel is a skin colour pixel. Otherwise, the output pixel is a non-skin colour pixel.

The neighbourhood can be represented by an arbitrary shape and size called the structuring element, and is chosen in this work to be a 3×3 square. The centre pixel in the structuring element represents the pixel of interest, while the elements in the matrix that are skin colour define the neighbourhood. Let I be the binary image such as those depicted in Figures 3-2 (Part b) and 3-3 (Part b) resulting from the histogram-based skin colour and the adaboost-based skin colour detection approaches respectively, and R, a detected skin colour region in I. The erosion of R by the structuring element S is defined as follows:

$$R \ominus S = \bigcap_{s \in S} R_{-s}$$

$$= \{ z \in I \mid S_z \subseteq R \} \tag{3-4}$$

where S_z is the translation of S by the vector z, that is, $S_z = \{ s + z \mid s \in S \}, \forall z \in I$ (s is the centre of the structuring element S and z is each pixel in the image I). The resulting binary images are illustrated in Figures 3-2 (Part c) and 3-3 (Part c) for adaboost-based and histogram-based skin colour detection respectively, and it can be observed that removing noise also costs on the skin colour detection task. This is compensated by the dilation operation.

3.2.3.2 Dilation and connected component labelling

Dilation is a morphological image processing operation that adds pixels to the boundaries of objects. Similarly to erosion, it was originally defined for binary images, later being extended to greyscale images, and subsequently to complete lattices [186]. It is used in this work to compensate for the loss in shape resulting from the noise removal process through erosion in the previous step: The state of any given pixel in the output image is determined by applying a rule to the neighbourhood of the corresponding pixel in the input image such that if any pixel in the input pixel's neighbourhood is a skin colour pixel, then the output pixel is a skin colour pixel. Otherwise, the output pixel is a non-skin colour pixel.

The structuring element neighbourhood is chosen in this work to be a 10×10 square, for probing and expanding the shapes contained in the input image. Let I be the binary image such as those displayed in Figures 3-2 (Part b) and 3-3 (Part b) resulting from the histogram-based skin colour and the adaboost-based skin colour detection approaches respectively, and R, a detected skin colour region in I. The dilation of R by the structuring element S is defined as follows:

$$R \oplus S = \bigcup_{s \in S} R_s$$
$$= \{ z \in I \mid (S^{'})_z \cap R \neq \phi \} \tag{3-5}$$

where $S^{'}$ denotes the symmetric of *S*, that is, $S^{'} = \{ s \in I \mid -s \in S \}$.

As illustrated in Figures 3-2 (Part d) and 3-3 (Part d) for histogram-based and adaboost-based skin colour detection respectively, the dilation operation is aimed at retrieving the integrity of the skin colour pixels, but retrieves also some of the noise that has not been completely removed by the erosion process. To address this problem a connected component labelling task is performed using the 8-connected neighbourhood approach, that is, two skin pixels belong to the same region if one is in any of its 8 neighbouring places of the other [187]. Subsequently, the assumption that only one face is present in the field of view and that it corresponds to the highest connected component in the image guides a simple decision rule that only retains the connected component with the highest number of pixels: Let $R_i \in I$, where R_i of size $X_i \times Y_i$ is the i^{th} connected region. The region portrayed in Figures 3-2 (Part e) and 3-3 (Part e) for histogram-based and adaboost-based skin colour detection respectively, chosen to be the face region is R_k where:

$$k = \arg\max_i \left\{ \sum_{x=1}^{X_i} \sum_{y=1}^{Y_i} R_i(x,y) \right\}_{i=\{1,\ldots,n\}} \tag{3-6}$$

with *n* the number of connected components in the binary image resulting from the dilation process.

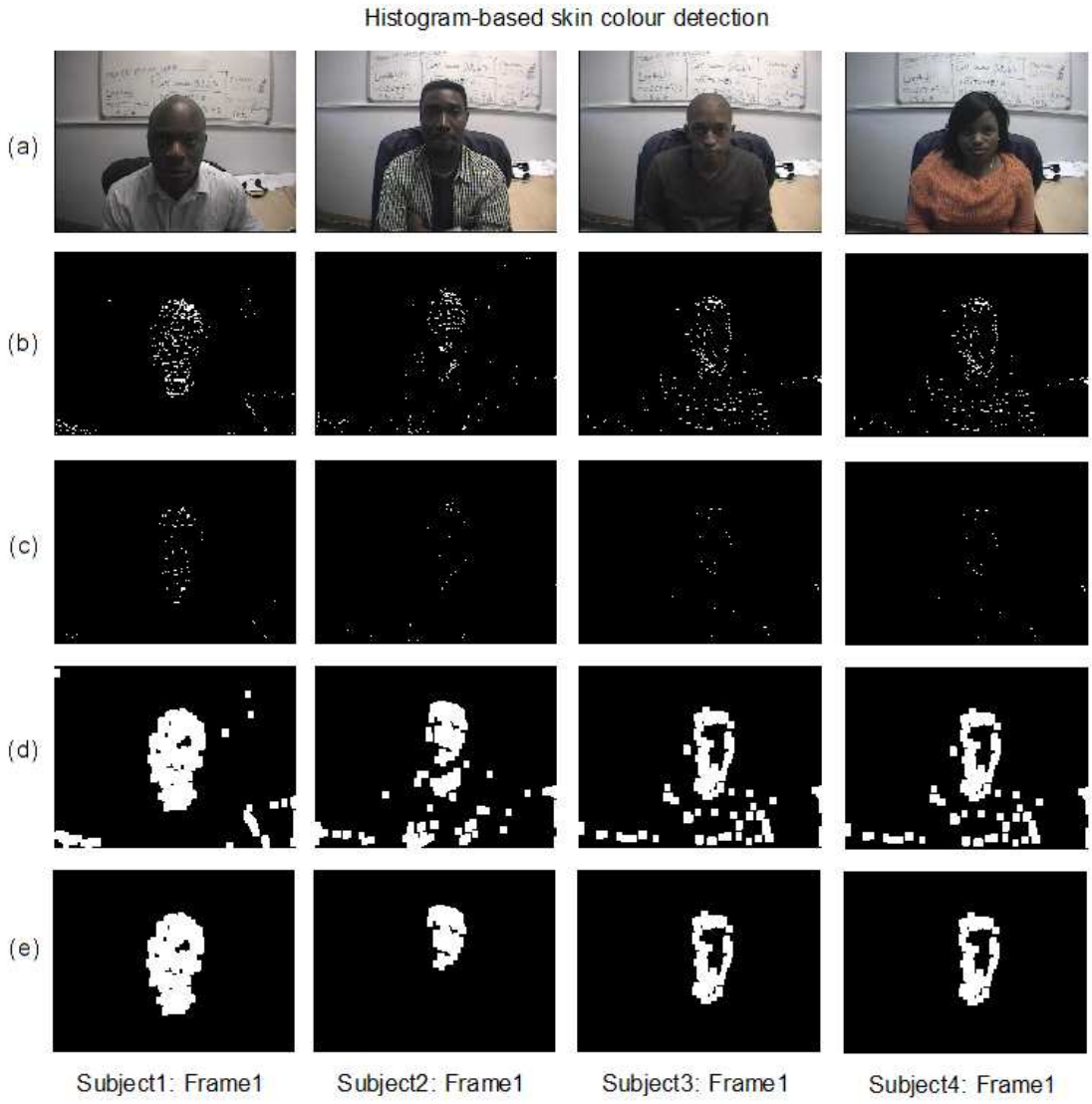

Figure 3-2: Histogram-based skin colour detection for face detection

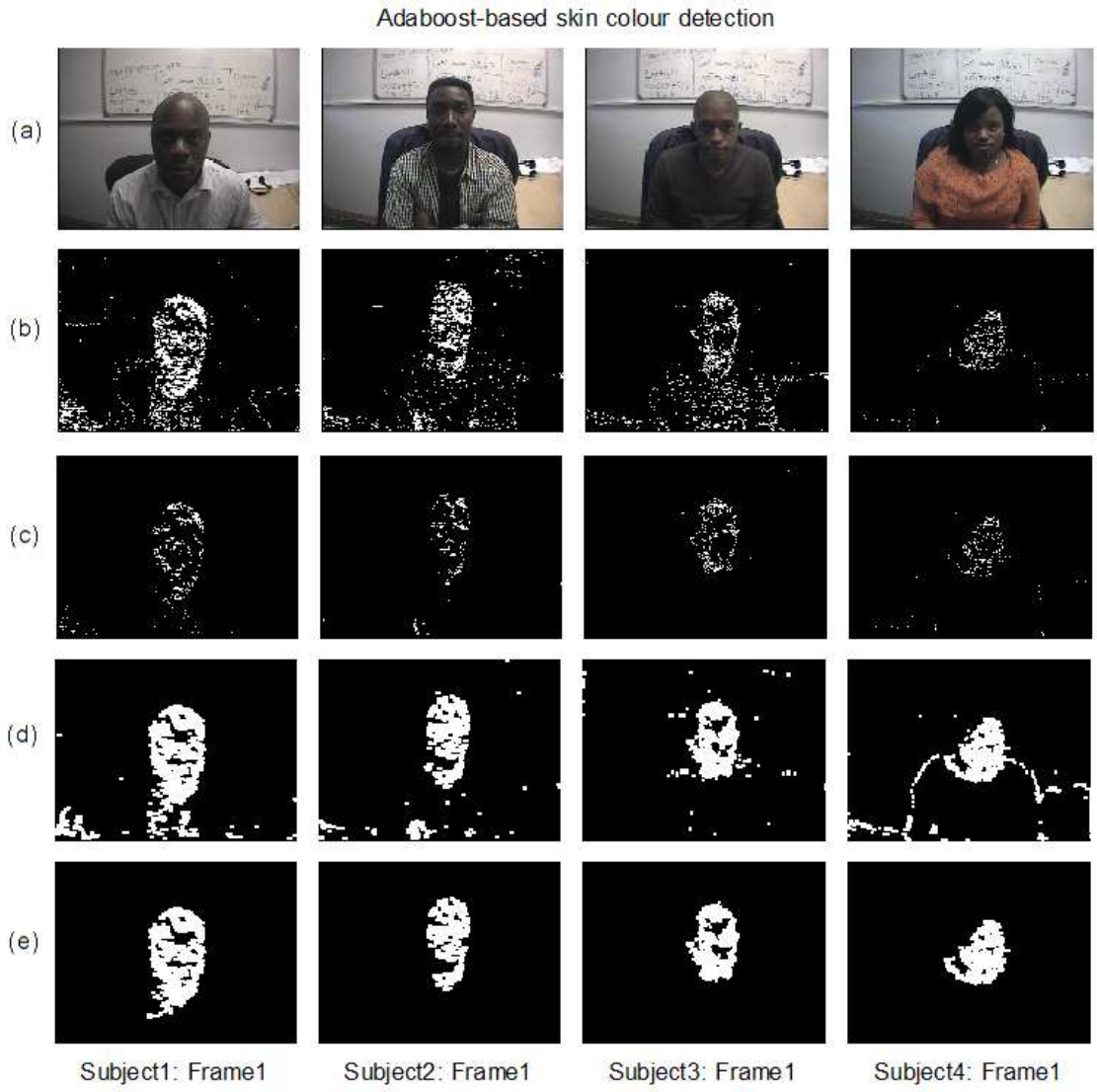

Figure 3-3: Adaboost-based skin colour detection for face detection

3.2.3.3 Principal Component Analysis

PCA stems from the fact that it is often advantageous to represent data in a reduced number of dimensions for improved classification performance. Essentially, dimensionality reduction can be achieved in two different ways:

- Feature selection: It identifies those variables that do not contribute to the classification task.
- Feature extraction: Also referred to as feature selection in the transformed space, it finds a transformation from the *p* measurements to a lower dimensional feature space.

PCA belongs to the second category. It is a linear feature extraction approach that finds linear projections to derive new variables in decreasing order of importance, which are linear combinations of the original variables and are uncorrelated. These projections capture the variability of features or separability of classes.

For face localisation, PCA maps a face into a lower dimensionality space through the generation of a set of eigenfaces: Suppose Γ is an $N^2 \times 1$ vector, corresponding to an $N \times N$ face image I where $N = 200$: the goal is therefore to represent Γ into a low-dimensional space. Let the training set of face images be I_1, I_2... I_M made of 10 different subjects with 5 faces each. These $N \times N$ images I_i are represented as $N^2 \times 1$ vectors Γ_i with $i = \{1,\ldots,M\}$, with $M = 50$. The mean face image is computed:

$$\Psi = \frac{1}{M} \sum_{i=1}^{M} \Gamma_i \tag{3-7}$$

The mean face is subtracted:

$$\Phi_i = \Gamma_i - \Psi \tag{3-8}$$

The $N^2 \times N^2$ covariance matrix C, which is a measure of how far the set of column vectors Φ_i of matrix A is spread out, is given as:

$$C = \frac{1}{M}\sum_{n=1}^{M}\Phi_n\Phi_n^T = AA^T \tag{3-9}$$

where $A = [\Phi_1 \Phi_2 ... \Phi_M]$ is an $N^2 \times M$ matrix. The eigenfaces are found by computing a set of eigenvectors of the covariance matrix C. These eigenvectors and their associated eigenvalues best describe the distribution of the data, as they represent the direction in which the face images in the training set differ from the mean image and how much these face images vary from the mean image in that direction respectively. Since the covariance matrix $C = AA^T$ is too large with a size of $N^2 \times N^2$, rendering the computation of the eigenvectors in Equation 3-10 not practical, the eigenvectors of the smaller $M \times M$ matrix A^TA can be computed (refer to Equation 3-11):

$$AA^T v_k = \lambda_k v_k \tag{3-10}$$

$$A^T A u_k = \lambda_k u_k \tag{3-11}$$

$$\begin{aligned} AA^T A u_k &= A\lambda_k u_k \\ &= \lambda_k A u_k \end{aligned} \tag{3-12}$$

where $k = \{1,...,M'\}$ with $M' = 15$. From Equations 3-10 and 3-12 it can be concluded that if u_k is an eigenvector of A^TA, then $v_k = Au_k$ is the eigenvector of C and therefore the eigenfaces. It has also been shown that all the M eigenvalues of A^TA are the M largest eigenvalues of AA^T [105]. Only M' eigenfaces, corresponding to the

M' largest eigenvalues among the M eigenvalues ($M' < M$) are kept. Figure 3-4 depicts the 10 eigenfaces associated with the 10 highest eigenvalues among the M' used in this work.

Figure 3-4: Examples of eigenfaces

These eigenfaces are the basis of the eigenspace (the training face space), and can be used to represent a new face by projecting it on the eigenspace and thereby recording how that new face differs from the original face: Given a new window in the skin colour detected region in the input image where a face must be detected (refer to Figures 3-2 (Part e) and 3-3 (Part e)). This window is scanned by an $N \times N$ (where N = 200) sub-windows to localize the face as follows:

- Representation of the sub-window image as an $N^2 \times 1$ vector Γ and normalise as follows:

$$\Phi = \Gamma - \Psi \tag{3-13}$$

where Ψ is given by Equation 3-7.

- Projection of Φ on the eigenspace:

$$\hat{\Phi} = \sum_{k=1}^{M'} w_k v_k \tag{3-14}$$

where $w_k = v_k^T \Phi$ and v_k's are eigenfaces and $\hat{\Phi}$ the projection of Φ on the eigenspace.

- The face Φ is therefore transformed into its eigenface components and represented by the vector:

$$\Omega = [\, w_1 \;\; w_2 \;\; \ldots \; w_{M'} \,]^T \tag{3-15}$$

- Calculation of the distance ε between the face image Φ and its projection $\hat{\Phi}$:

$$\varepsilon_d = \left\| \Phi - \hat{\Phi} \right\| \tag{3-16}$$

- If $\varepsilon_d < \lambda_d$, then Γ is a face, with $\lambda_d = 1.85$

Figure 3-5 (Part a) exhibits the face detection and localization for four different subjects using a histogram-based skin colour detection approach, and Figure 3-5 (Part b) illustrates the face detection and localization for four different subjects using an

adaboost-based skin colour detection method. Note that the tracking phase is implemented by repeating this detection task on a smaller search window around the detected face region of the previous frame.

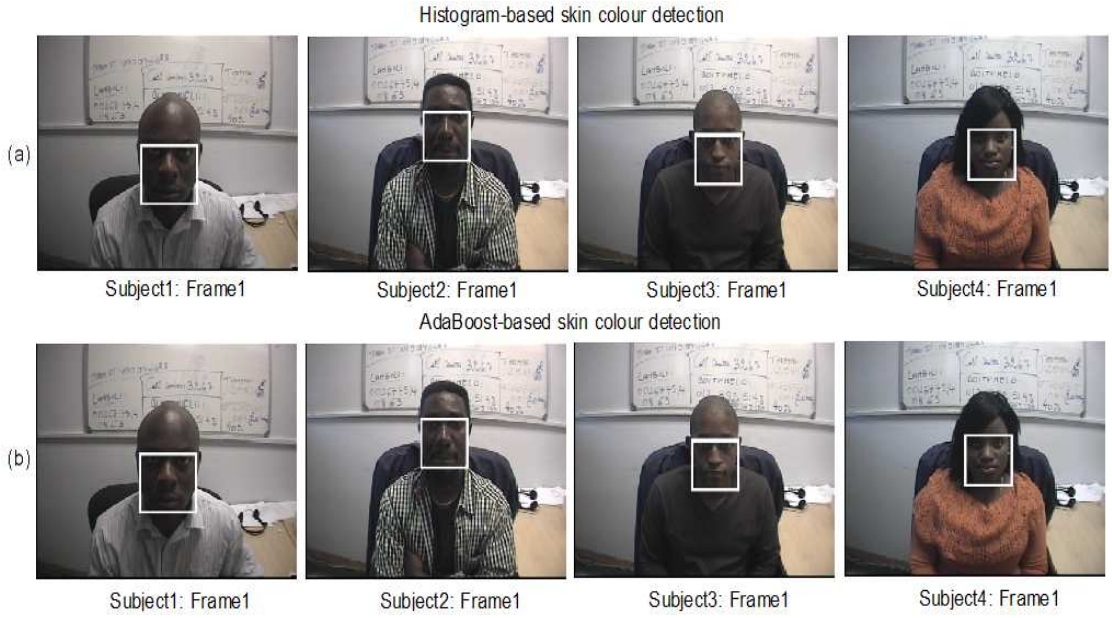

Figure 3-5: Face detection and localisation

3.3 Recognition of head-based direction intent

The estimated position of the face in rotation as depicted in Figure 3-6 is used as an intent indicator according to Table 3-2. As mentioned earlier, one of the motivations behind the choice of the head in motion as the intent indicator is its availability and flexibility for a wide range of disabilities.

Table 3-2: Head motion and corresponding direction intention

Motion of the head	Inferred Intention
Left rotation	Intent to go left
Right rotation	Intent to go right
No rotation (Centred position)	Intent to go straight

A symmetry-based approach is used to extract symmetry curves associated with the frontal view of the face. The assumption is that different positions of the face display different symmetry properties and therefore provide different symmetry curves. Intent recognition is implemented using the COGs of the symmetry curves and the *y*-intercepts of the lines approximating the symmetry curves combined with two different decision rules based on the difference of means and the statistics (means and standard deviation) in a Gaussian distribution of the COGs and the *y*-intercepts for single pose recognition but also based on the difference of means and the statistics (means and standard deviation) in a Gaussian distribution of the increasing, decreasing and constant tendencies, of the COG-based and *y*-intercept-based intention curves for intent recognition.

Figure 3-6: Frontal view of the head (face) in rotation

3.3.1 Symmetry-based Approach

The underlying assumption is that human faces viewed from the front are symmetric and when moved from their initial position (centred position), the symmetry they display breaks down. These separable patterns presented by these symmetry curves give the indication of a motion from the initial centred position to a new position (right or left). The symmetry curve, based on the work in [188], is given by

$$f(x) = \sum_{w=1}^{k} \sum_{y=1}^{Y} | I(x-w,y) - I(x+w,y) | \tag{3-17}$$

The symmetry-value $f(x)$ is evaluated $\forall x \in [k+1 \;\; X-k]$ where x is a pixel-column in the image, by taking the sum of the differences of two pixels at a variable distance $w: 1 \leq w \leq k$ from it on both sides making the pixel-column the centre of symmetry. This process is performed for each row and the resulting symmetry-value is the summation of these differences. The symmetry curve is composed of these symmetry-values calculated for all the pixel-columns in the interval $k+1 \leq x \leq X-k$. It was empirically established that the value of the maximum distance k that yields more separable symmetry curves associated with the head of the subjects among the different positions is $k = 9$. Figure 3-6 portrays three different positions of detected faces for five different subjects and Figure 3-7 depicts the symmetry curves associated with the different head poses depicted in Figure 3-6.

3.3.2 Centre of Gravity (COG) of the Symmetry Curve

The COG, also known as centre of mass, is the location in the symmetry curve at which all the values of the curve are considered to be centred and is given by

$$C = \frac{\sum_{i=1}^{n} x_i f(x_i)}{\sum_{i=1}^{n} f(x_i)} \tag{3-18}$$

where the symmetry curve is defined by the function: $f : x \rightarrow f(x)$ with $f(x)$ given by Equation 3-17 and x a pixel column in the face image. The symmetry curves displayed in Figure 3-7 differ for the three different positions of the face with which they are associated and therefore yield different COGs giving an indication of the position of the head. Figure 3-8 depicts the symmetry curves and the COGs associated with them for different positions of the head in rotation shown by the vertical lines on the plot of the symmetry curves.

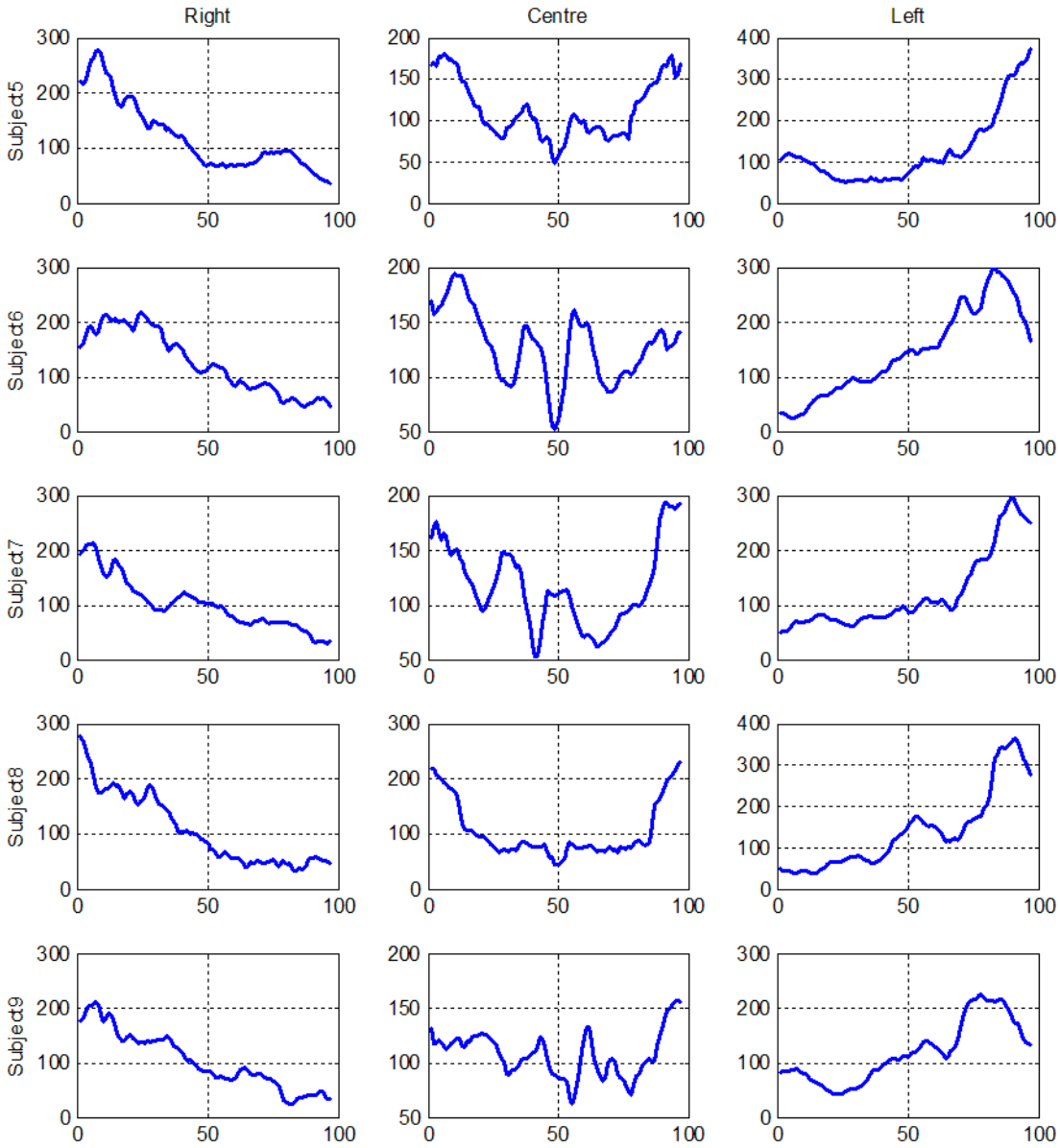

Figure 3-7: Symmetry curves for faces in Figure 3-6

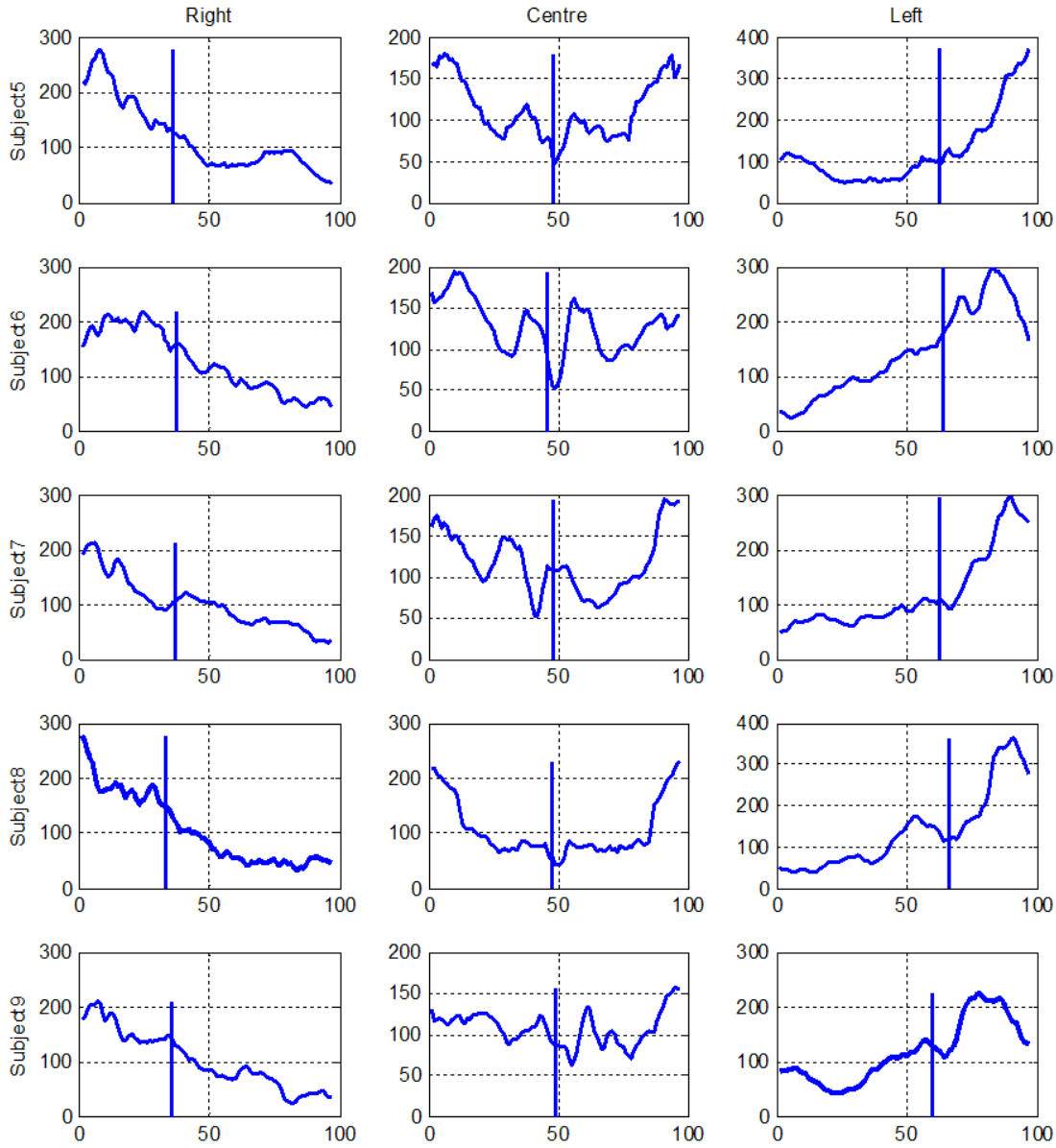

Figure 3-8: Symmetry curves with COG for faces in Figure 3-6

3.3.3 Linear Regression on the Symmetry Curve

Another way to classify the symmetry curves is to find the lines that approximate the symmetry curves as their *y*-intercepts differ for the three different positions. This can

be achieved by a linear regression approach on the symmetry curves: Given a curve $y = f(x)$, the goal of linear regression is to find the line that best predicts y from x where x is the independent variable and y the dependent one. Linear regression does this by finding the line that minimises the sum of the squares of the point's vertical distances from the line: Let $f : X \rightarrow Y = f(x)$ be a function describing a symmetry curve, a linear regression is a form of regression analysis in which the relationship between y and x is modelled by a least squares function called linear regression equation:

$$Y = X\beta + \varepsilon \tag{3-19}$$

where $Y = [\, y_1 \;\; y_2 \ldots y_N \,]$ and $X = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ x_1 & x_2 & \ldots & x_N \end{bmatrix}^T$.

The least squares estimate is thus given by

$$\beta = (X' X)^{-1} X' Y \tag{3-20}$$

where β gives the values of the y-intercept $\beta(1)$ and the angle $\beta(2)$ of the line with respect to the x-axis. It was empirically established that the y-intercepts of the resulting lines are more discriminative than the angles. Figure 3-9 displays the symmetry curves associated with the faces in rotation in Figure 3-6 with the lines approximating them resulting from this linear regression approach.

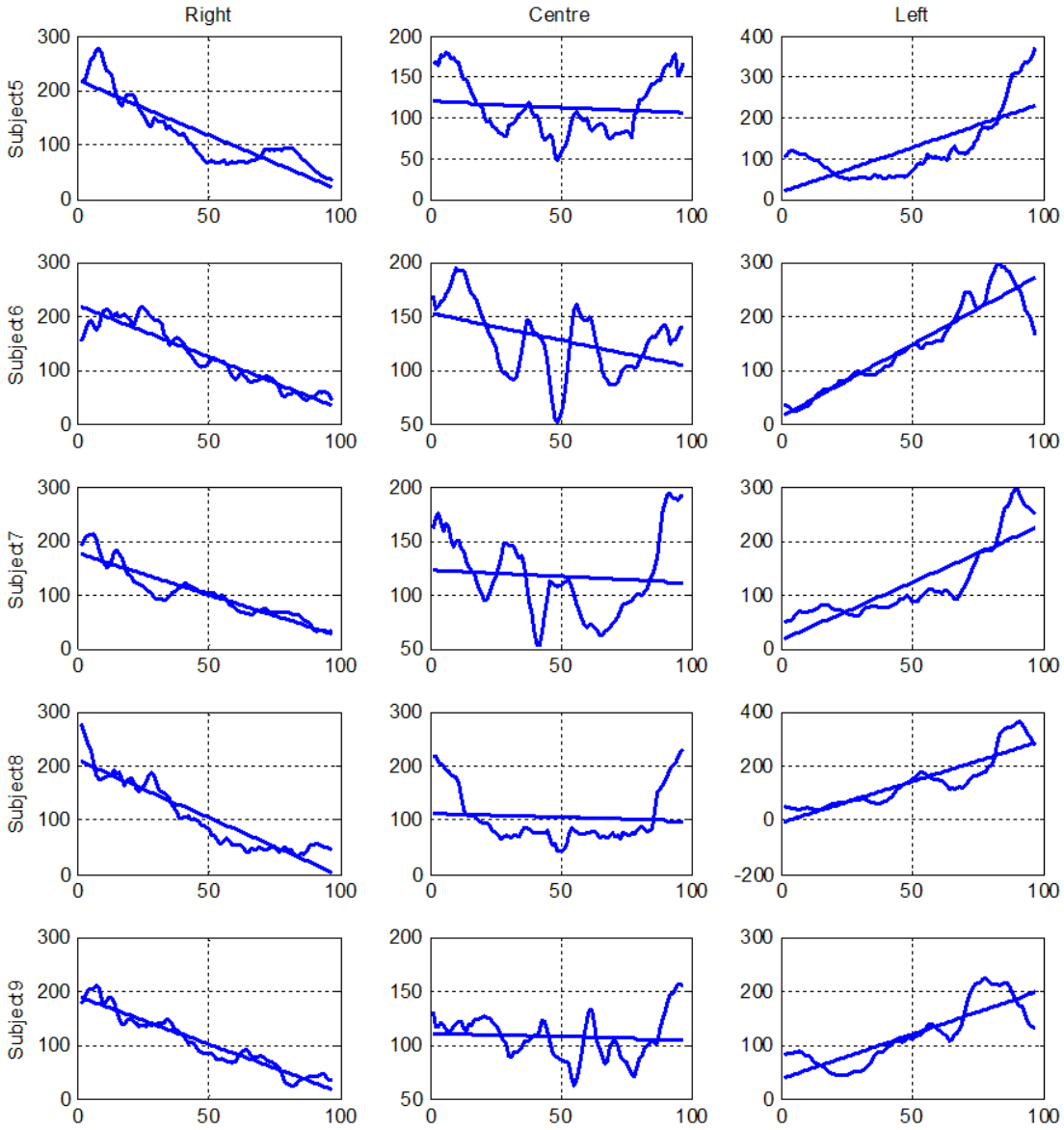

Figure 3-9: Lines approximating symmetry curves for faces in Figure 3-6

3.3.4 Single frame head pose classification

Two approaches are used to classify the heads' different positions into classes ω_1, ω_2, ω_3 corresponding to the centre, right and left position respectively:

Difference of means: Given a training set of symmetry curves associated with faces from each class (centre, right and left): The means μ_n of the symmetry curves' COGs and the *y*-intercepts of the lines approximating these symmetry curves are calculated for each training set. The difference between the COG/*y*-intercept *C* of the symmetry curve associated with the new face to be classified and the COG/*y*-intercept's mean for each class is obtained as

$$d_n = |C-\mu_n| \;\; \forall\, n = \{1,2,3\} \tag{3-21}$$

The decision rule *h* chooses the class *n* for which d_n is the smallest:

$$h(C) = \omega_m : m \;=\; arg\min_{n} \; (\{d_n\}_{n=\{1,2,3\}}) \tag{3-22}$$

Mean and standard deviation in a Gaussian distribution: Given a training set of symmetry curves associated with each class (centre, right and left): The means and the standard deviations of the symmetry curves' COGs/*y*-intercepts of the lines approximating these symmetry curves are calculated for each training set. Subsequently, they are each associated with a Gaussian distribution along with the symmetry curve's COG/*y*-intercept *C* resulting from the new face to be classified:

$$P_n = \frac{1}{\sqrt{2\pi}\sigma_n} exp\{-\frac{(C-\mu_n)^2}{2\sigma_n^2}\} \quad \forall\, n = \{1,2,3\} \tag{3-23}$$

The resulting highest probability measure corresponds to the class to which the given COG/*y*-intercept belongs: The decision rule *h* chooses the class ω_m for which P_n is the highest:

$$h(C) = \omega_m : m = \arg\max_n (\{P_n\}_{n=\{1,2,3\}}) \tag{3-24}$$

3.3.5 Head rotation detection: Head-based direction intent recognition

The task of intent recognition in the context of this work involves the detection of the direction that the subject intends to take looking at the motion of the head. In this section, the problem of monitoring the time sequence of individual positions of the head in rotation is addressed by looking at the sequence of the COG of the symmetry curves and the sequence of its *y*-intercepts obtained from a linear regression.

COG-based intention curve: Let $E = \{I_i : I_i$ is the i^{th} frame in a sequence of $N = 10$ frames$\}$: For each image frame in *E* the symmetry curve and its COG are obtained using Equations 3-17 and 3-18 respectively to form the intention curve:

$$\left\{ V(i) = \frac{\sum_{j=1}^{n} x_{ij} f(x_{ij})}{\sum_{j=1}^{n} f(x_{ij})} \right\}_{i=\{1,\ldots,10\}} \tag{3-25}$$

$\forall$ pixel column x_i in face images I_i belonging to a 10-frame video sequence and where *n* is the length of the symmetry curve. Figure 3-10 (Part a) displays the

resulting intention curves for the three different motions indicating that they exhibit separable patterns.

Y-intercept-based intention: Let $E = \{I_i : I_i$ is the i^{th} frame in a sequence of $N = 10$ frames$\}$: For each image frame in E the symmetry curve is obtained using Equation 3-17 and the y-intercept of the line approximating the symmetry curve is obtained using Equation 3-20 to form the intention curve:

$$\{V(i) = \beta_i(1)\}_{i=\{1,\ldots,10\}} \tag{3-26}$$

$\forall$ face images I_i in a 10-frame video sequence. Figure 3-10 (Part b) displays the resulting intention curves for the three different motions indicating that they exhibit separable patterns.

Note that the intention curves made of y-intercepts and those made of COG exhibit opposite patterns for left and right motion as illustrated in Figure 3-10.

Let $\{V_n(i)\}_{i=\{1,\ldots,10\}}$ be the intention curve (refer to Equations 3-25 and 3-26) in class ω_n $\forall n = \{1,2,3\}$ corresponding to the centre, right and left intentions respectively. The difference between these classes is determined by the constant, decreasing and increasing tendencies of the values in $\{V_n(i)\}_{i=\{1,\ldots,10\}}$ and is trained as follows:

$$\delta_n = \sum_{i=1}^{N-1} V_n(i) - V_n(i+1) \tag{3-27}$$

with N = 10 (the length of the intention curve), $V_n \in \omega_n$ and n = {1,2,3}. Let μ_n and σ_n be the statistics (means and standard deviations) of δ_n associated with the intention curves in the training set for class ω_n.

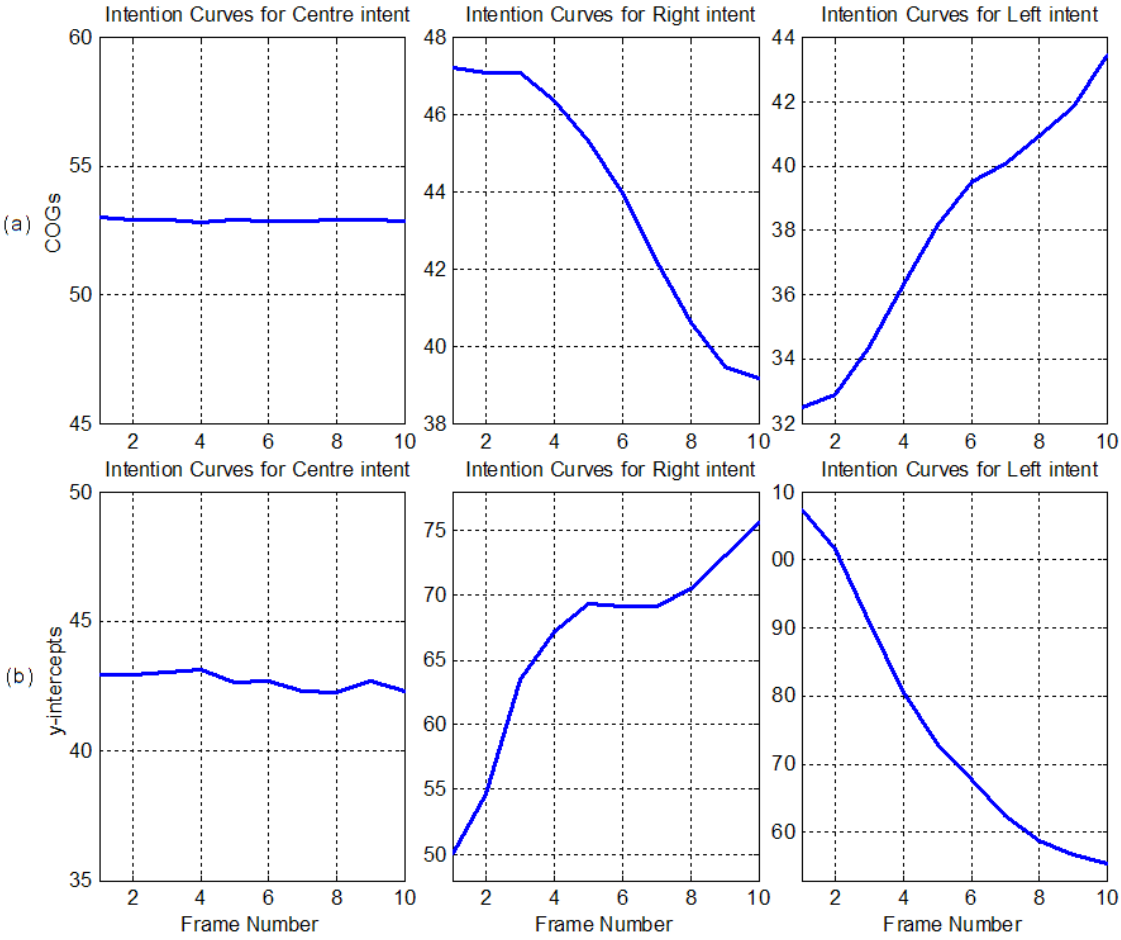

Figure 3-10: Intention curves based on COGs and *y*-intercepts

For a new intention curve $\{V(i)\}_{i=\{1,\ldots,10\}}$ to be classified, δ is obtained using Equation 3-27 and a decision rule h is defined in Equation 3-29 based on the "difference of means" approach (refer to Equation 3-28) and in Equation 3-31 based on the "statistics in Gaussian distribution" approach (refer to Equation 3-30).

$$d_n = |\delta - \mu_n| \quad \forall n = \{1,2,3\} \tag{3-28}$$

$$h(\{V(i)\}_{i=\{1,\ldots,10\}}) = \begin{cases} \omega_1,\ d_1 == min([d_n]_{n=\{1,2,3\}}) \\ \omega_2,\ d_2 == min([d_n]_{n=\{1,2,3\}}) \\ \omega_3,\ d_3 == min([d_n]_{n=\{1,2,3\}}) \end{cases} \quad (3\text{-}29)$$

$$P_n = \frac{1}{\sqrt{2\pi}\sigma_n} exp\{-\frac{(\delta-\mu_n)^2}{2\sigma_n^2}\} \quad (3\text{-}30)$$

$$h(\{V(i)\}_{i=\{1,\ldots,10\}}) = \begin{cases} \omega_1, P_1 == max([P_n]_{n=\{1,2,3\}}) \\ \omega_2, P_2 == max([P_n]_{n=\{1,2,3\}}) \\ \omega_3, P_3 == max([P_n]_{n=\{1,2,3\}}) \end{cases} \quad (3\text{-}31)$$

where $n = \{1,2,3\}$, ω_1, ω_2, ω_3 represent straight, right and left intents respectively. Note that if the straight motion intent is detected the next step is to determine the vertical motion of the head.

3.4 Recognition of head-based speed variation intent

The estimated position of the face in vertical motion as illustrated in Figure 3-11, is used as an intent indicator according to Table 3-3. The intents include moving at varied speeds (within the acceptable range for a wheelchair motion application) and stopping.

Table 3-3: Head motion and corresponding speed variation intention

Motion of the head	Inferred Intention
Down vertical motion	Increased speed intent
Up vertical motion	Decreased speed intent
No vertical motion (Centred position)	Intent to retain current speed

In a previous work [189], satisfactory results are obtained for the recognition of the vertical motion of carefully cropped faces from each frame of a video sequence using a symmetry-based approach, computed vertically, and where the resulting intention curves comprise the different positions of the symmetry curve's COG associated with the face as it moves through each frame of the video sequence. The results are however less convincing when performing the pre-processing steps of face detection and tracking.

Figure 3-11: Frontal view of the head (face) in vertical motion

In this thesis, the proposed solution makes use of another layer of PCA, where instead of using it for face detection and localisation in a skin colour detected region as described in Section 3.2.3, it is used to perform single vertical pose classification of the given faces into classes $\omega_m, \forall\, m = \{1,2,3\}$ corresponding to the centre, up and down position respectively. At the training stage three sets of *k* eigenfaces v_k^m (with m = {1,2,3}) are obtained from the training sets of $N \times N$ images I_m with the head in centre, up and down positions respectively: Figure 3-12 illustrates the 5 eigenfaces corresponding to the 5 highest eigenvalues of the covariance matrix $C_m = \frac{1}{M}\sum_{n=1}^{M} \Phi_{m,n}\Phi_{m,n}^T$, where Φ_m is the normalized $N^2 \times 1$ image in the m^{th} class and M is the number of example in the training set for each class. The weight vector associated with the mean image Ψ_m of each class is subsequently obtained as follows:

$$w_{m,k} = v_{m,k}^T \Psi_m, \qquad \forall\ m = \{1,2,3\} \tag{3-32}$$

where Ψ_m is given by Equation 3-7, $k = \{1,...,M'\}$ with M' = 10, and the weight vector representing the mean face for each class is given by:

$$\Omega_m = [\, w_{m,1}\ \ w_{m,2}\ \ ...\ w_{m,M'}\,]^T \tag{3-33}$$

Given a new face Φ whose vertical position must be classified, it is projected into the three eigenspaces associated with the three classes and the weight vectors Ω^m representing the new image face projected on the eigenspace associated with each class comprise the following weights:

$$w_k^m = v_{m,k}^T \Phi, \quad \forall\ m = \{1,2,3\} \tag{3-34}$$

Figure 3-12: Examples of eigenfaces for up, centre and down positions

The class ω_m is chosen to be the detected pose of the face for the m where the difference below (refer to Equation 3-35) is the lowest and below the threshold λ, that is:

$$d_m = \min_m \left\| \Omega^m - \Omega_m \right\| < \lambda \tag{3-35}$$

where Ω^m and Ω_m are the weight vector representing the new face image to be classified and the mean image for each class respectively, both projected on the eigenspace associated with class m, and the threshold $\lambda = 0.8$.

For intent recognition, let $\{d_m(i), \forall m = 1,2,3\}_{i=\{1,\ldots,10\}}$ be the set of intention curves (refer to Figure 3-13) made of sequences (made of 10 frames) of error measures based on Euclidean distance of a the given head image through the sequence $\Omega_i^m \; \forall i = \{1,\ldots,10\}$ and a generic face example Ω_m in a centred, up and down position of the head respectively.

$$\left\{ d_m(i) = \left\| \Omega_i^m - \Omega_m \right\| \right\}_{i=\{1,\ldots,10\}} \tag{3-36}$$

This set is used for classification into $\omega_1,\ldots,\omega_5$ corresponding to the centre, up (from centre-up), down (from centre-down), down (from up-centre) and up (from down-centre) intentions respectively, depending on their constant, decreasing and increasing values for $i = \{1,\ldots,10\}$.

The patterns of these set of n intention curves associated with each of the m classes is achieved as follows:

$$\delta_m = \sum_{i=1}^{N-1} d_m(i) - d_m(i+1) \tag{3-37}$$

with $N = 10$ (the length of the intention curve), $\{d_m, \forall m = 1,2,3\} \in \omega_m$. δ_m ($\forall m = 1,2,3$) is the tendency associated with the intention curves for class ω_m, obtained using sequences (made of 10 frames) of error measures based on the Euclidean distance between the mean face in class m and the given face to be classified projected on the eigenspace associated with vertical position associated with class ω_m.

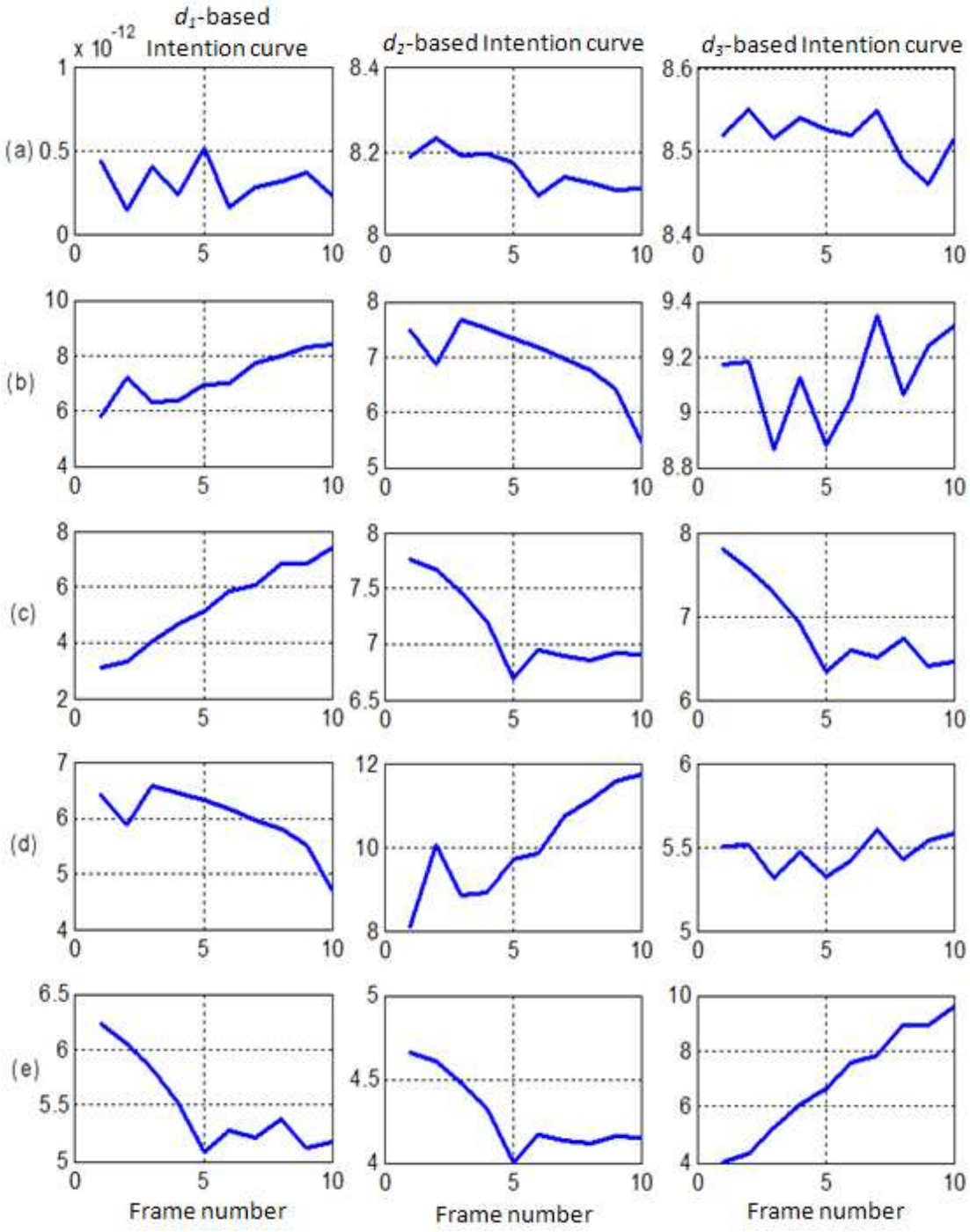

Figure 3-13: Intention curves based on distance measures d_1, d_2 *and* d_3

The classification of these intention curves is performed with the decision rule *h* defined below:

$$h(\{ d_m(i), \forall m = 1,2,3 \}_{i=\{1,..,10\}}) = \begin{cases} \omega_1, |\delta_1| \leq \lambda \wedge d_1(i) == min([d_n(i)]_{n=\{1,2,3\}}), \\ \omega_2, \delta_1 < \lambda \wedge \delta_2 > \lambda \wedge d_2(i) == min([d_n(i)]_{n=\{1,2,3\}}) \\ \omega_3, \delta_1 < \lambda \wedge \delta_3 > \lambda \wedge d_3(i) == min([d_n(i)]_{n=\{1,2,3\}}) \\ \omega_4, \delta_1 > \lambda \wedge \delta_2 < \lambda \wedge d_1(i) == min([d_n(i)]_{n=\{1,2,3\}}) \\ \omega_5, \delta_1 > \lambda \wedge \delta_3 < \lambda \wedge d_1(i) == min([d_n(i)]_{n=\{1,2,3\}}) \end{cases} \quad (3\text{-}38)$$

$\forall i = \{1,\ldots,10\}$ and $\lambda \geq 0$

3.5 Adaboost for head-based direction and speed variation recognition

To emphasise the merit of the proposed approach, an algorithm developed by Jia and Hu [66], [67] for an application similar to the one proposed in this thesis, is implemented in order to compare the results. The aim of their method is the detection, tracking, and recognition of the direction and the vertical position of human faces, which is intended to be used as a human-robot interaction interface for an intelligent wheelchair. Adaboost [190] is used for face detection and in subsequent frames camshift is used for tracking. A layer of adaboost is subsequently applied inside the comparatively small window, which is slightly bigger than the camshift tracking window, so that the precise position and direction (frontal, profile left or profile right)

of the face can be obtained rapidly. If the frontal (centred) face is detected, template matching [191] is used to indicate the nose position, and therefore to classify the centre, up and down positions for speed variation recognition.

3.5.1 Adaboost face detection

In Section 3.2.2, the adaboost algorithm is used to train skin colour as an alternative to classical threshold techniques as it performs more robustly especially, under poor or strong lighting conditions. In this section, however the classical adaboost algorithm for face detection proposed by Viola and Jones [190] and adapted by Jia and Hu [66], [67], is described: The adaboost algorithm uses a training set: $(x_1, y_1),\ldots,(x_n, y_n)$, where $x_i \in X$ (the domain, which in our case represents sample examples of the object of interest, that is, the face, and sample examples of non-faces) and $y_i \in Y$ (a class label set whose elements 0 or 1 indicate the category of the sample examples being non-faces or faces respectively). It calls a weak learning algorithm repeatedly in a series of T = 10 rounds giving weights to the training sets and updating the weights of these sets each time by utilising data from the last run of the weak learner and current weights. Each weak classifier is given by

$$h_j(x_i) = \begin{cases} 1 & \text{if} \quad p_j f_j(x_i) < p_j \theta_j \\ 0 & \text{otherwise} \end{cases} \tag{3-39}$$

where the polarity p_j indicates the direction of the inequality sign. At each round of boosting, the best weak classifier h_t with the lowest error ε_t is chosen.

Table 3-4: The adaboost algorithm [190]

- Given examples: $(x_i,y_i), \forall\, i = \{1,\ldots,N\}$, where x_i is the i^{th} example and y_i is its associated class label:

$$y_i = \begin{cases} 0, \textit{ for negative examples} \\ 1, \textit{ for positive examples} \end{cases}$$

- Initialise weights:

$$w_{1,i} = \frac{1}{N},$$

where N is the number training examples

- For $t = 1, \ldots, T$
  - Normalise the weights:

$$w_{t,i} = \frac{w_{t,i}}{\sum_{k=1}^{N} w_{t,k}}, \text{ so that } w_{t,i} \text{ is a probability distribution}$$

  - For each feature f_i train a classifier h_j: The error is evaluated with respect to the weight $w_{t,i}$:

$$\varepsilon_j = \sum_i w_{t,i} \left| h_j(p) - 1 \right|,$$

$$\text{where} \quad h_j(x_i) = \begin{cases} 0 & p_j f_j(x_i) < p_j \theta_j \\ 1 & \textit{otherwise} \end{cases},$$

$$\beta_t = \frac{\varepsilon_t}{1-\varepsilon_t}$$

  - Update the weights:

$$w_{t+1,i} = \begin{cases} w_{t,i} \times \frac{\varepsilon_t}{1-\varepsilon_t} & \textit{if} \quad h_t(x_i) = y_i \\ w_{t,i} & \textit{otherwise} \end{cases}$$

- The final strong classifier is obtained by combining all the selected weak classifiers h_t with $\alpha_t = log_2 \frac{1-\varepsilon_t}{\varepsilon_t}$ as:

$$H(x) = \begin{cases} 1 & \textit{if } \sum_{t=1}^{T} \alpha_t h_t(p) \geq \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\ 0 & \textit{otherwise} \end{cases}$$

For Haar like feature selection, instead of using the standard rectangle features used by Viola and Jones [190], a modified and enriched representation of the rectangle feature is used that includes edge features, centre-surround features, and line features as depicted in Figure 3-14. Fast feature computation is made possible by the use of integral images and their variant: Rotated Integral Image, Integral Rectangle, Rotated Integral Rectangle initially proposed in [192] and illustrated in Figure 3-15. For classification given a new image containing a face, a cascade of 5 strong classifiers trained with adaboost in a degenerated decision tree structure is used as represented in Figure 3-16.

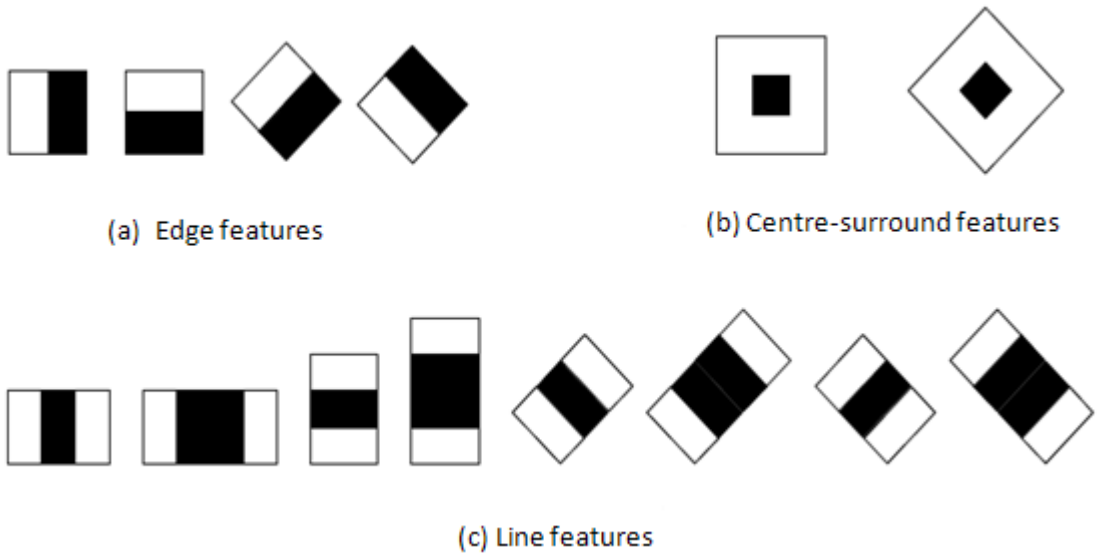

Figure 3-14: Rectangle features [66]

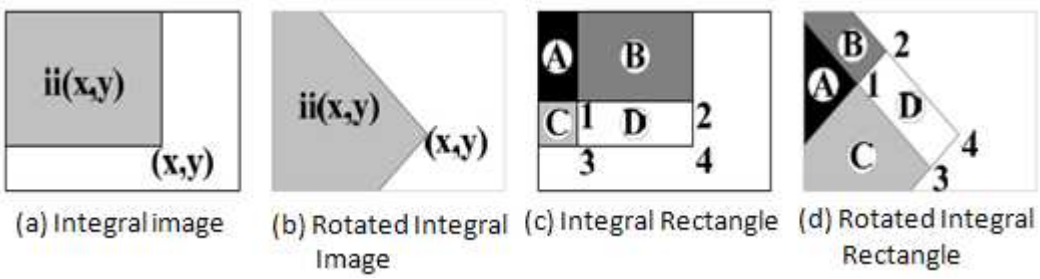

Figure 3-15: Integral Image and Integral Rectangle [66]

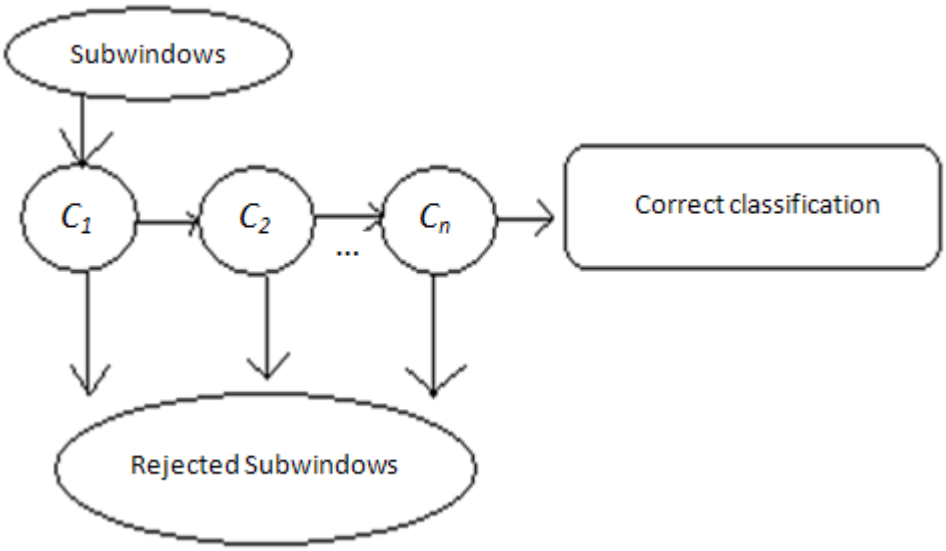

Figure 3-16: Cascade of $n = 5$ adaboost trained strong classifiers

3.5.2 Camshift tracking

Camshift stands for "Continuously Adaptive Mean Shift", which was introduced by Bradsky [193], [194] in 1998. It combines the basic Mean Shift algorithm with an adaptive region-sizing step using a kernel which is a simple step function applied to a skin colour probability map. Colour is represented as the Hue component from the HSV colour space. Since the kernel is a step function, the mean shift at each iteration is simply the average x and y of skin colour probability contributions within the current region. This is determined by dividing the first moments of the region by its zeroth moment at each iteration and shifting the region to the probability centroid as demonstrated in Table 3-5 that summarizes the steps of the camshift algorithm applied to the region $\Phi \subset I$ containing the detected face (resulting from the adaboost classification illustrated in Figure 3-16) in the input image frame I.

Table 3-5: Camshift Algorithm

1. *Let Φ be the search window: $\Psi \subset \Phi \subset I$ (Φ slightly larger than Ψ)*
2. *Choose centre of the initial location: $(x_c,y_c) \in \Phi$*
3. *Calculate Φ' the colour probability distribution in Φ $\forall (x,y) \in \Phi$.*
4. *Compute the mean location in Φ' using mean shift:*
   - *Find the zeroth moment:*
   $$M_{00} = \sum_x \sum_y \Phi'(x,y)$$
   - *Find the first moment for x and y:*
   $$M_{10} = \sum_x \sum_y x\Phi'(x,y);$$
   $$M_{01} = \sum_x \sum_y y\Phi'(x,y)$$
   - *The mean search window location (the centroid) is*
   $$x_c = \frac{M_{10}}{M_{00}};\ y_c = \frac{M_{01}}{M_{00}}$$
5. *Centre the new Φ' at (x_c,y_c)*
6. *Repeat Steps 4 and 5 until convergence (or until the mean location moves less than a preset threshold).*

The resulting centre (x_c,y_c) indicates the centre and therefore the window location of the face Ψ in the new frame. Another layer of adaboost is subsequently applied inside the comparatively small window Φ which is slightly bigger than the camshift tracking window, so that the precise face position and direction (frontal, profile left or profile right) can be determined. This is achieved by training frontal, left and right faces using adaboost for each class resulting in the weights α_t^i and the selected weak classifiers h_t^i, where i = {1,2,3} designate the frontal, left and right positions of the face respectively. Position classification is determined as follows:

$$p = arg \min_i d_i \tag{3-40}$$

$$d_i = \sum_{t=1}^{T} \left| \alpha_t^i h_t^i(\Psi) - \frac{1}{2}\alpha_t^i \right|, \quad \forall i = \{1,2,3\} \tag{3-41}$$

If the frontal face is detected, that is $p = 1$, nose template matching as described below (refer to Section 3.5.3) is used to indicate the vertical position of the head.

3.5.3 Nose template matching

For nose template matching a normalised cross-correlation template matching [191] is implemented: Given a face image Ψ of size $X \times Y$, let g_i be an $m \times n$ nose template of a face in vertical position $i = \{1,2,3\}$ corresponding to the centre, up and down position respectively. This template's instance must be detected in the face image Ψ: The obvious approach is to place the template at a location in an image and to detect its presence at that point by comparing intensity values in the template with the corresponding values in the image. Since in practice it is rare that the intensity values will match exactly, the criterion that the match should be perfect is unrealistic. As an alternative, a measure of dissimilarity between n the intensity values of the template and the corresponding values of the image can be used. The most popular measure, the sum of the squared errors is employed:

$$M = \sum_m \sum_n (\Psi - g_i)^2 \tag{3-42}$$

This measure can be computed indirectly and the computational cost can be reduced as follows:

$$M = \sum_m \sum_n (\Psi^2 - 2\Psi g_i + {g_i}^2)$$

$$= \sum_m \sum_n \Psi^2 - 2\sum_m \sum_n \Psi g_i + \sum_m \sum_n {g_i}^2 \qquad (3\text{-}43)$$

The latter expression clearly shows that the greater the middle term, the smaller the measure of dissimilarity and the more alike the image region with the template. This middle term can therefore be considered as the match measure between g_i and Ψ. Now, if we assume that Ψ and g are fixed, then $\sum_m \sum_n \Psi g_i$ gives a good measure of a mismatch: the mismatch $Ma_i(x,y)$ between the nose template g_i and each region in the image Ψ is computed from pixel $\Psi(x,y)$ to pixel $\Psi(x+m,y+n)$, as

$$Ma_i(x,y) = \sum_{k=1}^{m} \sum_{l=1}^{n} g_i(k,l)\Psi(x+k, y+l) \qquad (3\text{-}44)$$

This operation is called the cross-correlation between Ψ and g_i. A minor problem in the above computation is that Ψ and g_i are assumed to be constant. When applying this computation to images, the template g is constant, but the value of Ψ will vary. The value of Ma_i will then depend on Ψ and hence will not give a correct indication of the match at different locations. This problem is solved using a normalised cross-correlation. The match measure $Ma_i(x,y)$ between the nose template g_i and each region in the image Ψ is computed from pixel $\Psi(x,y)$ to pixel $\Psi(x+m,y+n)$, as

$$Ma_i(x,y) = \frac{\sum_{k=1}^{m}\sum_{l=1}^{n} g_i(k,l)\Psi(x+k,y+l)}{\sqrt{\sum_{k=1}^{m}\sum_{l=1}^{n}\Psi^2(x+k,y+l)}\sqrt{\sum_{k=1}^{m}\sum_{l=1}^{n} g_i^2(k,l)}} \tag{3-45}$$

where k and l are the displacements with respect to the template in the image. To find the location of the nose in the face image, this normalised cross-correlation operation is performed throughout the given face image for each template g_1, g_2 and g_3 associated with classes ω_1, ω_2, ω_3, corresponding to the centre, up and down positions respectively. The highest matching measure M_i for each template is obtained:

$$\{M_i = \max\,[Ma_i(x,y)]\}_{i=\{1,2,3\}} \;\forall x \in X \text{ and } y \in Y \tag{3-46}$$

Each of the highest matches M_i is subsequently classified using h to determine the vertical position of the face where again the highest match is chosen:

$$h(\Psi) = \omega_m : m = \arg\max_i \left(\{M_i\}_{i=\{1,2,3\}}\right) \tag{3-47}$$

The best match in the image is the highest value. Since this highest value criterion is not sufficient in cases where an image does not contain the object of interest, this highest value should also be above a certain threshold $\lambda = 2.5$ obtained empirically to indicate a match, that is, $max(\{M_i\}_{i=\{1,2,3\}}) \geq \lambda$. Note that these three nose templates are acquired from a single subject in the training set.

Table 3-6 provides a summary of the algorithm by Jia and Hu [66]. Figure 3-17 illustrates face detection using adaboost (Part a) and nose localisation (Part b) through a normalised cross-correlation template matching on the first frame in the video sequences of four different subjects. It must be noted that unlike the proposed approach in this thesis that looks at the entirety of the motion in order to make a decision, this approach looks at a single frame pose of the head to decide on the intention to move the wheelchair in a certain direction or varying its speed. The advantage of this method is that only one frame is used for intent recognition making it data efficient and fast. The disadvantage however is that when the head is in the left going to the right, the intention will remain left as long as the face is on the left side, while our proposed method makes provision for such back motions. A modified version of the approach by Jia and Hu is hence proposed to address this back motion problem.

Table 3-6: Head Gesture (*Tracked Face*) [66]

if *Frontal face is detected* **then**
  Keep Straight
  Nose template matching for speed intent recognition
else if *Only profile left face is detected* **then**
  Turn Left
else if *Only profile right face is detected* **then**
  Turn Right
else if *Both profile left/right faces are detected* **then**
    if *left size > right size* **then**
      Turn Left
    else if (*left size < right size*) **then**
      Turn Right
    else then
        Keep Straight

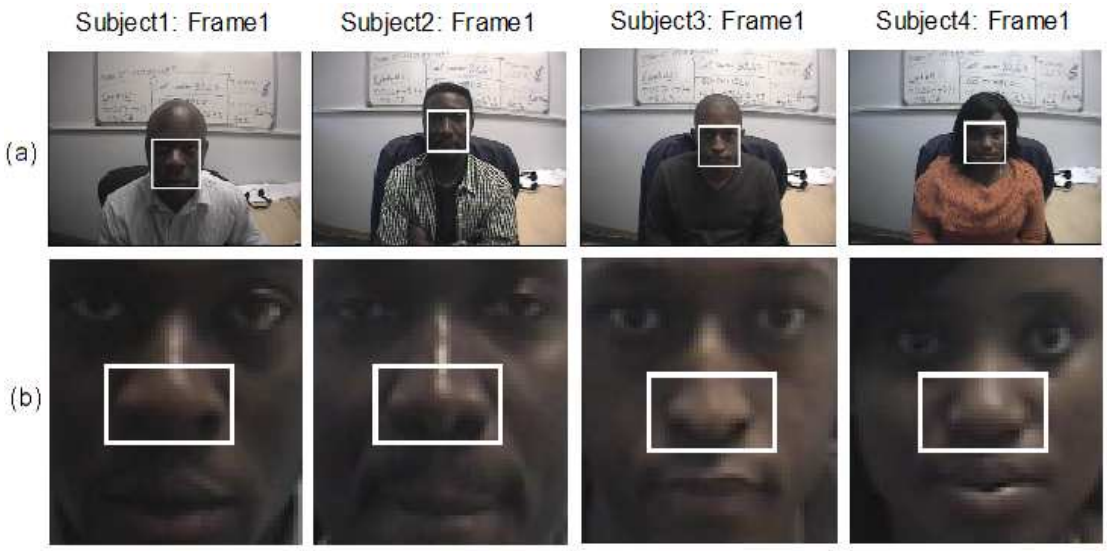

Figure 3-17: Adaboost face detection and nose template matching

For direction intent recognition through head rotation, intention curves of each motion are represented using changes in the difference between the linear combination of weighted weak classifiers for a given face and the resulting thresholds from adaboost learning associated with each class. As depicted in Table 3-4, the strong classifier trained by adaboost and used to distinguish between frontal, profile right and profile left is given by:

$$H(\Psi) = \begin{cases} 1 & if \sum_{t=1}^{T} \alpha_t h_t(\Psi) \geq \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\ 0 & otherwise \end{cases} \quad (3\text{-}48)$$

with Ψ a new instance (a face) to be classified as face or non-face. It was empirically established that the closer the two terms forming the inequality, the closer the instance *x* is to the examples in the training set.

Let $\{d_m(i), \forall\, m = 1,2,3\}_{i=\{1,\ldots,10\}}$ be the set of intention curves each composed of the difference between the linear combination of weighted weak classifiers used to classify each frame Ψ_i in sequence $\{\Psi_i\}_{i=\{1,\ldots,10\}}$ for a given face and the resulting thresholds from adaboost learning associated with each class:

$$\left\{ d_m(i) = \sum_{t=1}^{T} (\alpha_{m,t} h_t(\Psi_i) - \frac{1}{2}\alpha_{m,t}) \right\}_{i=\{1,\ldots,10\}} \tag{3-49}$$

As illustrated in Figure 3-18 a set of intention curves $\{d_m(i), \forall\, m = 1,2,3\}_{\,i\,=\,\{1,\ldots,10\}}$ is obtained for each class $\omega_1,\ldots,\omega_5$, corresponding to the centre, right (from centre-right), left (from centre-left), left (from right-centre) and right (from left-centre) intentions respectively. For a new input sequence, a set of intention curves $\{d_m(i), \forall\, m = 1,2,3\}_{i=\{1,\ldots,10\}}$ is obtained using Equation 3-49, δ_m is obtained using Equation 3-37 and classification is performed using Equation 3-38. In case of a centre intention, the next step is to determine the vertical motion of the head.

For speed variation intent recognition through vertical motion of the head, the intention curves of each motion are represented by the changes in template matching measures between the detected nose and the nose templates associated with a centre, an up and a down position of the head: Let $\{M_m(i), \forall\, m = 1,2,3\}_{i=\{1,\ldots,10\}}$ be the set of intention curves each composed of 10 matching measures (refer to Equation 3-45) of the given nose in the sequence of face images to be classified, with a centred nose template, an up nose template and a down nose template respectively. These sets of m intention curves $\{M_m(i), \forall\, m = 1,2,3\}_{i=\{1,\ldots,10\}}$ portrayed in Figure 3-19 exhibit different patterns for each of the intent classes $\omega_1,\ldots, \omega_5$, corresponding to the centre,

up (from centre- up), down (from centre-down), down (from up-centre) and up (from down-centre) intentions respectively (refer to Figure 3-19), and can therefore be classified using an appropriate decision rule.

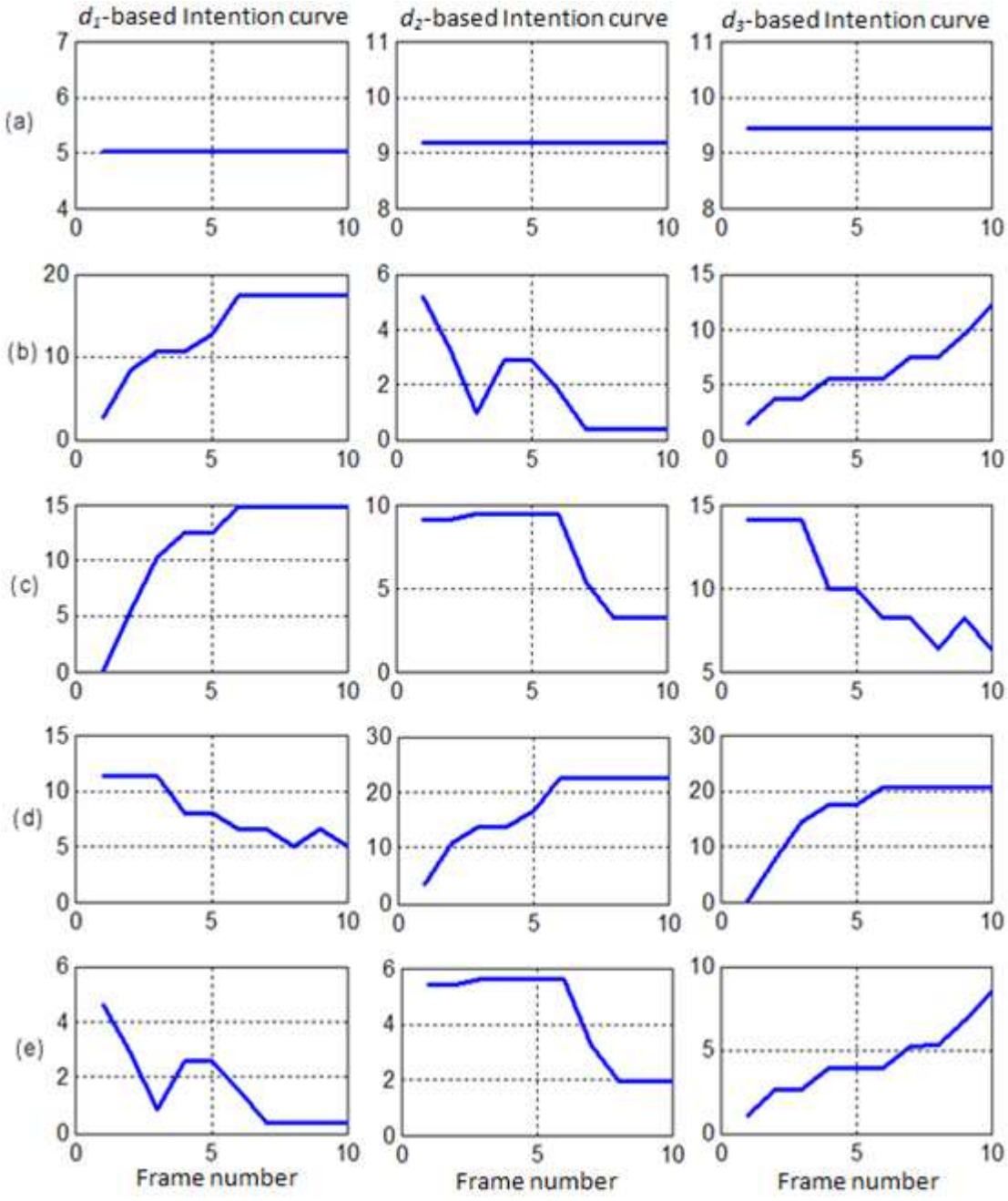

Figure 3-18: Intention curves based on differences d_1, d_2 and d_3

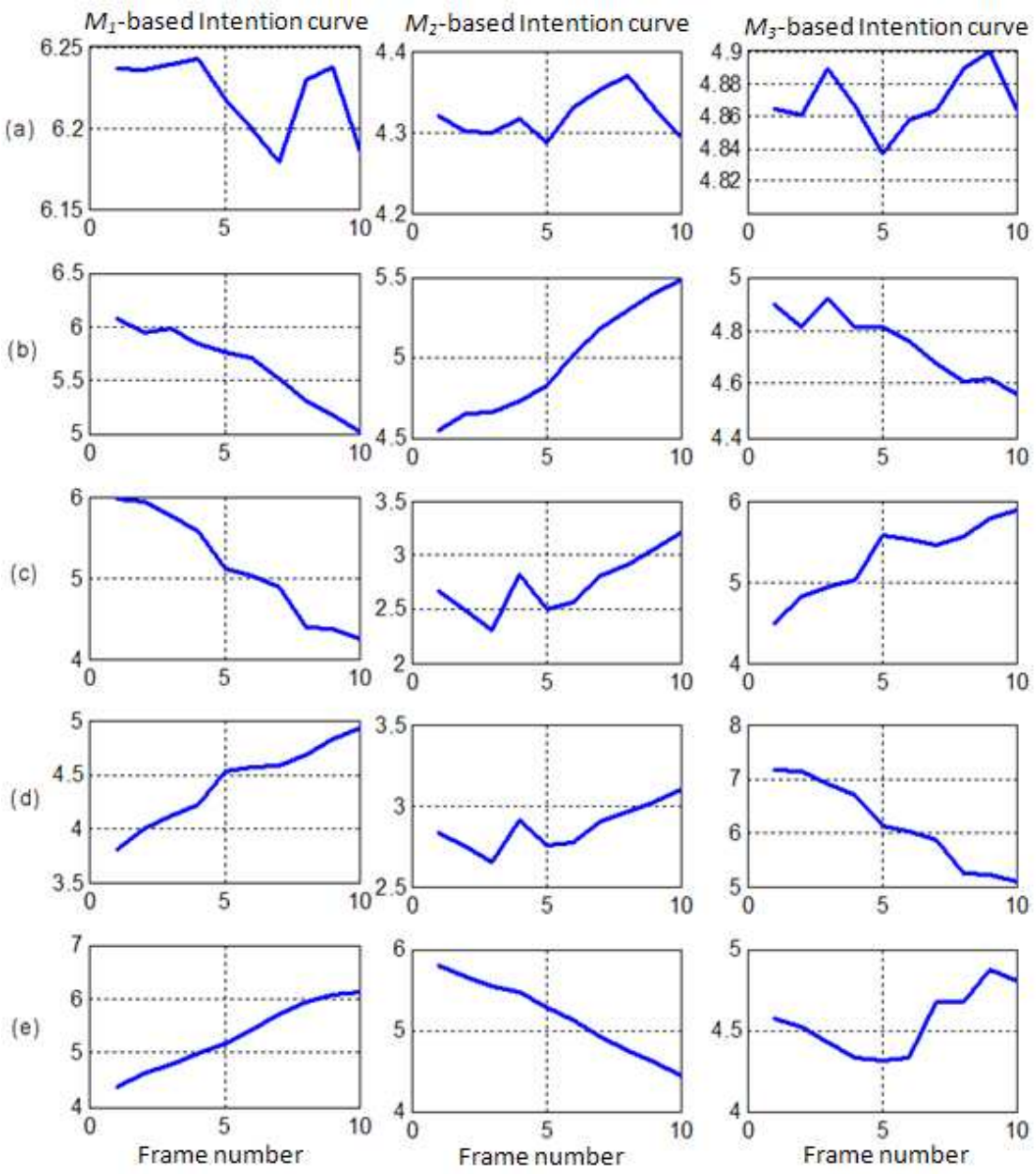

Figure 3-19: Intention curves based on matching measures M_1, M_2 and M_3

For a new input sequence, a set of intention curves $\{M_m(i), \forall\, m = 1,2,3\}_{i=\{1,\ldots,10\}}$ is obtained using Equation 3-45 and 3-46, δ_m is obtained using Equation 3-37 where $d_m = M_m$, and classification is performed with the decision rule h defined in Equation 3-50.

$$h(\{M_m(i), \forall m = 1,2,3\}_{i=\{1,\ldots,10\}}) = \begin{cases} \omega_1, |\delta_1| \leq \lambda \wedge M_1(i) == max([M_n(i)]_{n=\{1,2,3\}}) \\ \omega_2, \delta_1 > \lambda \wedge \delta_2 < \lambda \wedge M_2(i) == max([M_n(i)]_{n=\{1,2,3\}}) \\ \omega_3, \delta_1 > \lambda \wedge \delta_3 < \lambda \wedge M_3(i) == max([M_n(i)]_{n=\{1,2,3\}}) \\ \omega_4, \delta_1 < \lambda \wedge \delta_2 > \lambda \wedge M_1(i) == max([M_n(i)]_{n=\{1,2,3\}}) \\ \omega_5, \delta_1 < \lambda \wedge \delta_3 > \lambda \wedge M_1(i) == max([M_n(i)]_{n=\{1,2,3\}}) \end{cases} \quad (3\text{-}50)$$

$\forall i = \{1,\ldots,10\}$ and $\lambda \geq 0$

3.6 Conclusion

In summary, this chapter offers a detailed description of the algorithms proposed in this work aimed at visual head-based motion detection for intent recognition. The pre-processing steps (detection and tracking of the face) are implemented using skin colour detection, some image processing operations (erosion, dilation and connected components labelling) and PCA on the resulting skin colour region. The overview of the intent recognition algorithms consists of using a 10-frame video sequence as input that is mapped to a 10-point intention curve that presents separable patterns for each intention.

For direction intent recognition, a symmetry-based approach is used where the COGs and the *y*-intercepts of the resulting symmetry curves throughout the sequence form the intention curve. For speed variation intent recognition, a PCA-based algorithm is employed where the varying error distances throughout the sequence form the

intention curve. The appropriate decision rules are subsequently used to classify these intention curves for intent recognition.

The algorithm developed by Jim and Hu [66], [67] based on adaboost, camshift and nose template matching also aimed at detecting faces in rotation and vertical motion is implemented and compared with the solutions proposed in this work.

The next chapter discusses the solutions proposed for recognition of the hand in rotation and vertical motion and Chapter 5 (Section 5.2) discusses the results.

Chapter 4

Hand-Based Intent Recognition

4.1 Introduction

The second visual intent indicator used in this work is the hand in free motion without the constraint of manoeuvring a joystick. The aim is to provide an alternative to the joystick, where the hand is more flexible, especially in scenarios where the disabilities did not impair the hand, but at the same time where manoeuvring the joystick is a difficult task. The solution therefore requires a camera with the dorsal view of the hand as the object of interest in its field of view. The hand performs two types of motions, rotation and vertical motion, to indicate an intention in direction and speed variation respectively. Hand rotation in a particular direction (right or left) is selected to indicate the chosen direction that the subject intends to take, while vertical hand motion (up or down) is chosen to indicate the subject's speed variation intent where the hand going up is chosen to indicate a decrease in speed.

The proposed visual solution accepts a video sequence as input, with the hand in rotation and vertical motion as the object of interest and gives direction and speed variation intent respectively as output. Intent recognition is achieved by analysing the motion of the hand through the video sequence rather than looking at a single frame and 10 frames are used as input to the proposed algorithm and mapped to an intention curve.

This chapter furnishes a detailed description of the hand-based intent recognition methods proposed in this thesis. The pre-processing steps of detection and tracking of the dorsal view of the hand within a sequence are implemented using the same skin colour detection schemes used in Section 3.2 for face detection and tracking.

Skin colour detection combined with a prior knowledge of the length of a typical hand is therefore sufficient for hand segmentation. Similarly to the face, the tracking task only consists of repeating the detection task on a smaller region that is slightly bigger than the hand detected region of the previous frame. For hand-based direction recognition through hand rotation detection, a variant to the symmetry-based approach used in Section 3.3.1 for head rotation is employed to calculate the symmetry vertically rather than horizontally as previously. The statistics (mean and standard deviation) of the symmetry curves are used as 2D data features and three different machine learning methods (two supervised and one unsupervised) are employed for classification, namely, a neural network, a support vector machine, and *k*-means clustering. Another method is proposed based on a normalised cross-correlation template matching of the region in the hand containing the fingers, as previously implemented in Section 3.5.3 for nose detection to indicate the vertical motion of the head. For the vertical motion of the hand, the same template matching-based approach is implemented on the region of the hand containing the fingers and another proposed approach is based on the geometric constraints of the hand contour, where a mask in the shape of an ellipse is used to determine its vertical position. For comparison to the proposed methods for vertical motion of the hand, a feature selection found in the literature known as the Histogram of Oriented Gradient (HOG) [195] is implemented.

All these approaches result in intention curves, and appropriate decision rules are used to classify these intention curves for intent recognition.

To distinguish between the two different sets of motions, detection of the vertical motion of the hand for speed variation intent recognition is first performed. If no significant change in its vertical pose is observed, rotation detection of the hand for direction intent recognition is then performed. Note also that although for in this thesis the right hand is used, all these methods can be adapted for the use of the left hand.

4.2 Pre-processing steps: Hand detection and tracking

Section 3.2 furnishes the details of the skin colour detection approaches and the image processing operations used for hand detection except that the erosion operation is not required since very little noise is present in the binary image resulting from the skin colour detection process as depicted for the hands of the four subjects in Figures 4-1 (Part b) and 4-2 (Part b) for histogram-based and adaboost-based skin colour detection respectively. The dilation and the connected component labelling (refer to Section 3.2.3.2) are performed as illustrated in Figures 4-1 (Part c) and 4-2 (Part c) for histogram-based and adaboost-based skin colour detection respectively.

The assumptions for the proposed solution offer a field of view constraint such that only the hand and a part of the arm can be skin colour regions and where the camera is intended to be positioned at a fixed distance from the hand. A prior knowledge of the length of a typical hand is used for hand segmentation from the arm as follows:

Two points in the skin colour detected region are obtained at the right (assumed to be the tip of the fingers in the present application) and the left limit of the region: Let R be the skin colour detected region in the resulting binary image:

$$x_{max} = \max(x),\ x_{min} = \min(x),\ \forall\ (x,y) \in R, \tag{4-1}$$

$$y_1 = \max(y), \forall\ y \in R(x_{max},y) \tag{4-2}$$

$$y_2 = \min(y), \forall\ y \in R(x_{min},y) \tag{4-3}$$

These two points (x_{max},y_1) and $(x_{min},y_2) \in R$ are used to determine a line with angle θ with respect to the horizontal, which is considered parallel to the hand:

$$\theta = tan^{-1}\left(\frac{y_1 - y_2}{x_{max} - x_{min}}\right) \tag{4-4}$$

If the hand is not horizontal, its length is used as the hypotenuse of the right triangle with the horizontal and vertical lines as the other two sides. The segmentation is then performed using the right limit of the skin colour detected region as it is assumed to be the tip of the finger and therefore the end of the hand from which the fixed distance of the hand is measured to get the other end so as to keep the arm out of the region in the image containing the segmented hand: Let H_p = Length of the typical hand (30 pixels), the four points of the bounding box for detection are (x_{min},y_{min}); (x_{min},y_{max});

(x_{max},y_{max}) and (x_{max},y_{min}), where x_{max} is given above in Equation 4-1, and x_{min}, y_{max} and y_{min} are given below:

$$x_{min} = x_{max} - H_p \times \cos\theta \tag{4-5}$$

$$y_{max} = max(y) \text{ and } y_{min} = min(y), \ \forall y \in R \tag{4-6}$$

Figures 4-1 (Part d) and 4-2 (Part d) exhibit the resulting hand detection for four different subjects using the histogram-based and the adaboost-based skin colour detection respectively.

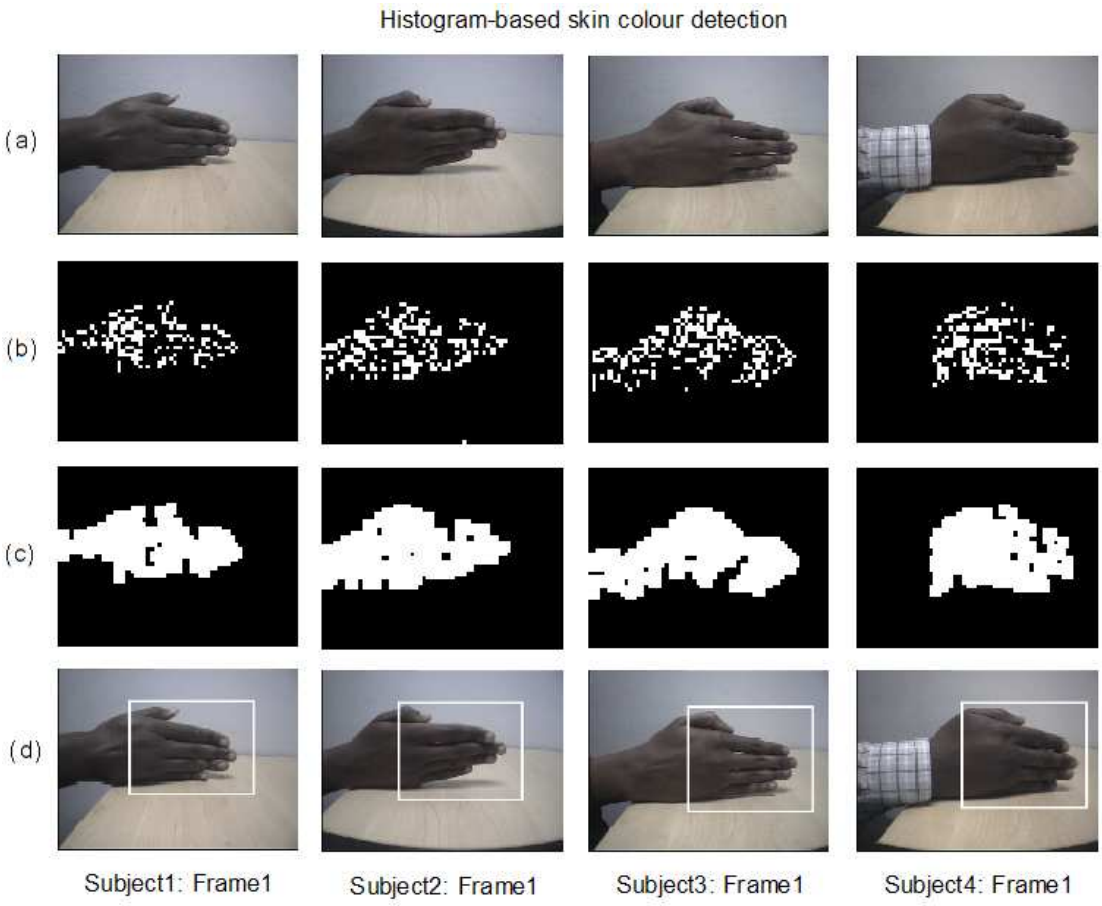

Figure 4-1: Hand detection using histogram-based skin colour detection

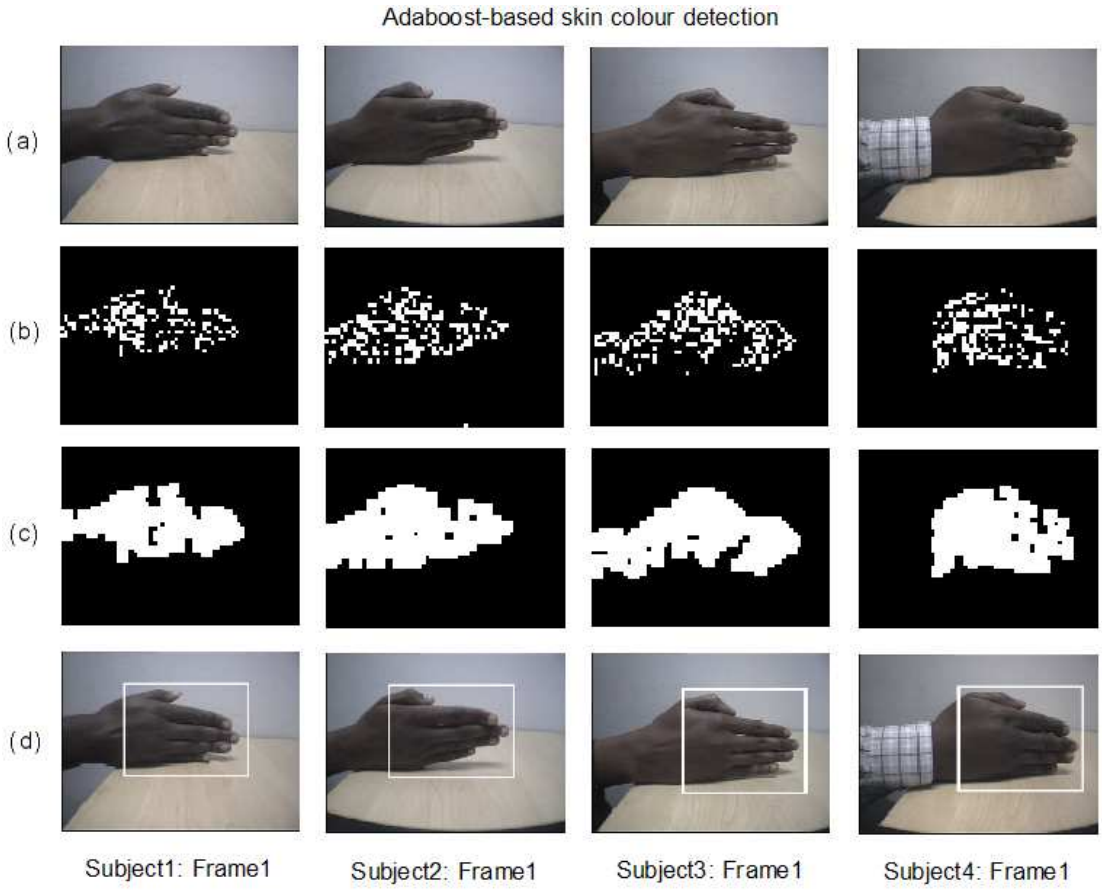

Figure 4-2: Hand detection using adaboost-based skin colour detection

4.3 Recognition of hand-based direction intent

The object of interest is the dorsal view of the detected hand in rotation (refer to Figures 4-1 (Part a) and 4-2 (Part a)) and the change in the estimated position of the hand over a sequence is used as an intent indicator, according to Table 4-1.

Table 4-1: Hand motion and corresponding direction intention

Motion of the hand	Inferred Intention
Left rotation	Intent to go left
Right rotation	Intent to go right
No rotation (Centred position)	Intent to go straight

4.3.1 Vertical symmetry-based direction intent recognition

The underlying assumption is that the dorsal view of a human hand, although not as symmetric as the face (refer to Section 3.3.1), exhibits separable symmetry properties for different positions (for the hand in rotation) when the symmetry is calculated vertically rather than horizontally: A particular symmetry signature is given when the hand is centred, and that symmetry signature changes when the hand is moved from this centred position. This gives the indication of a motion from the initial centred position to a new position (right or left). The symmetry curve is calculated vertically as follows:

$$f(y) = \sum_{w=1}^{k} \sum_{x=1}^{X} | I(x, y-w) - I(x, y+w) | \qquad (4\text{-}7)$$

The symmetry-value $f(y)$ is evaluated $\forall\, y \in [k+1 \; Y-k]$ where y is a pixel-row in the image, by taking the sum of the differences of two pixels at a variable distance w : $1 \leq w \leq k$ from it on both sides, making the pixel-row the centre of symmetry. This process is repeated for each column and the resulting symmetry-value is the summation of these differences. The symmetry curve is composed of these symmetry-values calculated for all the pixel-rows in interval $k+1 \leq y \leq Y-k$. It was empirically established that the value of the maximum distance k that gives more separable

symmetry curves among the different positions is given by k = 35. Figure 4-3 portrays the three different positions of the hand of four subjects and Figure 4-4 depicts their corresponding symmetry curves. For classification of these symmetry curves, two statistical features of the curves are used namely the means μ and standard deviations δ:

$$\mu = \frac{1}{N} \sum_{y=k+1}^{Y-k} f(y) \tag{4-8}$$

$$\delta = \sqrt{\frac{1}{N} \sum_{y=k+1}^{Y-k} (f(y) - \mu)^2} \tag{4-9}$$

where $N = (Y-k)-(k+1)$. Figure 4-5 illustrates the scatter plot of the feature points given by $x_i = (\mu_i, \sigma_i)$ $\forall\, i = \{1,2,3\}$, the symmetry curve's statistics for the three different categories ω_1, ω_2, ω_3 corresponding to the centre, right and left hand position respectively. Two are therefore sufficient to provide separable patterns for the vertical symmetry curves associated with hands in these different positions. Three machine learning approaches; a multilayer perceptron artificial neural network (a projection-based classification), a support vector machine (a kernel-based classification method) and *k*-means clustering; are used for single frame hand pose classification. The choice of these particular machine learning approaches simply aims at showing the merit of the proposed 'vertical symmetry curve' approach in extracting features that can be easily classified using different classification methods, supervised and unsupervised. For the choice of supervised approaches however, non-linear methods had to be used given the nature of the data that are not linearly separable, that is, there is no

discriminant function $g_i(x) = w_i^T x + w_{i,0}$ that separates the data as depicted in Figure 4-5. The MLP is a modification of the standard linear perceptron, which can distinguish data that is not linearly separable. It is a very flexible model, giving good performance on a wide range of problems in discrimination including the one at hand as revealed in Chapter 5. SVMs can produce accurate and robust classification results, even when input data are non-linearly separable as is the case in this work. To emphasize further the merit of the vertical symmetry-based approach, the statistics of the resulting symmetry curves as illustrated in Figure 4-5 can be approached as an unsupervised learning problem where the data points are assumed not to be labelled, and *k*-means clustering is used to label each data point.

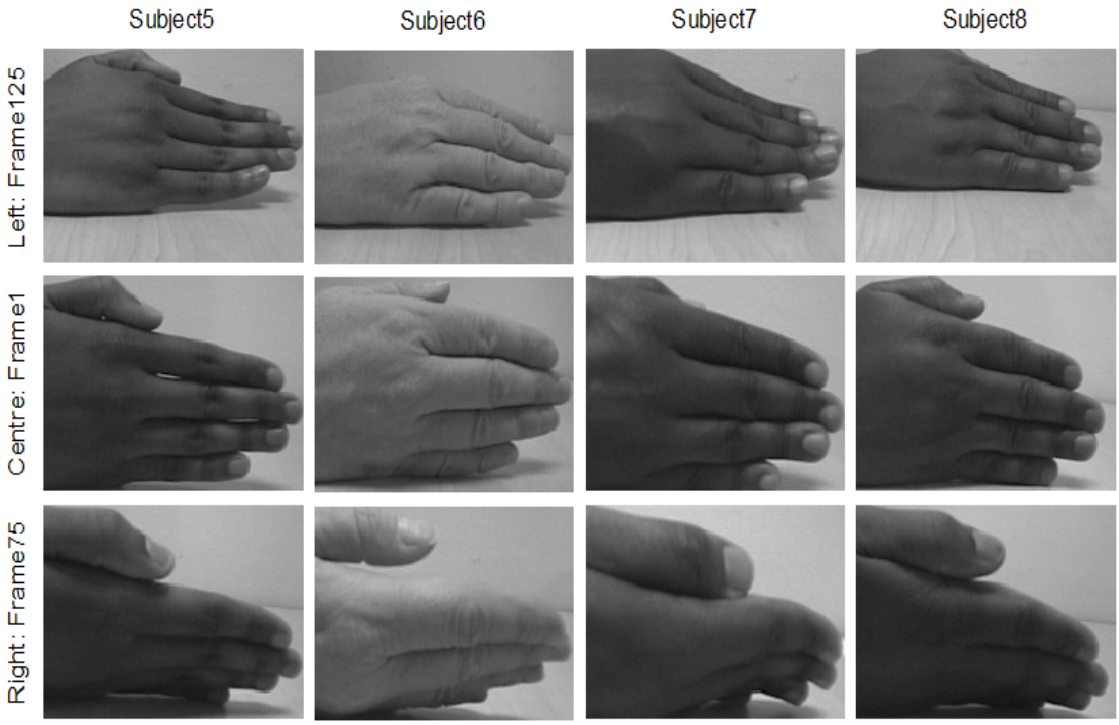

Figure 4-3: Three different positions of the hand (dorsal view) in rotation

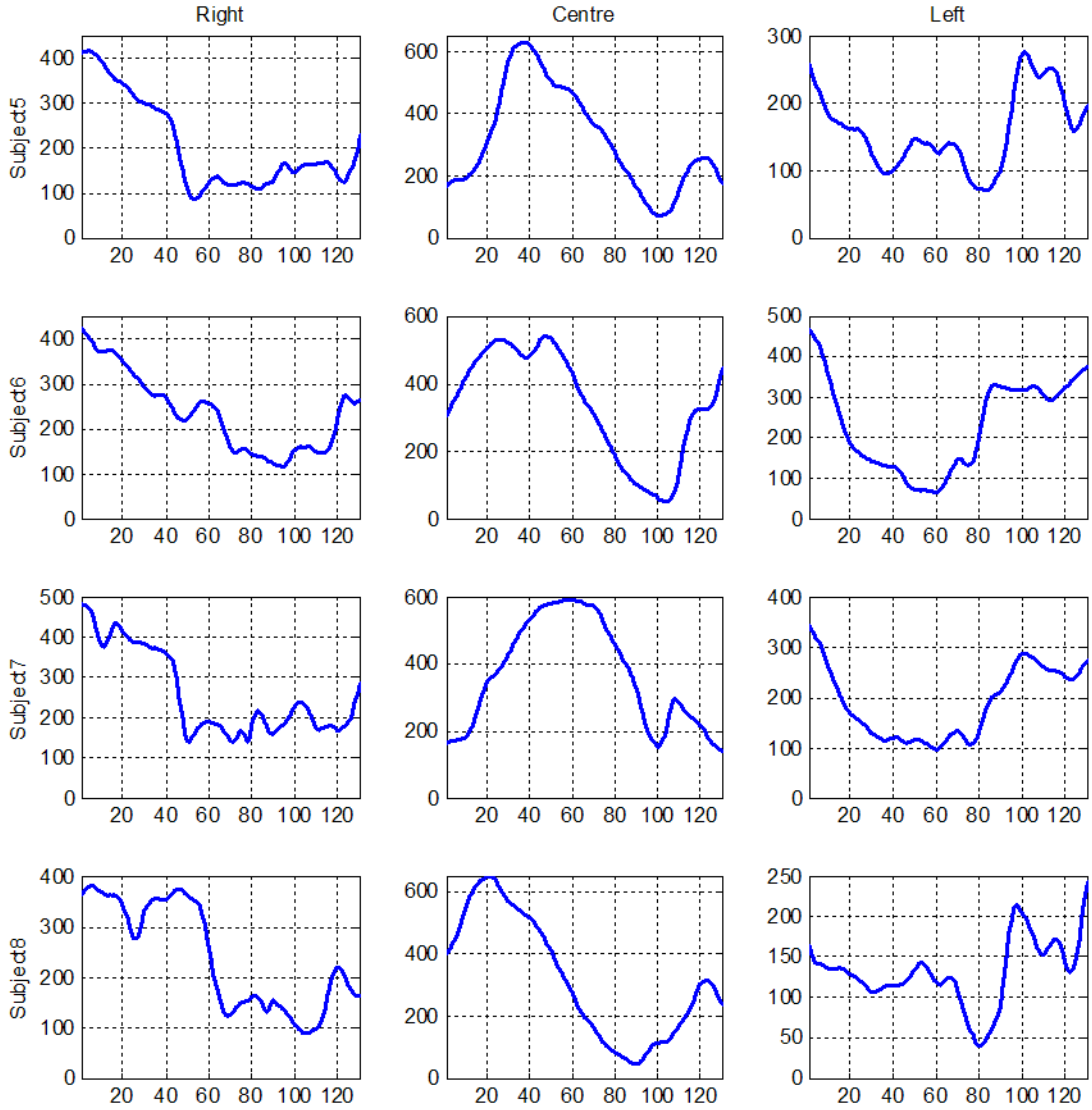

Figure 4-4: Symmetry curves corresponding to the hands in Figure 4-3

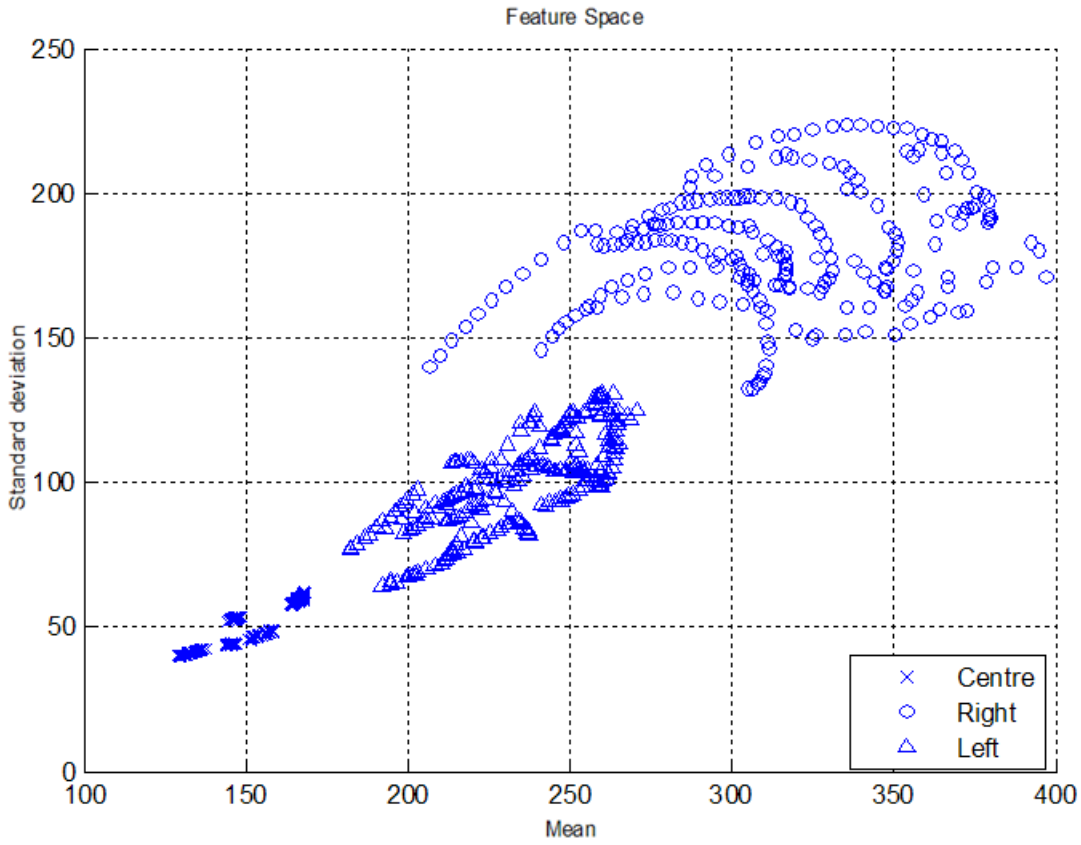

Figure 4-5: Features of different positions of the hand in rotation

4.3.2 Artificial Neural Networks (Multilayer Perceptron)

As a powerful data modelling tool, the neural network's ability to learn non-linear relationships [196] from data such as those portrayed in Figure 4-5 is used. A multilayer perceptron (MLP), which is a feedforward artificial neural network, produces a transformation of a pattern $\boldsymbol{x} \in R^k$ to a q-dimensional space according to Equation 4-10. From the empirical study conducted with the given data, the topology of the multilayer perceptron (MLP) is chosen to be a 3 layers perceptron, consisting of a 2 neurons input layer (k = 2), a 10 neurons hidden layer and a 3 neurons output layer (q = 3) as illustrated in Figure 4-6. Note that different authors refer to the above network as having either 3 layers according to the number of layers of neuron (input,

hidden and output), or 2 layers according to the number of layers of adaptive weights, and this work uses the former convention.

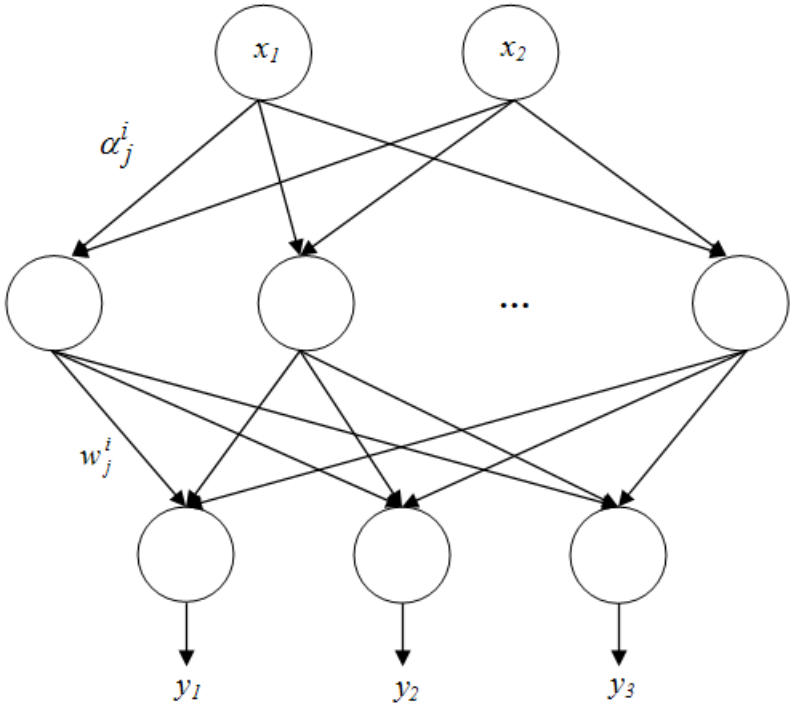

Figure 4-6: Multilayer perceptron

The training is performed using a back propagation algorithm. Given a labelled training set consisting of n data points $\boldsymbol{x_{ip}} = [\mu_{ip},\sigma_{ip}] \in \omega_j$, where $i = \{1,\ldots,n\}$, $p = \{1,2,3\}$ corresponding to the centre, right and left position respectively, and with their accompanying labels $\boldsymbol{y_p} = [y_{l,p}]_{l=\{1,2,3\}}$ where $y_{l,p} = \begin{cases} 1, & p = l \\ 0, & p \neq l \end{cases}$. The MLP produces the transformation $g(\boldsymbol{x})$ of a pattern $\boldsymbol{x} \in R^2$ to a 3-dimensional space according to

$$g_j(\boldsymbol{x}) = f\left(\sum_{k=1}^{m} w_j^k f\left(\sum_{i=1}^{n} \alpha_j^i x_i + \alpha_0^i\right) + w_j^0\right) \tag{4-10}$$

where $m = 2$ is the number of input neurons x_i from the previous layer, w_j^k is the weight associated with x_i, b is the offset from the origin of the feature space and f is the activation function chosen to be the sigmoid function:

$$f(z) = \frac{1}{1+e^{-z}} \tag{4-11}$$

The weights are updated using

$$w_j^i = w_j^i + \Delta w_j^i \tag{4-12}$$

where $\Delta w_j^i = -\eta \frac{\partial E^i}{\partial w_j^i}, E^i = \frac{1}{2}\sum_{k=1}^{N_k}(y_k - g_k)^2$ and y_k and g_k are the target and actual output of the network respectively and the weights α_j^i are updated in the same manner.

4.3.3 Support Vector Machines

Support Vector Machines (SVM) have become increasingly popular tools in data mining tasks such as regression, novelty detection and classification [197] and can therefore be used for the classification problem at hand: Given a labelled training set consisting of a set of data points $x_i = [\mu_i, \sigma_i]$ with their accompanying labels y_i and i = {1,2,3}, corresponding to the centre, right and left position respectively. The discriminant function is given by

$$g(x) = \langle x, w \rangle + b \tag{4-13}$$

where w and b are the weights (giving the shape of the hyperplane) and the offset from the origin respectively, and x is the data. The SVM can be considered as a tool for finding the optimal separating hyperplane for linearly separable data that can be extended to situations when the data are not linearly separable. For non-linear problem (refer to Figure 4-5), the kernel trick is used to construct the hyper plane that consists in mapping the data into a transformed feature space with a higher dimension, and to construct a linear classifier in that space [197], [198].

$$g(x) = w^T \phi(x) + b \tag{4-14}$$

where $\phi(x)$ is the transformation.

- $g(x) > 0 \Rightarrow x \in \omega_1$ represented by the numeric value $y_i = +1$
- $g(x) < 0 \Rightarrow x \in \omega_2$ represented by the numeric value $y_i = -1$

The SVM method determines the maximum margin solution through the maximisation of the dual form of the Lagrangian given by

$$L_D = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \phi^T(x_i) \phi(x_j) \tag{4-15}$$

where $y_i = \pm 1$ are class indicator values and α_i is the i^{th} Lagrange multiplier satisfying

- $0 \leq \alpha_i \leq C$ (for a regularisation parameter C)
- $\sum_{i=1}^{n} \alpha_i y_i = 0$

The value of *w* and *b* that maximises the margin between the hyperplane and the support vectors is obtained using

$$\arg\max_{w,b}(L_D) \tag{4-16}$$

yielding

$$w = \sum_{i=1}^{N} \alpha_i y_i \phi(x_i) \tag{4-17}$$

$$b = \frac{1}{N_{SV'}} \left\{ \sum_{i \in SV'} y_i - \sum_{i \in SV, j \in SV'} \alpha_i y_i \phi^T(x_i) \phi(x_j) \right\} \tag{4-18}$$

in which *SV* is the set of support vectors with associated values of α_i satisfying $0 < \alpha_i < C$ and *SV'* is the set of $N_{SV'}$ support vectors at the target distance of $\frac{1}{|w|}$ from the separating hyperplane, and where α_i is the i^{th} Lagrange multiplier and N_{SV} are the numbers of support vectors which are found to be 63, 253 and 122 for the centred class, the right class and the left class respectively. Classification of a new data sample *x* is performed according to the sign of *g*(*x*) given below:

$$g(x) = \sum_{i \in SV} \alpha_i y_i \phi^T(x_i) \phi(x) + b \tag{4-19}$$

To avoid computing the transformation $\phi(x)$ explicitly, a kernel function K can replace the scalar product:

$$K(x,y) = \phi^T(x)\phi(y) \tag{4-20}$$

Different types of kernel may be used in SVM. They must be expressible as an inner product in a feature space: A kernel $K(x,y)$ with $x, y \in R^p$, is an inner product in a feature space, that is $K(x,y) = \phi^T(x)\phi(y)$ if and only if

- $K(x,y) = K(y,x)$
- $\int K(x,y) f(x) f(z) dxdz \geq 0$

The polynomial kernel $K(x,y) = (1 + x^T y)^d$ is used and the discriminant function becomes

$$g(x) = \sum_{i \in SV} \alpha_i y_i K(x_i, x) + b$$

$$= \sum_{i \in SV} \alpha_i y_i (1 + x_i^T y)^d + b \tag{4-21}$$

where the degree of the polynomial kernel d = 1. Note that SVMs are binary classifiers, and therefore for the three class problem described in this thesis, a “one against one” decomposition of the binary classifiers is used. The “one against one” strategy, also known as “pairwise coupling”, “all pairs” or “round robin”, consists in constructing one SVM for each pair of classes. Thus, for a problem with c = 3 classes, $c(c\text{-}1)/2$ = 3 binary classifiers are trained to distinguish the samples of one class from

the samples of another class. The classification of an unknown pattern is therefore performed according to the maximum voting, where each SVM votes for one class.

4.3.4 K-means clustering

K-means is one of the simplest unsupervised learning algorithms that solve the well known clustering problem [199], [200]. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (three clusters in the case at hand namely: centre, right and left) fixed a priori. An objective function is used that expresses how good a representation is, and then an algorithm is constructed to obtain the best representation. To obtain the objective function given the three clusters, a centre is defined for each cluster: Let c_j be the centre of the j^{th} cluster and the i^{th} element to be clustered is described by a feature vector x_i: The assumption is that elements are close to the centre of their cluster, yielding an objective function that represents the sum of point-to-centre distances, summed over all *k* clusters:

$$\sum_{j=1}^{k}\sum_{i=1}^{n}\left\|x_i^{(j)}-c_j\right\|^2, \tag{4-22}$$

where $\left\|x_i^{(j)}-c_j\right\|^2=(x_i^{(j)}-c_j)^{\mathrm{T}}\times(x_i^{(j)}-c_j)$ is a chosen distance measure known as the squared Euclidean distance between a given data point $x_i^{(j)}$ in a cluster and the cluster centre c_j. This measure is an indicator of the distance of the *n* data points from their respective cluster centres. The *k*-means algorithm uses a two-phase iterative algorithm to minimise the objective function:

- Phase 1: Assume the cluster centres are known, and allocate each point to the closest cluster centre to form a cluster C_j:

$$C_j^{(t)} = \left\{ x_i : \left\| x_i - c_j^{(t)} \right\| \leq \left\| x_i - c_m^{(t)} \right\| \quad \forall 1 \leq m \leq k \right\} \tag{4-23}$$

where each x_i belongs to one C_j and t indicate the t^{th} iteration.

- Phase 2: Assume the allocation is known, and choose a new set of cluster centres:

$$c_j^{t+1} = \frac{1}{\left| C_j^{(t)} \right|} \sum_{x_i^j \in C_j^t} x_i^j \tag{4-24}$$

Initially the cluster centres are randomly chosen, and then the iteration between these two stages is performed until the process eventually converges to a local minimum of the objective function.

The *k*-means algorithm is used for clustering the training set into three classes. For validation the distance (squared Euclidean distance) between a given test point $x \in R^2$ and the centre $c_j \in R^2$ of each class resulting from the *k*-means clustering algorithm is measured and the class associated with the closest centre is chosen:

$$d_j = \left(\sum_{i=1}^{2} (x_i - c_j^i)^2 \right)^{\frac{1}{2}} \tag{4-25}$$

$$p = \arg \min_j d_j \tag{4-26}$$

4.3.5 Hand rotation detection: Direction intent recognition

Direction intent recognition is achieved by mapping a video sequence of 10 frames with the hand in rotation as the object of interest to a set of two intention curves $\{V_1(i),V_2(i)\}_{i=\{1,\ldots,10\}}$, consisting of the means of the symmetry curves V_1 associated with the faces in each frame and the outputs $V_2 = g(\boldsymbol{x})$ (refer to Equations 4-10, 4-21 and 4-26) from the above mentioned single frame pose classification using the three different machine learning techniques (refer to Sections 2.1.1, 2.1.2 and 2.1.3) respectively. Let $E = \{I_i : I_i$ is the i^{th} frame and $1 \leq i \leq 10$ frames$\}$, a sequence of 10 consecutive frames: $\forall\ I_i \in E$,

$$\left\{V_1(i) = \frac{1}{N}\sum_{k=1}^{N} f_i(y)\right\}_{i=\{1,\ldots 10\}} \tag{4-27}$$

where f_i is the symmetry curve (refer to Equation 3-17) associated with I_i. The resulting intention curve V_1 is depicted in Figure 4-7 for each scenario and it can be observed that rotation from the centre to either side (right or left) exhibits the same patterns while rotation from either side to the centre also exhibits the same patterns, but is different from that of the previously mentioned rotation from the centre to either side. It is therefore possible using V_1, to distinguish between a rotation from the centre position to either side and a rotation from either side to the centre position. However, insufficient information is provided to distinguish between rotation to the left and rotation to the right. To address this problem, a preliminary step is implemented that consists of using the other intention curve V_2 consisting of the output classes $n = \{1,2,3\}$ of the MLP, the SVM or the *K*-means corresponding to the centre, right and

left positions respectively. For a centred motion, 10 consecutive 1s are expected, while 10 consecutive or at least a majority of 2s and 3s are expected for right and left scenarios respectively. A majority vote scheme is used, which counts the number of 1s, 2s and 3s that are found in V_2, and classify it as a centre, right or left indicator:

$$d'(n) = \sum(\{V_2(i)\}_{i=\{1,\ldots 10\}} == n), \ \forall\, n = \{1,2,3\} \tag{4-28}$$

$$n' = \arg\max_{n} (d') \tag{4-29}$$

Let $\{V_1(i)\}_{i=\{1,\ldots,10\}}$ be the intention curve (refer to Equation 4-27) to be classified into classes $\omega_1,\ldots,\omega_5$ corresponding to the centre, right (from centre-right), left (from centre-left), left (from right-centre) and right (from left-centre) intentions respectively. δ is obtained using Equation 3-27 (where V_1 replaces) and d_n and P_n are obtained using Equations 3-28 and 3-30 respectively: A decision rule h is defined using Equation 4-30 for a "difference of means approach", and Equation 4-31 for a "statistics in Gaussian distribution" approach:

$$h(\{V_1(i),V_2(i)\}_{i=\{1,..,10\}}) = \begin{cases} \omega_1, n' == 1 \wedge |\delta| \leq \lambda \wedge d_1 == min([d_n]_{n=\{1,2,3\}}) \\ \omega_2, n' == 2 \wedge \delta < \lambda \wedge d_2 == min([d_n]_{n=\{1,2,3\}}) \\ \omega_3, n' == 3 \wedge \delta < \lambda \wedge d_3 == min([d_n]_{n=\{1,2,3\}}) \\ \omega_4, n' == 2 \wedge \delta > \lambda \wedge d_2 == min([d_n]_{n=\{1,2,3\}}) \\ \omega_5, n' == 3 \wedge \delta > \lambda \wedge d_3 == min([d_n]_{n=\{1,2,3\}}) \end{cases} \tag{4-30}$$

$$h(\{V_1(i),V_2(i)\}_{i=\{1,..,10\}})$$

$$= \begin{cases} \omega_1, n' == 1 \wedge |\delta| \leq \lambda \wedge P_1 == max([P_n]_{n=\{1,2,3\}}) \\ \omega_2, n' == 2 \wedge \delta < \lambda \wedge P_2 == max([P_n]_{n=\{1,2,3\}}) \\ \omega_3, n' == 3 \wedge \delta < \lambda \wedge P_3 == max([P_n]_{n=\{1,2,3\}}) \\ \omega_4, n' == 2 \wedge \delta > \lambda \wedge P_2 == max([P_n]_{n=\{1,2,3\}}) \\ \omega_5, n' == 3 \wedge \delta > \lambda \wedge P_3 == max([P_n]_{n=\{1,2,3\}}) \end{cases} \quad (4\text{-}31)$$

where $\lambda \geq 0$

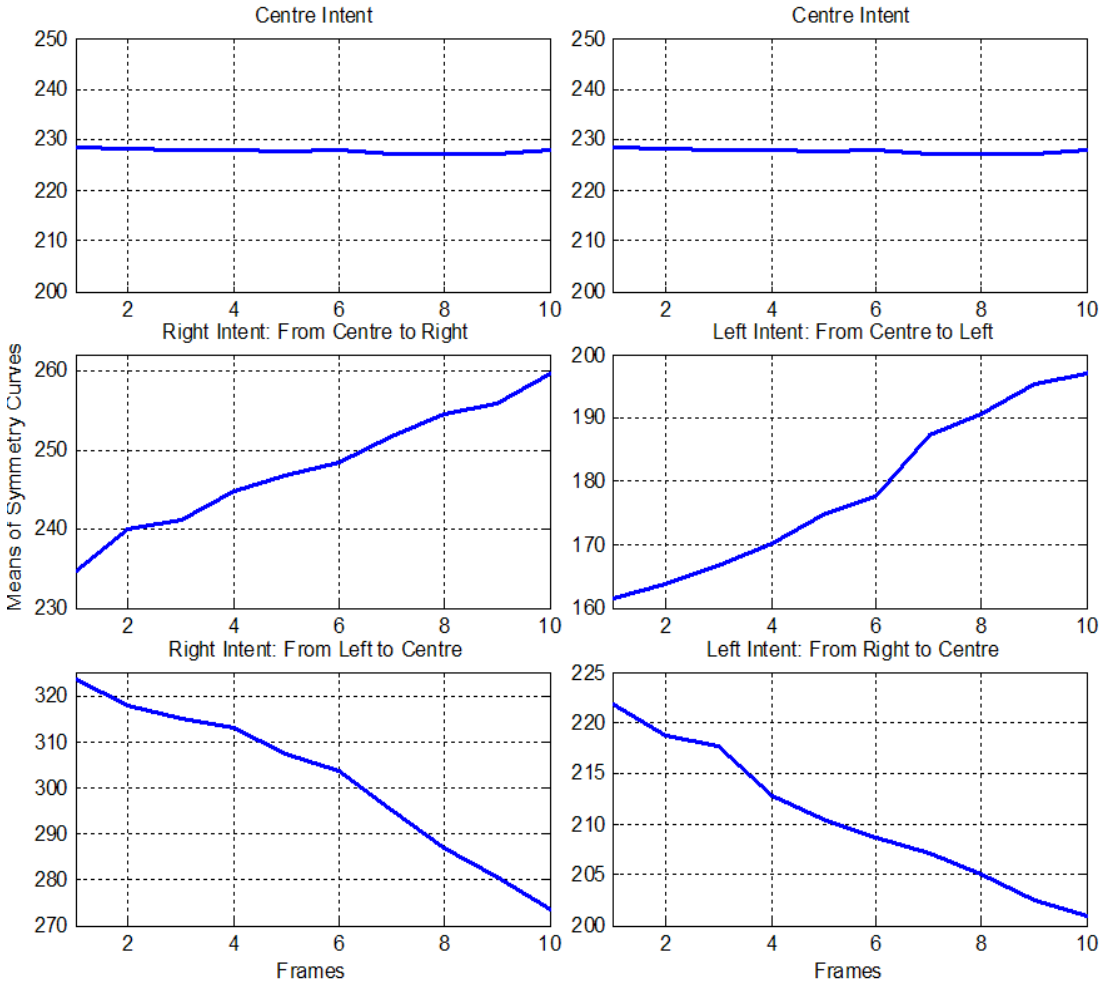

Figure 4-7: Intention curve V_1 made of symmetry curves' means

4.3.6 Template-matching-based direction intent recognition

The other proposed solution is based on the difference in appearance of the different positions (centre, right and left) of the hand due to the difference in finger edge appearance and orientation. A template matching [188], [191], which is a simple task of performing cross-correlation between a template and a new image, is performed on the hand region containing the fingers to classify a single frame hand pose, and a decision rule is used to classify the resulting intention curve represented by the varying template matching measures throughout a 10-frame video sequence. The template consists of the region of the hand containing the fingers and the template matching task first detects the sub-window within the image containing the hand (refer to Figure 4-3 for examples of hand images) that is closest to each template (refer to Equation 3-46). Subsequently, the inferred position corresponds to that of the template where the match to the given hand is the highest (refer to Equation 3-47). Figure 4-8 depicts the hands in rotation for three different positions (Part a) and their corresponding sub-windows containing only the finger region (Part b).

Let g be an $m \times n$ template of the finger region of a hand and its instances must be detected in an image I. A normalised cross-correlation template matching is implemented using Equations 3-45, 3-46 and 3-47 for single pose classification. Three templates from a single subject are used, consisting of the region comprising the fingers in a centred, right and left hand as portrayed in Figure 4-8 (Part b). The best match in the image is the highest value of $M(r,c)$. Since this highest value criterion is not sufficient; in cases where an image does not contain the object of interest, this highest value should also be above a certain threshold $\lambda = 2.5$ obtained empirically by trial and error to indicate a match, that is, $max\left(\{M_i\}_{i=\{1,2,3\}}\right) \geq \lambda$.

For direction intent recognition through rotation of the hand, the intention curves of each motion are represented by the changes in template matching measures between the detected hand region and the hand region templates associated with a centre, right and left position of the hand.

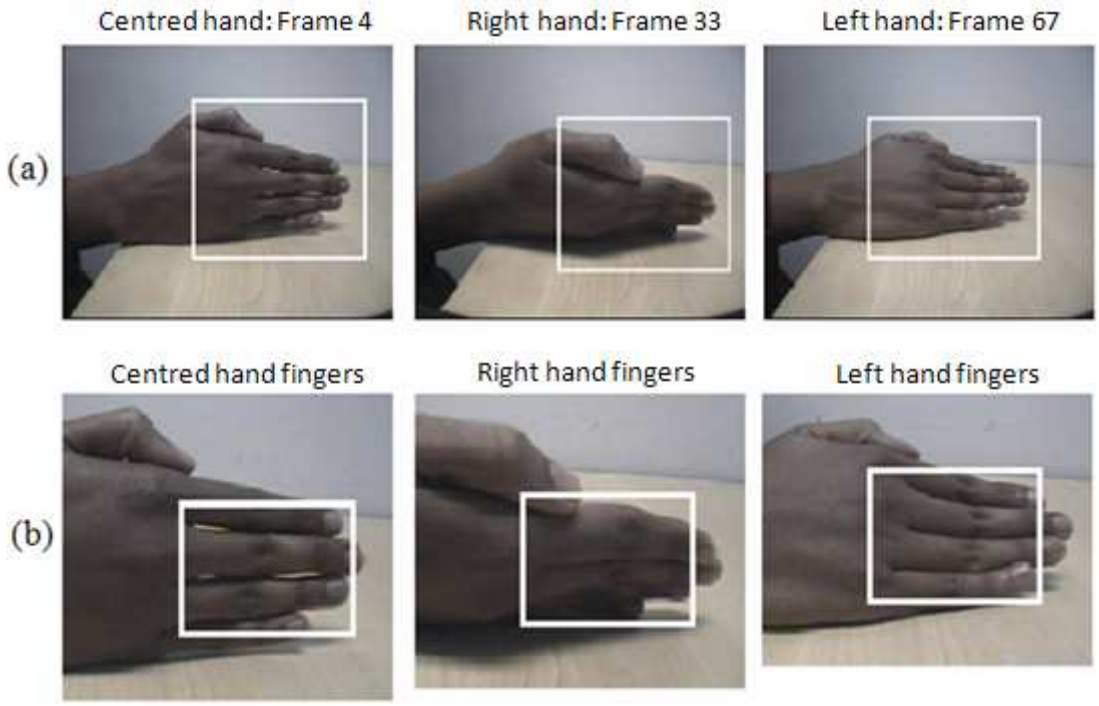

Figure 4-8: Detection of hands in rotation and their finger regions

Let $\{M_m(i), \forall\, m = 1,2,3\}_{i=\{1,\ldots,10\}}$ be the set of intention curves each composed of a 10-point sequence of matching measures (refer to Equation 3-45) of the finger region of the hand images with a centred, a right and a left finger region template respectively. These sets of m intention curves $\{M_m(i), \forall\, m = 1,2,3\}_{i=\{1,\ldots,10\}}$ depicted in Figure 4-9 exhibit separable patterns for each of the intent classes $\omega_1,\ldots,\omega_5$, corresponding to the centre, right (from centre-right), left (from centre-left), left (from right-centre) and right (from left-centre) intentions respectively. δ_m is obtained using Equation 3-37 where $d_{n,m} = M_{n,m}$, and a decision rule h is defined using Equation 3-50.

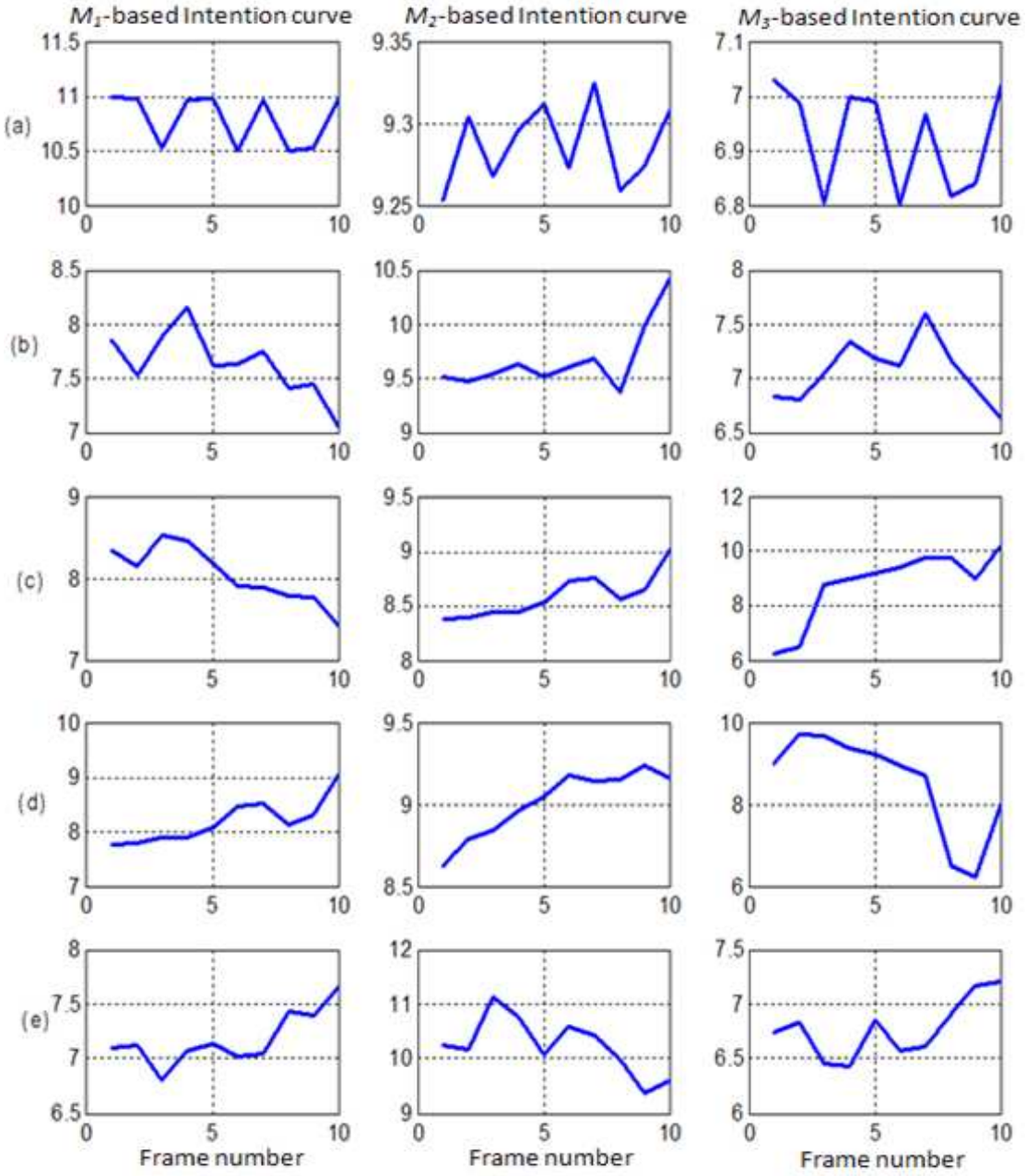

Figure 4-9: Intention curves based on matching measures M_1, M_2 and M_3

4.4 Recognition of hand-based speed variation intent

Speed variation is inferred by observing the vertical motion of the hand according to Table 4-2. Two solutions are proposed: The first is a normalised cross-correlation template matching as described in the previous section (Section 4.3.6) for hand rotation. The second solution is based on the position of the detected hand's contour using an ellipse shaped mask.

Table 4-2: Hand vertical motion and corresponding speed variation intention

Motion of the hand	Inferred Intention
Down vertical motion	Increased speed
Up vertical motion	Decreased speed
No vertical motion (Centred position)	Intent to remain in current speed

4.4.1 Template Matching-based speed variation recognition

This approach is based on the difference in appearance of the different positions (centre, up, down) of the hand where the orientation of the edges of the fingers presents separable patterns for those different vertical positions. Figure 4-10 displays three different positions of the hands in vertical motion (Part a) and their corresponding sub-windows containing only the finger region (Part b). A normalised cross-correlation template matching (refer to Sections 3.5.3 and 4.3.6 where the same approach is used for nose template matching head vertical motion detection and for hand direction recognition respectively) is used to classify the different single frame positions of the hand.

For speed variation intent recognition through vertical motion of the hand, the intention curves of each motion are represented by the changes in template matching measures between the detected hand region and the hand region templates associated with a centre, right and left position of the hand: Let $\{M_m(i), \forall\, m = 1,2,3\}_{i=\{1,\ldots,10\}}$ be the set of intention curves each composed of a 10-point sequences of matching measures (refer to Equation 3-45) of the given finger region in the hand images with a centred, up and down finger region template respectively. These sets of m intention curves $\{M_m(i), \forall\, m = 1,2,3\}_{i=\{1,\ldots,10\}}$ illustrated in Figure 4-11 exhibit separable patterns for each of the intent classes $\omega_1,\ldots,\omega_5$, corresponding to the centre, up (from centre- up), down (from centre-down), down (from up-centre) and up (from down-centre) intentions respectively. δ_m is obtained using Equation 3-37 where $d_{n,m} = M_{n,m}$, and a decision rule h is defined using Equation 3-50.

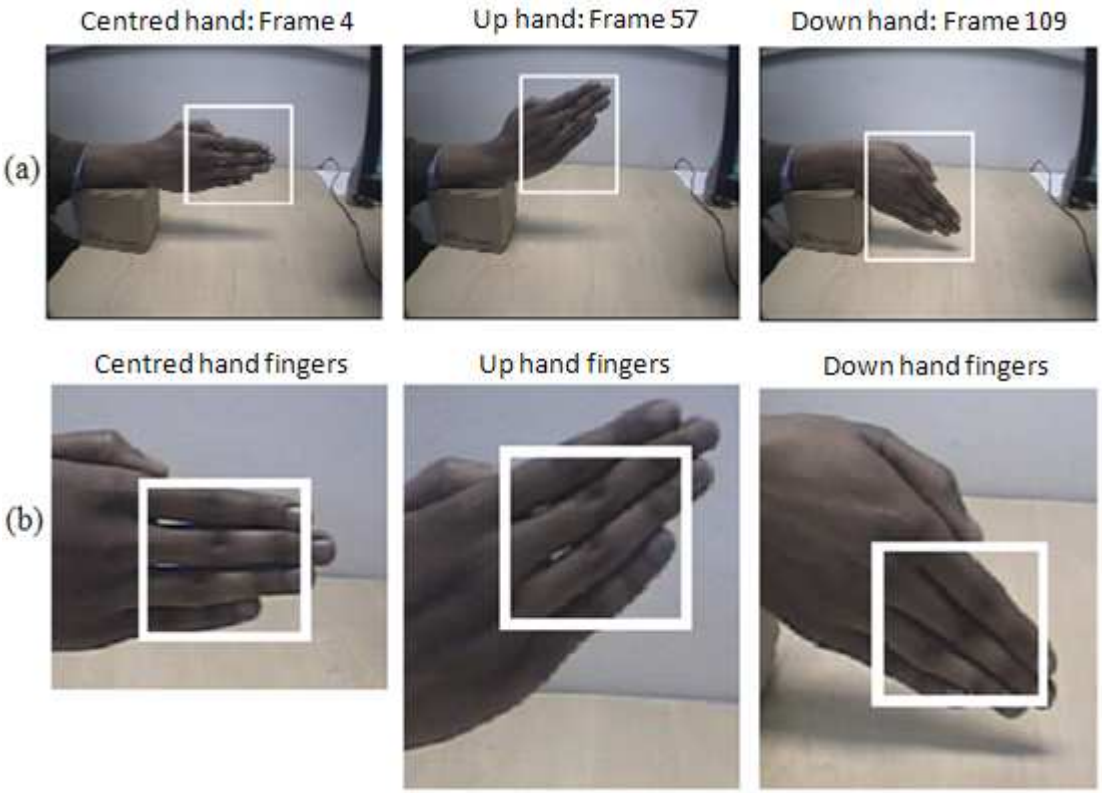

Figure 4-10: Detection of hands in vertical motion and their finger regions

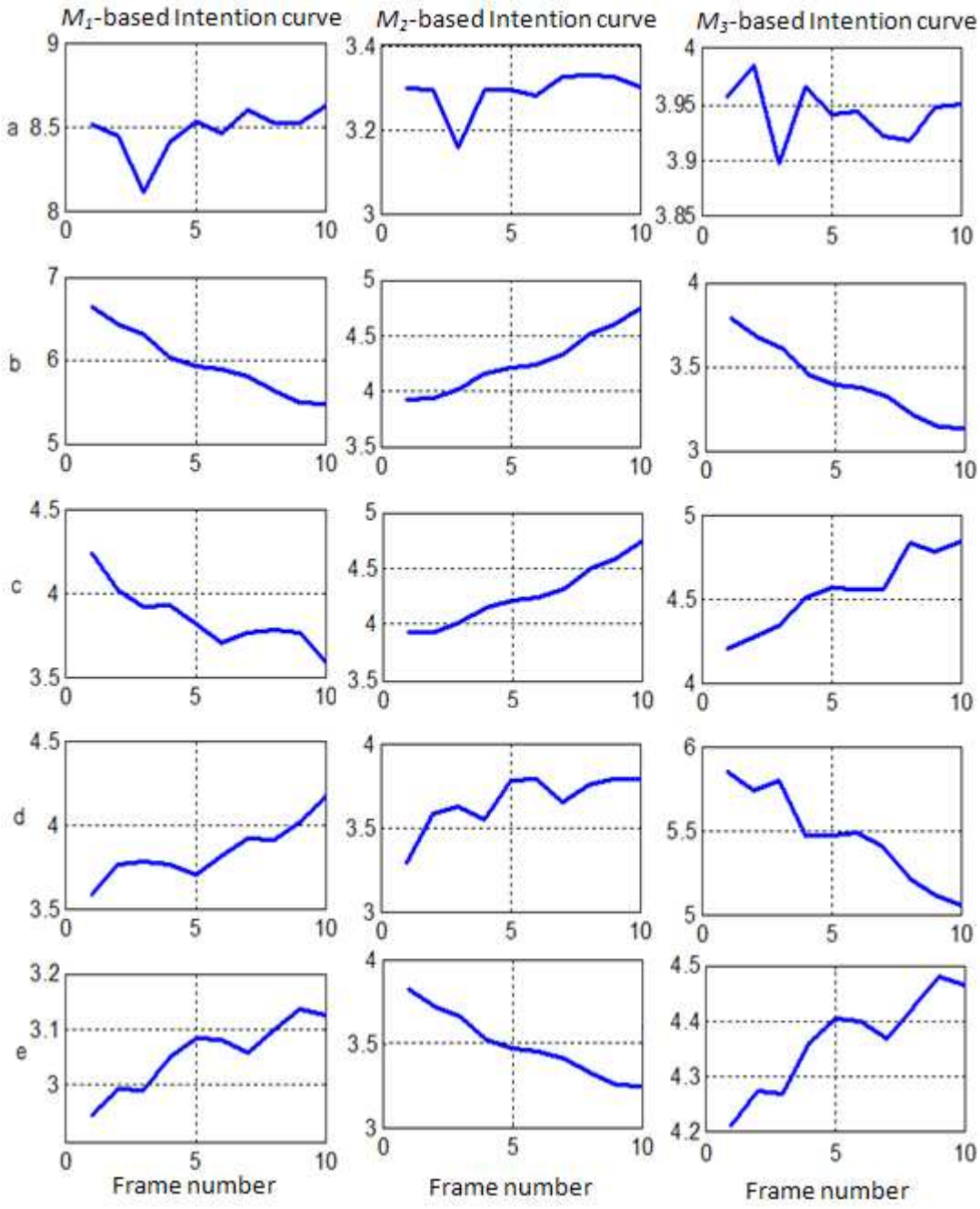

Figure 4-11: Intention curves based on matching measures M_1, M_2 and M_3

4.4.2 Speed variation recognition based on ellipse shaped mask

This approach is based on the position of the detected hand's contour with respect to the horizontal line evaluated using a mask in the shape of an ellipse whose major axis' length is equal to the length (determined empirically) of a typical hand at a fixed distance from the camera. As illustrated in the binary image in Figure 4-13 (Part a), the detected skin colour region R corresponding to the hand has a shape close to that of an ellipse. The centre (x_c,y_c) of R is found and an ellipse centred at that point (x_c,y_c) is used as a contour mask. Subsequently, a search is performed by rotating the ellipse mask around that point until the maximum number of skin colour pixels within the ellipse is reached. The rotation ranges from $-\pi/6$ to $\pi/6$, a practical range for a hand in vertical motion: Let $\boldsymbol{\theta} = \{\theta_i: -\pi/6 \leq \theta_i \leq \pi/6\}$ be the set of angles between the line y_m containing the major axis of the ellipse $E_i \ \forall \ i = \{1,2,3\}$ (corresponding to the centre, up and down position respectively) and the horizontal line $y = y_c$ through the ellipse's centre (x_c,y_c) (refer to Figure 4-12 that shows θ for three different positions of the ellipse corresponding to the three different positions of the hand):

Let $\Phi = \left\{ \sum_{\theta_i} (x,y) \right\}_{i=\{1,2,3\}} \quad \forall (x,y) \in R \cap E_i$, that is (x,y) is a skin colour pixel and $\forall \theta_i \in \boldsymbol{\theta}$ (Φ is therefore a function of θ_i) the inclination corresponding to the vertical position of the hand is given below:

$$\varphi = \arg\max_{\theta_i} \Phi(\theta_i) \tag{4-32}$$

The resulting position φ of the ellipse corresponds to the position of the hand and belongs to the class $\omega_i \ \forall \ i = \{1,2,3\}$ corresponding to the centre, up and down position respectively, according to the decision rule *h* defined as:

$$h(\varphi) = \begin{cases} \omega_1, |\varphi| \leq \lambda_1 \\ \omega_2, \varphi < \lambda_1 \\ \omega_3, \varphi > \lambda_1 \end{cases} \quad (4\text{-}33)$$

where $\lambda_1 > 0$. Figure 4-13 (Part b) depicts the binary image containing the detected skin colour region corresponding to the hand for three different positions of the hand, and the ellipse mask associated with them. The inclination angle φ (in radian) between the major axis of the ellipse y_m and the horizontal axis through the centre $y = y_c$ determines the single frame hand pose and varies differently for the different motions (centre, up and down).

For speed variation intent recognition through vertical motion of the hand, the intention curves for each motion are represented by the changes in angle values between the major axis y_m of the ellipse mask approximating the skin colour detected region and the horizontal axis $y = y_c$ for hands in vertical motion: Let $\{\theta(i)\}_{i=\{1,\ldots,10\}}$ be a sequence of angles between the major axis of the ellipse and the horizontal axis through all the frames in a sequence of 10 frames and θ' the angle of the ellipse mask associated with the first frame in the sequence. δ is defined in Equation 4-34 as the constant, increasing and decreasing tendencies of the successive values of the sequences $\{\theta(i)\}_{i=\{1,\ldots,10\}}$ that exhibit different patterns for the different motions as portrayed in Figure 4-14. *h* is used for classification of these intention curves in classes $\omega_1,\ldots,\omega_5$, corresponding to the centre, up (from centre-up), down (from

centre-down), down (from up-centre) and up (from down-centre) intentions respectively.

$$\delta = \sum_{i=1}^{N-1} \theta(i) - \theta(i+1) \tag{4-34}$$

$$h(\{\theta(i)\}_{i=\{1,\ldots,10\}}) = \begin{cases} \omega_1, |\theta'| \le \lambda_1 \wedge |\delta| \le \lambda_2 \\ \omega_2, \theta' < \lambda_1 \wedge \delta > \lambda_2 \\ \omega_3, \theta' > \lambda_1 \wedge \delta < \lambda_2 \\ \omega_4, \theta' < \lambda_1 \wedge \delta < \lambda_2 \\ \omega_5, \theta' > \lambda_1 \wedge \delta > \lambda_2 \end{cases} \tag{4-35}$$

where $\lambda_1, \lambda_2 > 0$

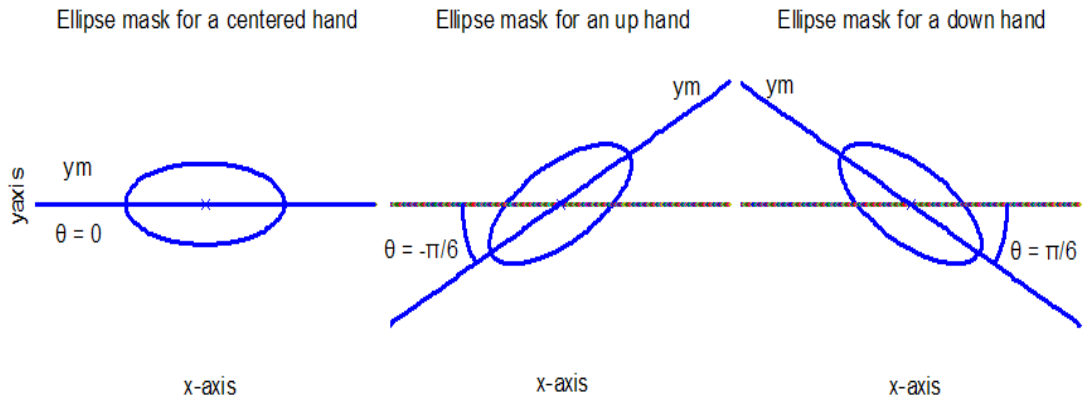

Figure 4-12: Three different positions of an ellipse used as a mask

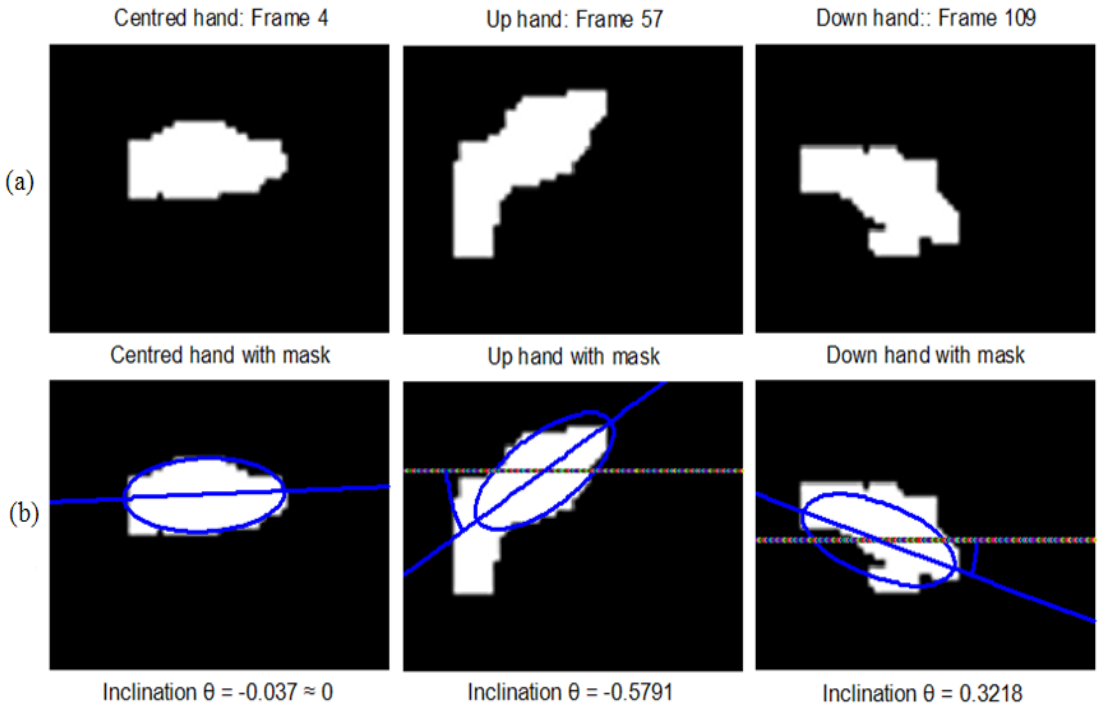

Figure 4-13: Ellipse mask used to determine the vertical position θ of the hand

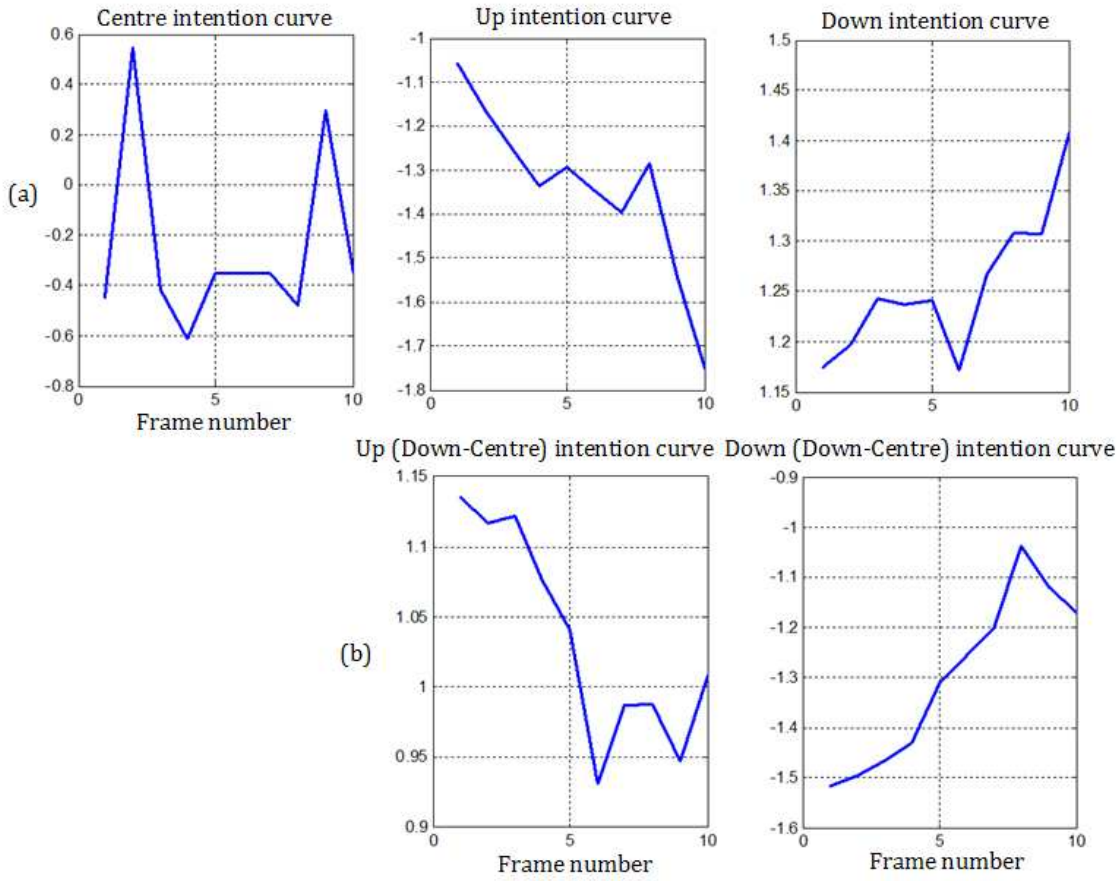

Figure 4-14: Intention curves based on changes in θ for each hand motion

4.5 Histogram of oriented gradient (HOG) for hand-based speed variation recognition

To emphasise the merit of the proposed approach, an algorithm based on HOG is implemented to compare the results for detection of hands in vertical motion. For hands in rotation, the methods surveyed in the literature including the HOG remain inadequate because of the nature of the motion where these methods would detect little changes. Therefore, no comparative method is proposed and implemented.

The literature indicates that hand gesture recognition can be achieved by using orientation histograms [179]. For the application at hand, however, unlike classical gesture recognition where the hand significantly changes its shape or contour and where significant translations of the hand occur, the shape of the hand remains rigid, merely changing its vertical position. A more appropriate approach turns out to be the HOG used in the literature for human activity recognition [195], [201], [202]. HOG is a feature descriptor inspired by the Scale-Invariant Feature Transform (SIFT) descriptors. The essential idea behind the HOG descriptors is that local object appearance and shape within an image can be described by the distribution of intensity edge directions.

The implementation of these descriptors is achieved by dividing the input image I into small 4×4 non-overlapping rectangular regions R, called cells. For each cell, a histogram of gradient orientations is compiled for the pixels within the cell by counting the occurrences of the gradient orientation in that cell: A rectangular Gaussian filter G_{rect} is used to produce the rectangular regions R by means of the convolution of the edge image I_E resulting from a canny edge detection approach, with this rectangle filter G_{rect}:

$$G_{rect}(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \tag{4-36}$$

$$R(x,y) = G_{rect}(x,y) \otimes I_E(x,y) \tag{4-37}$$

$\forall (x,y) \in R$ and where G_{rect} is a zero-padded rectangular patch of the 2D Gaussian.

The combination of these histograms represents the HOG descriptor with five components: the horizontal component, the vertical, the two diagonals, and the non-directional component. This descriptor is a good detector of the orientation of finger edges as illustrated in Figure 4-15 where three different positions of a hand in vertical motion and the HOG descriptors associated with them are given, using only three of the five gradient orientation components namely the horizontal, and the two diagonal orientations.

For single frame hand pose classification, a centred hand is selected if the horizontal component is the highest, an up hand is chosen if the first diagonal component is the highest and a down hand is classified if the second diagonal component is the highest. For speed variation intent recognition, through vertical motion of the hand, the intention curves for each motion are represented by the changes in this set of three HOG components for hands in vertical motion: Let $\{S_n(i) \; \forall n = 1,2,3\}_{i=\{1,\ldots,10\}}$, be the set of sequences for the horizontal, first diagonal and second diagonal components respectively in the HOG descriptor associated with a sequence of hand images. As depicted in Figure 4-16, they exhibit separable patterns for different motions and therefore classification of these set of sequences into $\omega_1,\ldots,\omega_5$, corresponding to the centre, up (from centre-up), down (from centre-down), down (from up-centre) and up (from down-centre) intentions respectively is achieved as follows:

$$\delta_n = \sum_{i=1}^{N-1} S_n(i) - S_n(i+1) \tag{4-38}$$

$h(\{S_n(i)\ \forall n = 1,2,3,4\}_{i=\{1,\ldots,10\}})$

$$= \begin{cases} \omega_1, |\delta_1| \leq \lambda \wedge S_1(i) > S_m(i), \forall m = \{2,3,4\}, i = \{1,\ldots,10\} \\ \omega_2, \delta_1 > \lambda \wedge \delta_3 < \lambda \wedge |\delta_3| > |\delta_m| \ \forall m = \{1,2,4\} \\ \omega_3, \delta_1 > \lambda \wedge \delta_4 < \lambda \wedge |\delta_4| > |\delta_m| \ \forall m = \{1,2,3\} \\ \omega_4, \delta_1 < \lambda \wedge \delta_3 > \lambda \wedge \delta_3 > \delta_m \ \forall m = \{1,2,4\} \\ \omega_5, \delta_1 < \lambda \wedge \delta_4 > \lambda \wedge \delta_4 > \delta_m \ \forall m = \{1,2,3\} \end{cases} \quad (4\text{-}39)$$

where $\lambda = 2$ and was chosen empirically by trial and error.

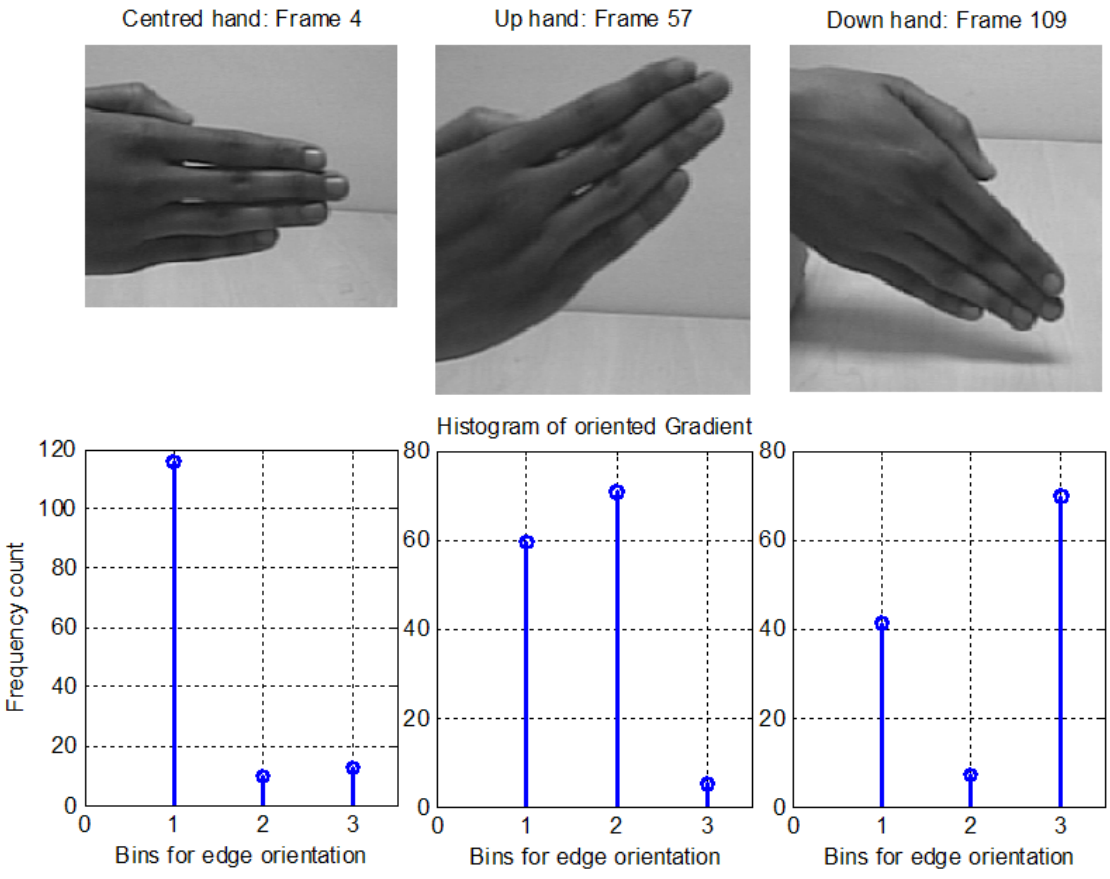

Figure 4-15: HOG descriptor for hands in vertical motion

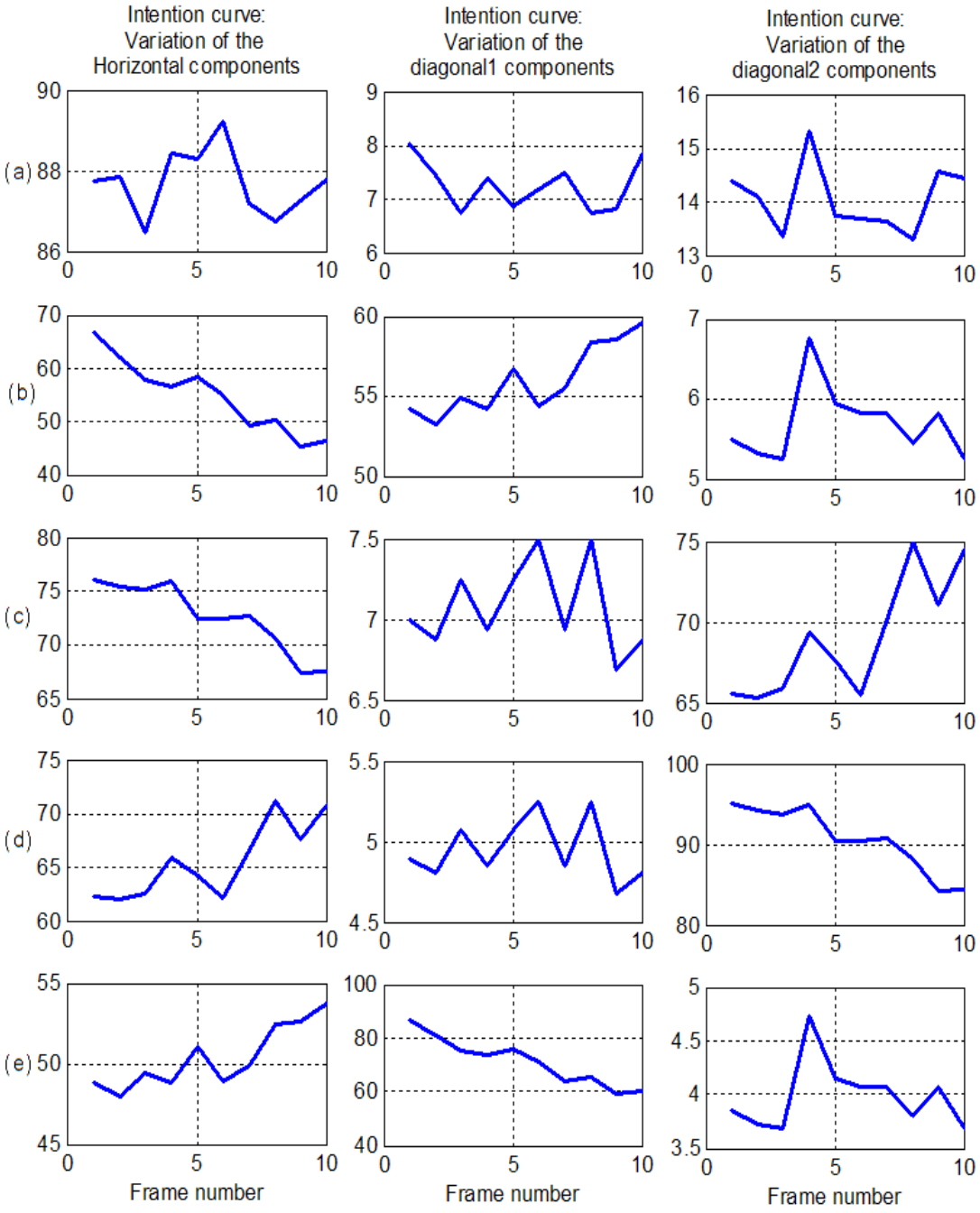

Figure 4-16: Intention curves based on changes in the HOG components

4.6 Conclusion

In summary, this chapter offers a detailed description of the algorithms proposed in this thesis aimed at visual hand-based motion detection for intent recognition. The pre-processing steps (detection and tracking of the hand) are implemented using skin colour detection, some image processing operation (dilation and connected components labelling) and a prior knowledge of the dimensions of a typical hand. The overview of the intent recognition algorithm consists of using a 10-frame video sequence as input that is mapped to an intention curve that presents separable patterns for each possible intention.

For direction intent recognition, a vertical symmetry-based approach along with three machine learning approaches (neural network, support vector machines and *k*-means clustering) are used to form two sets of intention curves. Another approach is based on template matching where the varying matching measures are used to form the intention curves. For speed variation intent recognition, the template matching approach is also used to form the intention curve, while the other proposed method uses a mask in the shape of an ellipse to determine the vertical position of the hand. The intention curve is formed using the varying vertical position throughout the sequence. For comparison with the proposed solutions for detection of vertical motion of the hand, an HOG descriptor [195] is implemented and the resulting intention curves are formed using the varying values of 3 of the 5 HOG components namely the horizontal and the two diagonal components. The appropriate decision rule is then used to classify these intention curves for intent recognition.

The next chapter discusses the results of the methods described in this chapter (refer to Section 5.3) as well as those described in Chapter 3 (refer to Section 5.2).

Chapter 5

Results and Discussion

5.1 Introduction

As mentioned in Section 1.1, the type of datum used is a sequence of 576×768 image frames captured from a CCD camera (Hi-Resolution Dome Camera - 1/3" CCD, 470 TV lines, 0.8 lux, 3.6mm (F2.0) Lens) and a “25 frames per second” E-PICOLO-PRO-2 frame grabber. The two intent indicators considered in this work are the head and the hand in motion and are therefore the objects of interest in the video sequences. Experimental results have therefore been obtained by collecting 2 sets of video sequences of 20 different subjects with 5 long sequences each, with the head and the hand in motion as objects of interest respectively. These long video sequences are divided into several 10-frame sequences for intention inference. The video sequences of 10 subjects are used for all the training tasks, and the video sequences of the 10 others are used for validation.

For head rotation (refer to Chapter 3, Section 3.3), the symmetry property of the head is used as the basis of the proposed method where a symmetry-based approach that maps a face to a symmetry curve is implemented. From this symmetry-based approach four different methods are proposed according to the feature selection process (centre of gravity of the symmetry curve or *y*-intercept of the line approximating the symmetry curve) and the decision rule (based on difference of

means or statistics in a Gaussian distribution) for pose and intent recognition. Figure 5-1 summarizes the methods used for head rotation detection given video frames from the testing set. For the vertical motion of the head (refer to Chapter 3, Section 3.4), PCA is used for pose and intent recognition. A method proposed by Jia and Hu [66], [67] based on adaboost, camshift and nose template matching is also implemented for comparison. The approach uses adaboost and camshift for face detection and tracking, and another layer of adaboost and a nose template matching for rotation and vertical motion recognition respectively (refer to Figure 5-2). For the vertical motion of the hand (refer to Chapter 4, Section 4.4), an ellipse shaped mask is used to determine its position. The other proposed approach is based on a normalised cross-correlation template matching and for comparison to the proposed methods for vertical motion of the hand; a feature selection technique found in the literature known as histogram of oriented gradient [195] is implemented. For hand rotation (refer to Chapter 4, Section 4.3), a vertical symmetry-based approach is used. For classification, the statistics (mean and standard deviation) of the resulting symmetry curves are subsequently used as 2D data features for three different machine learning methods (two supervised and one unsupervised): a neural network, a support vector machine and *k*-means clustering. Another method is proposed based on a normalised cross-correlation template matching. Figures 5-3 and 5-4 summarize the proposed methods for vertical motion and rotation detection of the hand respectively.

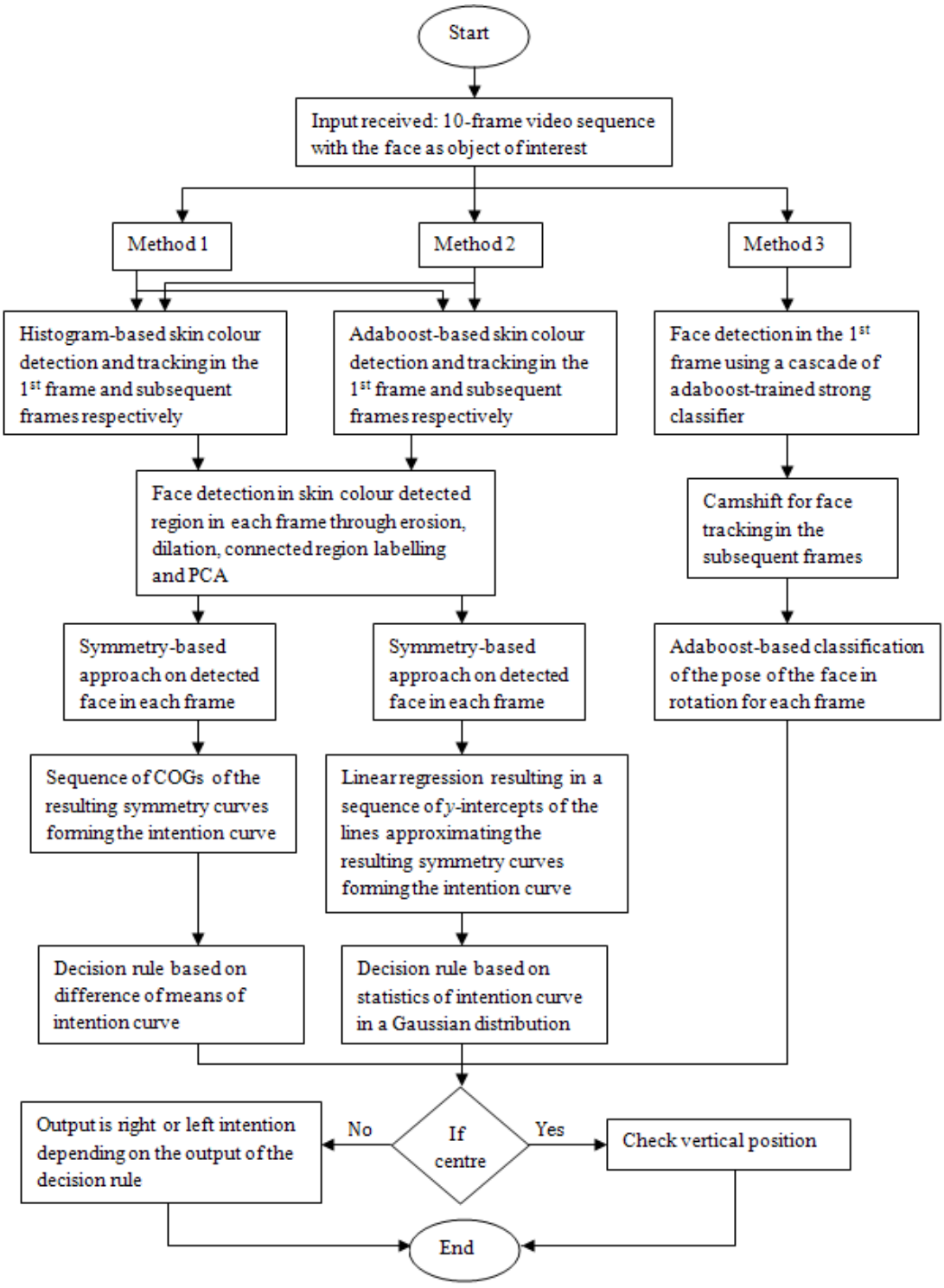

Figure 5-1: Summary of the methods used for head rotation detection

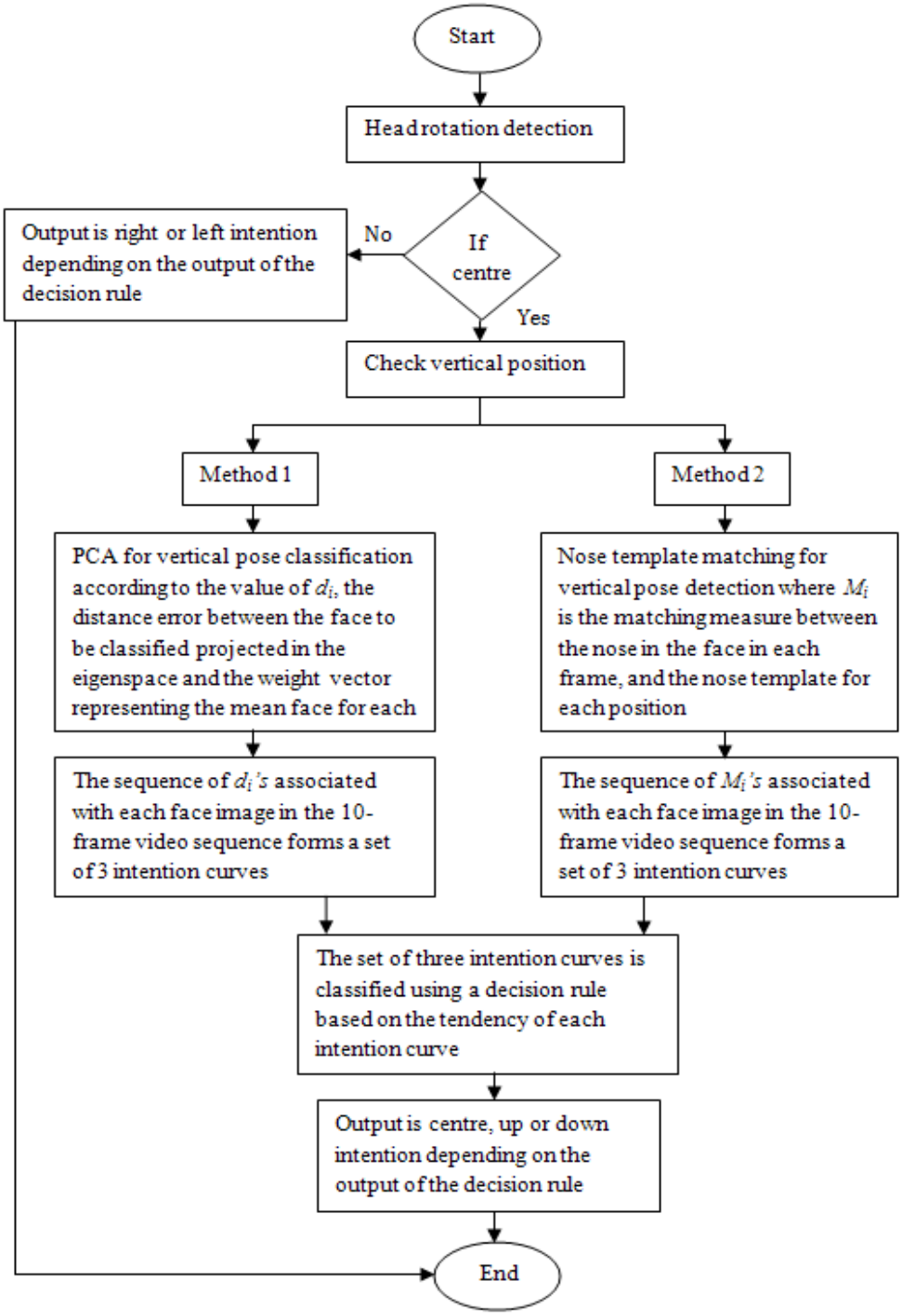

Figure 5-2: Summary of the methods used for head vertical motion detection

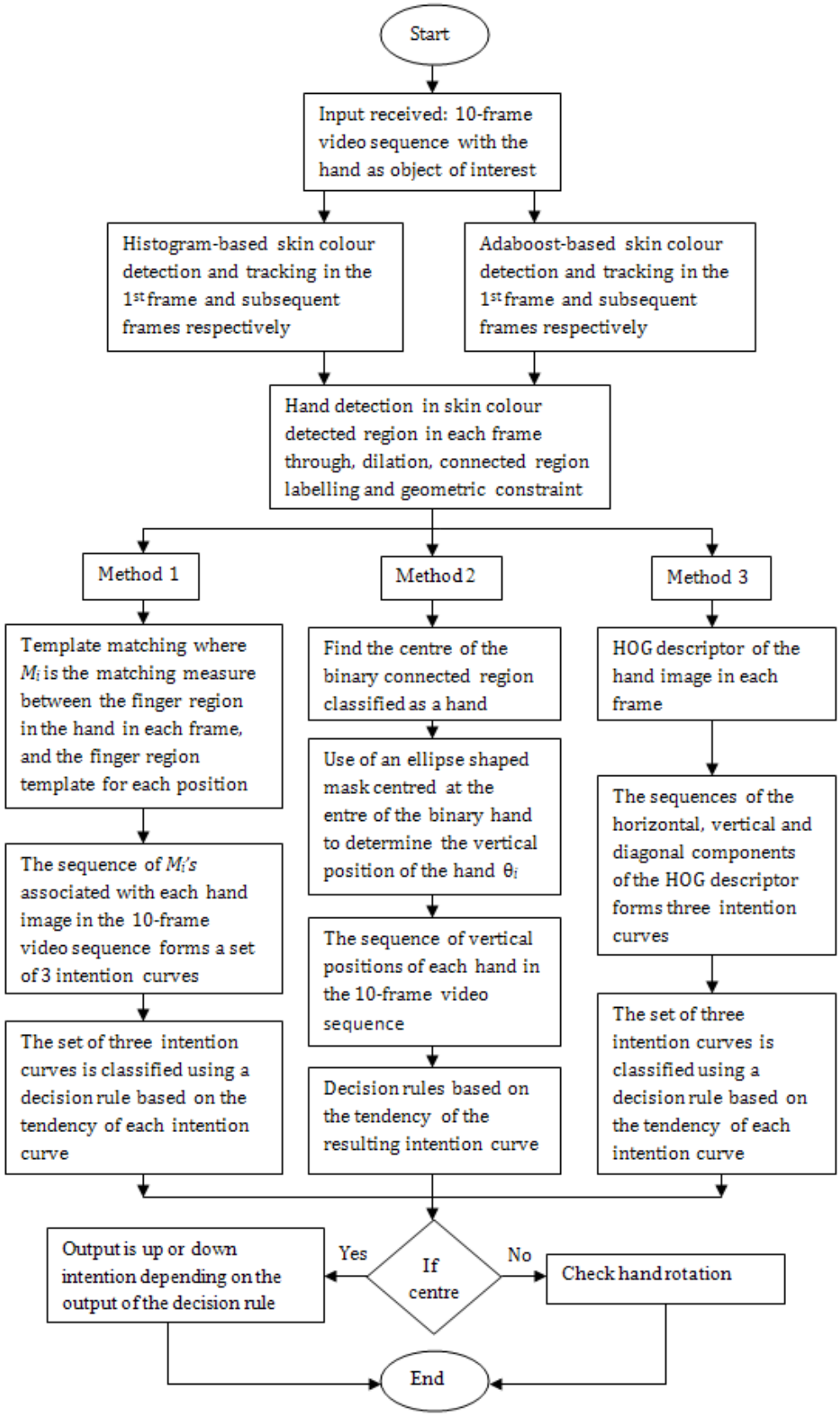

Figure 5-3: Summary of the methods used for hand vertical motion detection

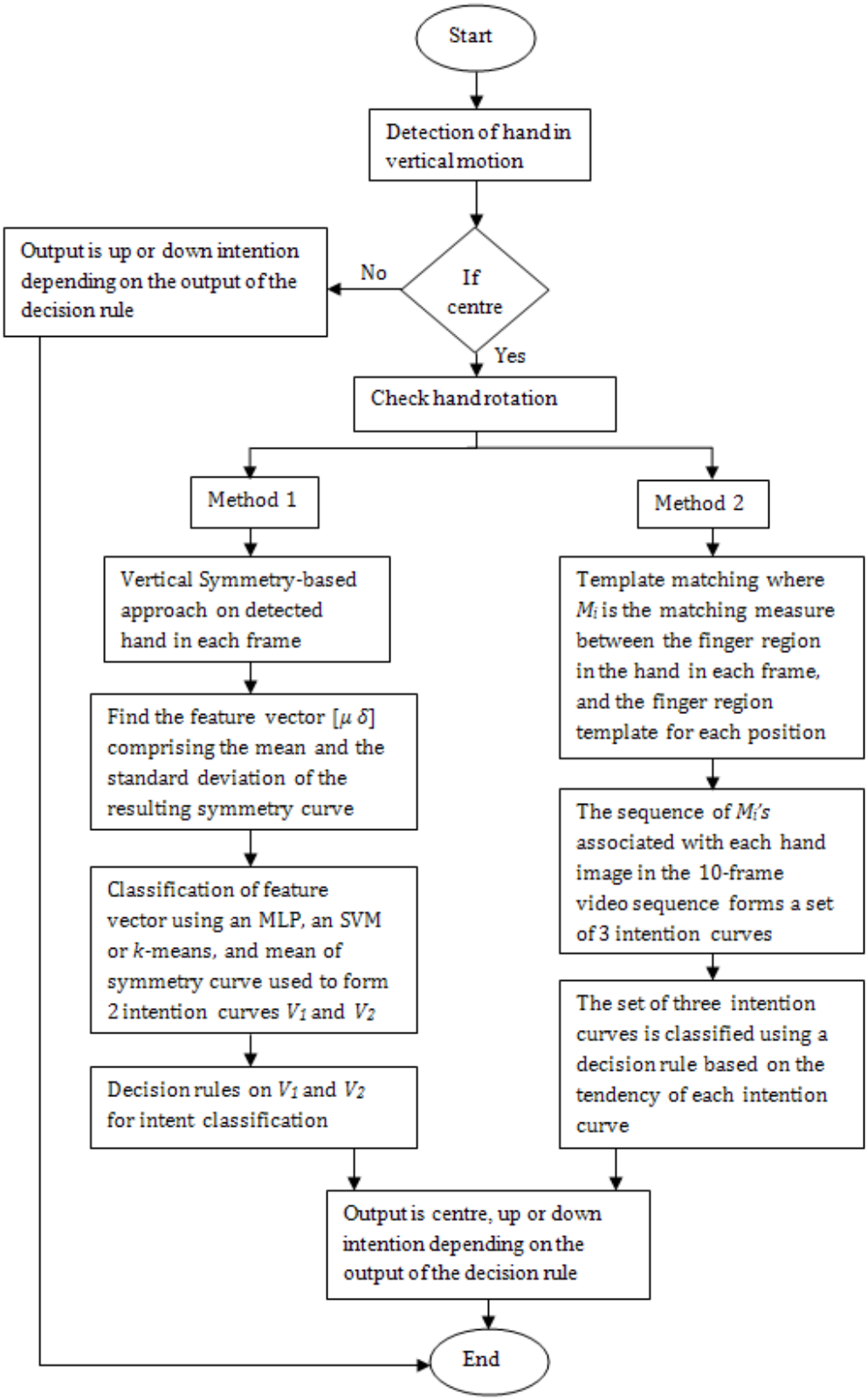

Figure 5-4: Summary of the methods used for hand rotation detection

For performance evaluation this work makes use of the hold-out method where the data set at hand is divided in two mutually exclusive parts; one for training while the other is held out for testing. This is a popular method to assess the system's performance [203] and appropriate for our data sets where the subjects used for testing are different from those used for training. This method makes an inefficient use of the data (using only part of it to train the classifier) and therefore gives a pessimistically biased error estimate [198].

Three sets of results are given below for each proposed and implemented method: the first set shows the performance for single frame pose classification, and the second set depicts the performance for intent recognition through classification of intention curves. For single frame pose classification, Figure 5-5 depicts the range of right head poses from (a) to (b), left head poses from (c) to (d), up head poses from (e) to (f) and down head poses from (g) to (h). After detection/tracking, the frame is converted from colour to greyscale. A similar illustration is given for the hand in Figure 5-6. For intent recognition, the right/left intents include motion from centre to right/left as well as the back motion from right to centre for left motion and left to centre for right motion. The same applies for up/down intents. The third set of results depicts the performance when in a 10-frame video sequence, the number of processed frames is reduced by choosing every 2,3,4,5 frames resulting in the respective numbers of processed frames 5,3,2,2 to form the intention curves that are subsequently extrapolated into a 10-point intention curve. Note that the values of the thresholds used in decision rules described in Equations 3-38, 3-50, 4-30, 4-31 and 4-35 and set empirically by trial and error using the training sets are given in Table 5-1 below:

Table 5-1: Thresholds used in decision rules

Equation	λ
(3-38)	$\lambda = 0.8$
(3-50)	$\lambda = 0.35$
(4-30)	$\lambda = 0.6$
(4-31)	$\lambda = 0.6$
(4-33)	$\lambda_1 = 0.01$
(4-35)	$\lambda_1 = 0.01\ rad$, $\lambda_2 = 0.5$

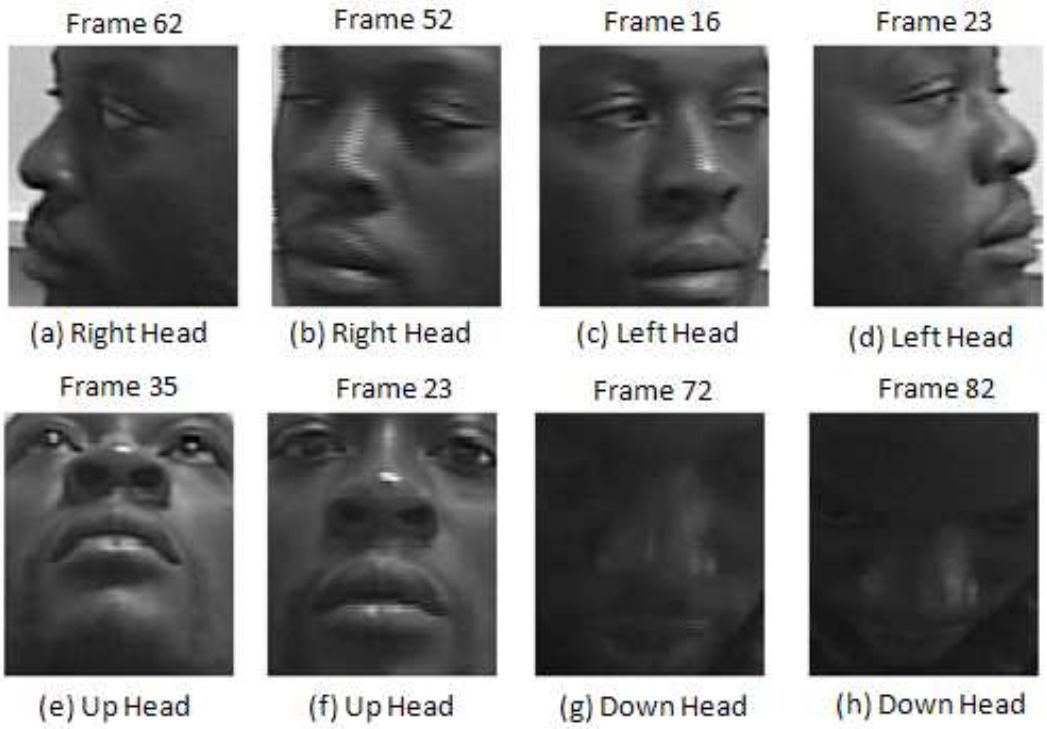

Figure 5-5: Range of right, left, up and down head poses

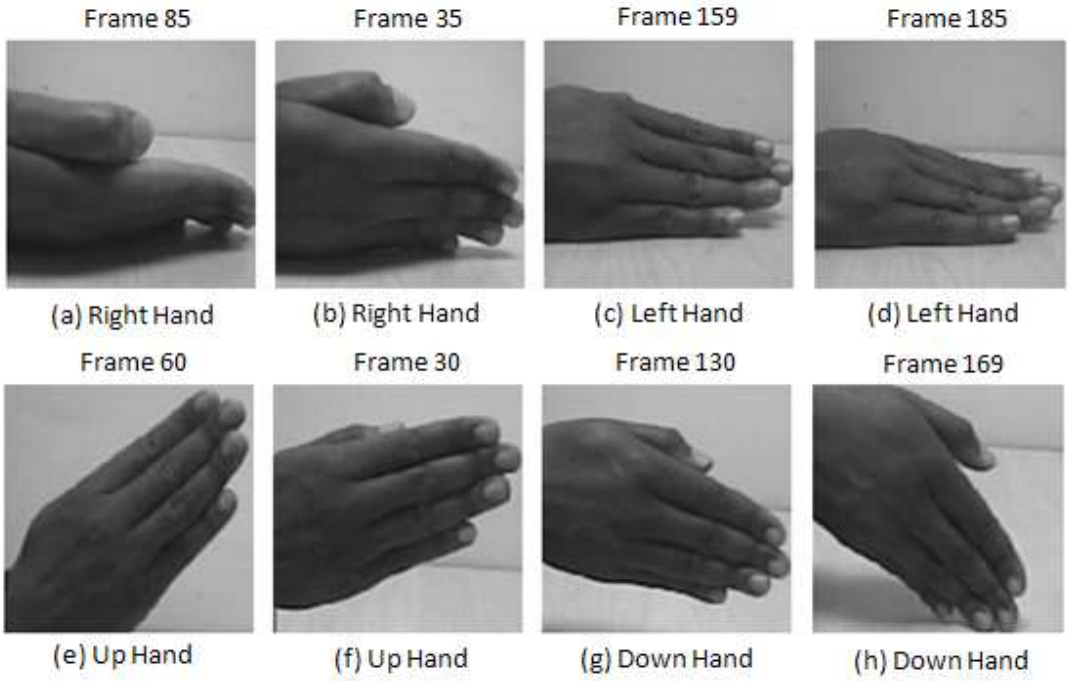

Figure 5-6: Range of right, left, up and down hand poses

5.2 Head-based intent recognition

For classification of single frame positions, a set of 650 frames was selected for each class (centre, right, left, up and down) through all the 20 subjects and divided in half to form a training set (through 10 subjects) and a testing set (through 10 subjects) made of 325 frames each. Figure 5-5 depicts the range within which a head is labelled right (from (a) to (b)), left (from (c) to (d)), up (from (e) to (f)) and down (from (g) to (h)). For intent recognition, groups of 10 frames are processed resulting in 10-point intention curves. For head rotation, the training set is made of 400 intention curve examples while the testing set also contains 400 intention curves used for validation,

and for vertical motion of the head, the training set comprises 600 intention curve examples while the testing set also contains 600 intention curves used for validation.

5.2.1 Performance for the recognition of the head in rotation: direction recognition

As depicted in Tables 5-2 and 5-3, good results are obtained because as mentioned in Section 5.1, in a hold out approach for performance evaluation an inefficient use of the data is made, giving a pessimistic recognition rate that is mostly above 80% in this work. This demonstrates the viability of the proposed algorithms as an alternative visual head pose estimation for direction intent recognition. For single frame head pose classification, it is evident that the symmetry-based approach combined with the difference of means of the resulting symmetry curve's COG yields the best recognition rate with 95.5%. The adaboost-based approach found in the literature [66], [67] and implemented for comparison, yields a slightly better recognition rate than all the proposed methods (95.3%) except the one previously referred to based on the difference of means of the symmetry curve's COG. It is also evident that the COG of a symmetry curve is a better pose indicator than the *y*-intercept of the line approximating that symmetry curve: Table 5-2 shows the recognition rate of 95.5% and 93.3% for the approach based on the difference of means and the approach based on the statistics in a Gaussian distribution respectively, both used on the symmetry curves COGs, against 92.4% and 92% for these same methods used on the *y*-intercept of the line approximating the resulting symmetry curve.

For head direction intent recognition, it can be observed that the proposed approach based on the statistics (mean and standard deviation) in a Gaussian distribution of COG-based intention curves, exhibits the best recognition rate with

93.7%. For each method, the centre class is the one where the recognition rate is the best. The proposed method by Jia and Hu in [66], [67], instead of looking at the motion of the head throughout a sequence of frames, looks at the position of the head in a single frame yielding the second best single frame pose classification rate of 95.3% as depicted in Table 5-2. For such a solution however, only the last frame in the sequence is used for intent recognition and the disadvantage is that back motions (from left to centre, and right to centre for right and left motions respectively) are misclassified, thus significantly affecting the overall results (refer to Table 5-3 where it displays the worst result with 72%). A modified version of the method in [66] that uses the full 10-frame video sequence for recognition rather than the last frame, is proposed, which yields better results (87. 7%).

Table 5-2: Single-frame pose classification rate of heads in rotation

Methods	Class	Training set	Testing set	Correct classification	Incorrect classification	Classification rate
Difference of means of symmetry curve's COG	Centre:	325	325	320	5	98. 5%
	Right:	325	325	286	39	88%
	Left:	325	325	325	0	100%
	Total:	975	975	931	44	95.5%
Statistics a in Gaussian distribution of symmetry curve's COG	Centre:	325	325	320	5	98.5%
	Right:	325	325	290	30	90.8%
	Left:	325	325	290	30	90.8%
	Total:	975	975	900	65	93.3%
Difference of means for *y*-intercepts (of lines approximating a symmetry curve)	Centre:	325	325	312	13	96%
	Right:	325	325	266	59	81.8%
	Left:	325	325	323	2	99.4%
	Total:	975	975	900	74	92.4%
Statistics in a Gaussian distribution of *y*-intercepts (of lines approximating a symmetry curve)	Centre:	325	325	306	19	94.1%
	Right:	325	325	266	59	81.8%
	Left:	325	325	325	0	100%
	Total:	975	975	897	78	92%
Adaboost (combined with nose template matching) [66]	Centre:	325	325	324	1	99.7%
	Right:	325	325	312	13	96%
	Left:	325	325	293	32	90.1%
	Total:	975	975	929	46	95.3%

It can also be observed that the intention curves based on the COGs of symmetry curves throughout a sequence is a slightly better intent indicator than the intention curve based on *y*-intercepts of the line approximating these symmetry curves when using the same approach (difference of means: 86.6% and 85% respectively, and statistics in Gaussian distribution: 93.7% and 93.4%).

Table 5-3: 10-frame intent recognition rate for heads in rotation

Methods	Class	Training set	Testing set	Correct classification	Incorrect classification	Classification rate
COG-based rotation detection using Difference of means	Centre:	400	400	400	0	100%
	Right:	400	400	326	74	81.5%
	Left:	400	400	313	87	78.2%
	Total:	1200	1200	1039	161	86.6%
COG-based rotation detection Statistics in a Gaussian distribution	Centre:	400	400	382	18	95.5%
	Right:	400	400	379	21	94.7%
	Left:	400	400	363	37	90.7%
	Total:	1200	1200	1124	76	93.7%
y-intercept-based rotation detection using Difference of means	Centre:	400	400	400	0	100%
	Right:	400	400	322	78	80.5%
	Left:	400	400	298	102	74.5%
	Total:	1200	1200	1020	180	85%
y-intercept - based rotation detection using Statistics in a Gaussian distribution	Centre:	400	400	391	9	97.7%
	Right:	400	400	373	27	93.2%
	Left:	400	400	357	43	89.2%
	Total:	1200	1200	1121	79	93.4%
Adaboost-based rotation detection [66]	Centre:	400	400	398	2	99.5%
	Right:	400	400	264	136	66%
	Left:	400	400	202	198	50.5%
	Total:	1200	1200	864	336	72%
Modified adaboost-based rotation detection	Centre:	400	400	398	2	99.5%
	Right:	400	400	322	78	80.5%
	Left:	400	400	332	68	83%
	Total:	1200	1200	1052	148	87.7%

5.2.2 Performance for the recognition of the head in vertical motion: speed variation recognition

As displayed in Tables 5-4 and 5-5, good results (for the same reasons given in Section 5.2.1) were obtained and demonstrate the viability of the proposed algorithms as an alternative visual head pose estimation for speed variation intent recognition. For single frame head pose classification, both methods perform very well with our proposed PCA-based approach yielding a slightly better recognition rate (97.8%) than the adaboost-based method proposed in [66] (96.8%). For each method, the centre class is the one where the recognition rate is the best with 100% recognition. For head-based speed variation intent recognition, it can be observed once more that our proposed PCA-based approach exhibits the better recognition rate with 91.2%. For each method, the centre class is again the one where the recognition rate is the best with 100% for the adaboost-based approach (including the modified version proposed in this thesis), against 93.7% for the proposed PCA-based approach.

As mentioned earlier, instead of looking at the motion of the head throughout a sequence of frames, the proposed method in [66] looks at the position of the head in a single frame. For such a solution, only the last frame in a sequence is used for intent recognition and the disadvantage is that back motions (from down to centre, and up to centre for up and down motions respectively) are misclassified, significantly affecting the overall results (refer to Table 5-5 where it displays the worst result with 61.2%). It can also be noted from Table 5-5 that the classification rate of this method for the ‘up’ class (24.2%) is much lower than the one for the ‘down’ class (59.3%). This difference can be explained by the fact that the ‘up’ motion is only detected by this method when the head has gone sufficiently far from its centred position, while the

'down' motion is already detected when the head is closer to the centred position. Note also that a modified version of the method in [66], which uses the full 10-frame video sequence for recognition rather than the last frame yields improved results (83. 8%).

Table 5-4: Single-frame pose classification rate of heads in vertical motion

Methods	Class	Training set	Testing set	Correct classification	Incorrect classification	Classification rate
PCA	Centre:	325	325	325	0	100%
	Up:	325	325	325	0	100%
	Down:	325	325	304	21	93.5%
	Total:	975	975	954	21	97.8%
Adaboost (combined with nose template matching) [66]	Centre:	325	325	325	0	100%
	Up:	325	325	305	20	93.8%
	Down:	325	325	314	11	96.6%
	Total:	975	975	944	31	96.8%

Table 5-5: 10-frame intent recognition rate for heads in vertical motion

Methods	Class	Training set	Testing set	Correct classification	Incorrect classification	Classification rate
PCA-based vertical motion detection	Centre:	600	600	562	38	93.7%
	Up:	600	600	560	40	93.3%
	Down:	600	600	520	80	86.7%
	Total:	1800	1800	1642	158	91.2%
Adaboost-based vertical motion detection [66]	Centre:	600	600	600	0	100%
	Up:	600	600	145	455	24.2%
	Down:	600	600	356	160	59.3%
	Total:	1800	1800	1101	615	61.2%
Modified Adaboost-based vertical motion detection	Centre:	600	600	600	0	100%
	Up:	600	600	482	118	80.3%
	Down:	600	600	426	174	71%
	Total:	1800	1800	1508	292	83.8%

5.3 Hand-based intent recognition

As described in Chapter 4, for classification of single frame positions, a set of 650 frames was selected for each class (centre, right, left, up and down) through all the 20 subjects and was divided into half to form a training set (through 10 subjects) and a testing set (through 10 subjects) comprising 325 frames each. Figure 5-6 depicts the range within which a hand is labelled right (from (a) to (b)), left (from (c) to (d)), up (from (e) to (f)) and down (from (g) to (h)). For intent recognition, groups of 10 frames are processed, resulting in intention curves. For both hand rotation and vertical motion, the training set consists of 600 intention curve examples while the testing set also contains 600 intention curves used for validation.

5.3.1 Performance for the recognition of the hand in rotation: direction recognition

As depicted in Tables 5-6 and 5-7, the task of classifying hands in rotation using the vertical symmetry-based approach is not as successful as for the face because it is not a symmetrical object. However, it yields a recognition rate far greater than 50% (78.5%, 81.5% and 81.2% for MLP, SVM and *k*-means respectively for single frame hand pose classification and 76.8%, 79.3% and 77.9% for MLP, SVM and *k*-means respectively for hand-based direction intent recognition) because the rotation of the hand to some extent presents separable symmetry curves, whose statistics can be learned by a machine learning approach (in this thesis a neural network, a support vector machine and *k*-means clustering are used). The template matching-based approach, however, performs better (93.4% for single frame hand poses classification

and 89% for hand-based direction intent recognition). These satisfactory results demonstrate the viability of the proposed algorithms as alternative visual hand pose estimation solutions for direction intent recognition.

Note that no method for comparison with our proposed method is found in the literature because typical hand gesture recognition method requires a more explicit motion of the hand than the micro-operation (rotation) defined in this thesis. It can also be observed that for the hand in rotation, the left pose and intent display the worst results due to the similarity in appearance between left hands and centred hands (refer to Figure 5-6) especially when the left hand is closer to the centre (refer to Figure 5-6.c). The entire false negatives for the left class are therefore found in the centre class.

Table 5-6: Single-frame pose classification rate of hands in rotation

Methods	Class	Training set	Testing set	Correct classification	Incorrect classification	Classification rate
MLP	Centre:	325	325	259	66	79.7%
	Right:	325	325	261	64	80.3%
	Left:	325	325	245	80	75.4%
	Total:	975	975	765	210	78.5%
SVM	Centre:	325	325	300	25	92.3%
	Right:	325	325	253	72	77.8%
	Left:	325	325	242	83	74.5%
	Total:	975	975	795	180	81.5%
KMEANS	Centre:	325	325	255	70	78.5%
	Right:	325	325	262	63	80.6%
	Left:	325	325	255	70	78.5%
	Total:	975	975	792	203	81.2%
Cross-correlation template matching	Centre:	325	325	325	0	100%
	Right:	325	325	315	10	96.9%
	Left:	325	325	271	54	83.4%
	Total:	975	975	911	64	93.4%

Table 5-7: 10-frame intent recognition rate for hands in rotation

Methods	Class	Training set	Testing set	Correct classification	Incorrect classification	Classification rate
MLP-based rotation detection	Centre:	600	600	493	107	82.2%
	Right:	600	600	470	130	78.3%
	Left:	600	600	419	181	69.8%
	Total:	1800	1800	1382	418	76. 8%
SVM-based rotation detection	Centre:	600	600	518	82	86.3%
	Right:	600	600	482	118	80.3%
	Left:	600	600	427	173	71.2%
	Total:	1800	1800	1427	373	79.3%
KMEANS-based rotation detection	Centre:	600	600	506	94	84.3%
	Right:	600	600	474	126	79%
	Left:	600	600	422	178	70.3%
	Total:	1800	1800	1402	398	77. 9%
Template matching-based rotation detection	Centre:	600	600	600	0	100%
	Right:	600	600	572	28	95.3%
	Left:	600	600	431	169	71.8%
	Total:	1800	1800	1603	197	89%

5.3.2 Performance for the recognition of the hand in vertical motion: speed variation recognition

As displayed in Tables 5-8 and 5-9, good results are obtained, demonstrating the viability of the proposed algorithms as alternative visual hand pose estimation methods for speed variation intent recognition. For single frame hand pose classification, the approach based on the ellipse shaped mask yields the best recognition rate (97.2%) and the HOG-based approach implemented for comparison with our proposed methods, displays the worst recognition rate (94%). For hand-based speed variation intent recognition, it can be observed once more that our ellipse shaped mask approach exhibits the best recognition rate with 94.7%.

Table 5-8: Single-frame pose classification rate of hands in vertical motion

Methods	Class	Training set	Testing set	Correct classification	Incorrect classification	Classification rate
Cross-correlation template matching	Centre:	325	325	325	0	100%
	Up:	325	325	293	32	90.1%
	Down:	325	325	325	0	100%
	Total:	975	975	943	32	96.7%
Ellipse shaped Mask based approach	Centre:	325	325	308	17	94.8%
	Up:	325	325	316	9	97.2%
	Down:	325	325	324	1	99.7%
	Total:	975	975	948	27	97.2%
HOG-based approach	Centre:	325	325	325	0	100%
	Up:	325	325	303	22	93.2%
	Down:	325	325	289	36	88.9%
	Total:	975	975	917	58	94%

Table 5-9: 10-frame intent recognition rate for hands in vertical motion

Methods	Class	Training set	Testing set	Correct classification	Incorrect classification	Classification rate
Template matching-based vertical motion detection	Centre:	600	600	600	0	100%
	Up:	600	600	507	93	84.5%
	Down:	600	600	521	79	86.8%
	Total:	1800	1800	1628	172	90.4%
Ellipse shaped Mask-based vertical motion	Centre:	600	600	600	0	100%
	Up:	600	600	549	51	91.5%
	Down:	600	600	556	44	92.7%
	Total:	1800	1800	1705	95	94.7%
HOG-based vertical motion detection	Centre:	600	600	545	55	90.8%
	Up:	600	600	559	41	93.2%
	Down:	600	600	549	51	91.5%
	Total:	1800	1800	1653	143	91.8%

5.4 Extrapolation for data efficiency

So far, intent recognition was performed by mapping a 10-frame video sequence to a 10-point vector referred to in this work as “intention curve” that is subsequently used

for the classification task. An experiment that consists of using fewer frames for intent recognition, and observing how the recognition rate is impacted, is performed. This experiment aims at reducing the amount of data used for recognition resulting in a faster and more data efficient recognition algorithm. However, instead of using fewer consecutive frames, a number n of frames are skipped throughout a sequence of 10 frames: In a set of 10 frames, every n frames where $n = \{1,2,3,4,5\}$ is considered, the intention curve consisting of only m points where $m \leq 10$ and $m = \{10,5,3,2,2\}$ is then obtained and used to extrapolate a 10-point intention curve.

The extrapolation is performed as follows: For every (x_a,y_a) *and* (x_b,y_b) two consecutive points in the intention curve, we determine a mean point (x_c,y_c) that we assume belongs to the intention curve such that $x_a < x_c < x_b$ and $y_a < y_c < y_b$ where $x_c = x_a+(x_b-x_a)\times\Pi$ and $y_c = y_a+(y_b-y_a)\times\Pi$, where Π is a uniformly distributed pseudo-random number ($0 \leq \Pi \leq 1$). This process is repeated until the curve contains 10 points.

A subset of our dataset was used for this experiment spanning all the 20 subjects using one long video sequence of each. Figures 5-7, 5-8, 5-9 and 5-10 illustrate the recognition rate changes where n frames are skipped (with $n = \{1,2,3,4,5\}$) for the head in rotation, the head in vertical motion, the hand in rotation and the hand in vertical motion respectively. It can be observed that for each case, the higher the value of n, the lower the recognition rate.

For head rotation, it can be observed that although the recognition rate decreases as n increases, it still remains above 75% and 70% for the proposed methods based on the difference of means and the statistics (mean and standard deviation) in a Gaussian distribution of COG-based intention curves respectively. The other methods including the method by Jia and Hu [66] decrease further to below 65%. For vertical motion of

the head, it is evident that the recognition rate decreases for both methods (PCA-based and adaboost-based); however, the overall performance remains above 70%. For hand rotation, the *k*-means-based approach excluded (where the recognition rate decreases almost to 60%), the overall recognition rate does not decrease below 70%. For vertical motion of the hand, both proposed methods (based on template matching and ellipse shaped mask) do not decrease below 85% while the method implemented for hand vertical motion comparison based on HOG decreases to nearly 80%.

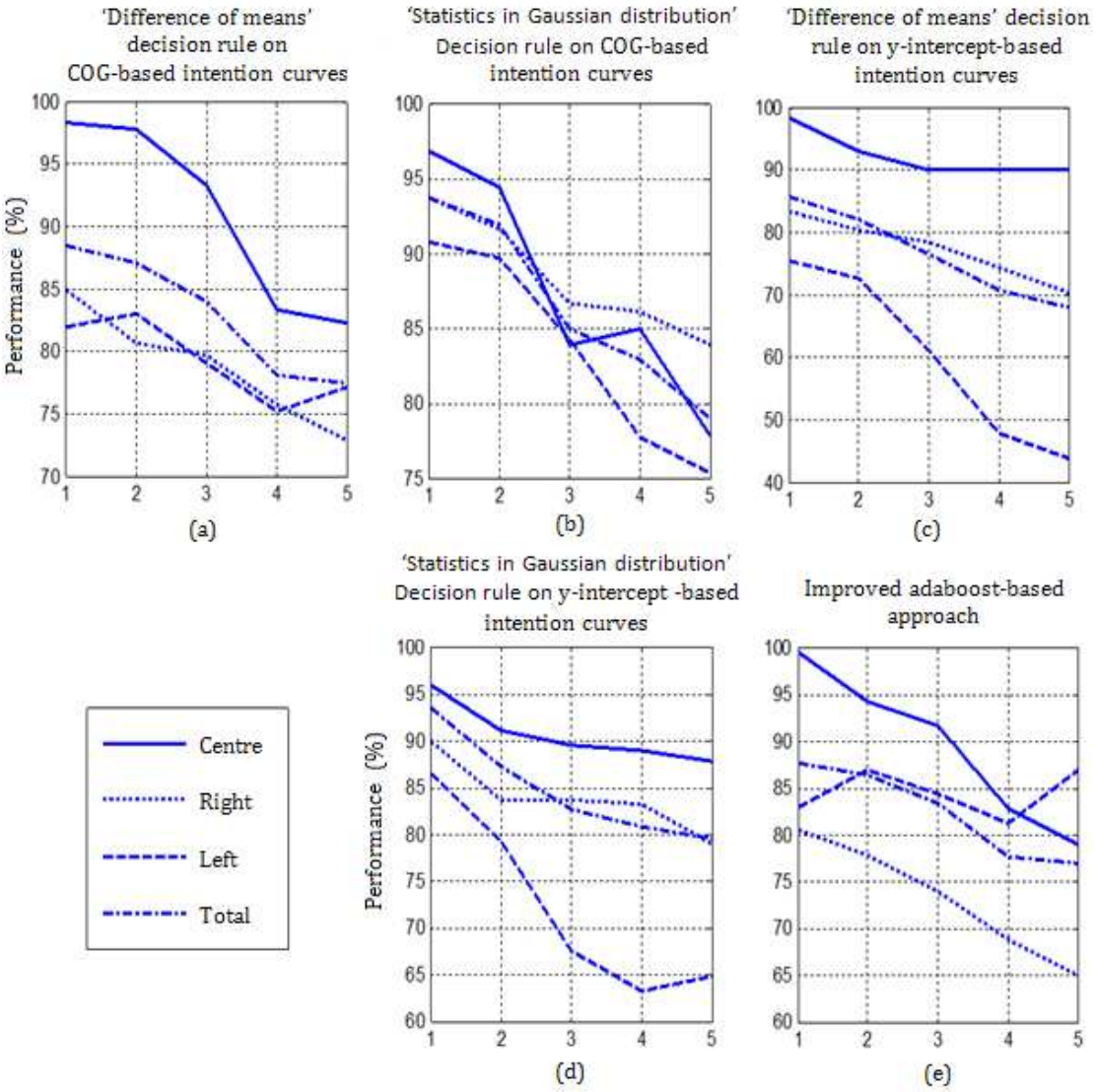

Figure 5-7: Recognition rates for heads in rotation for different numbers of frames skipped before selection.

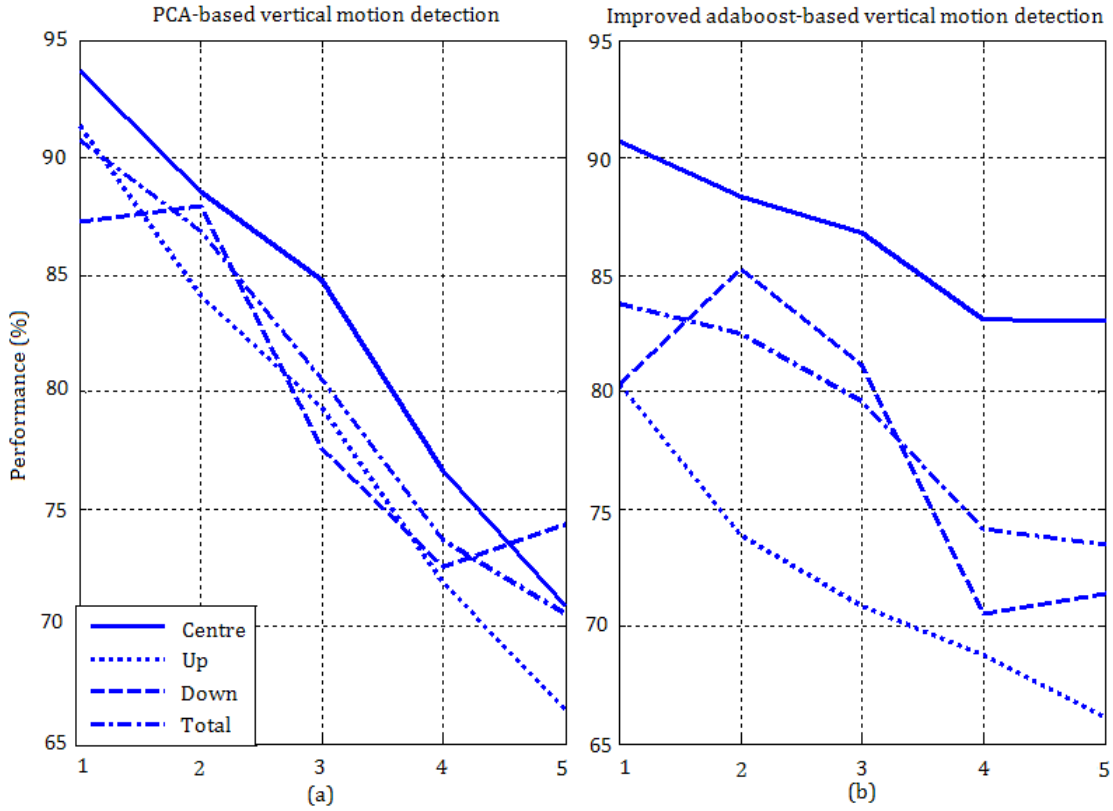

Figure 5-8: Recognition rates for heads in vertical motion for different numbers of frames skipped before selection

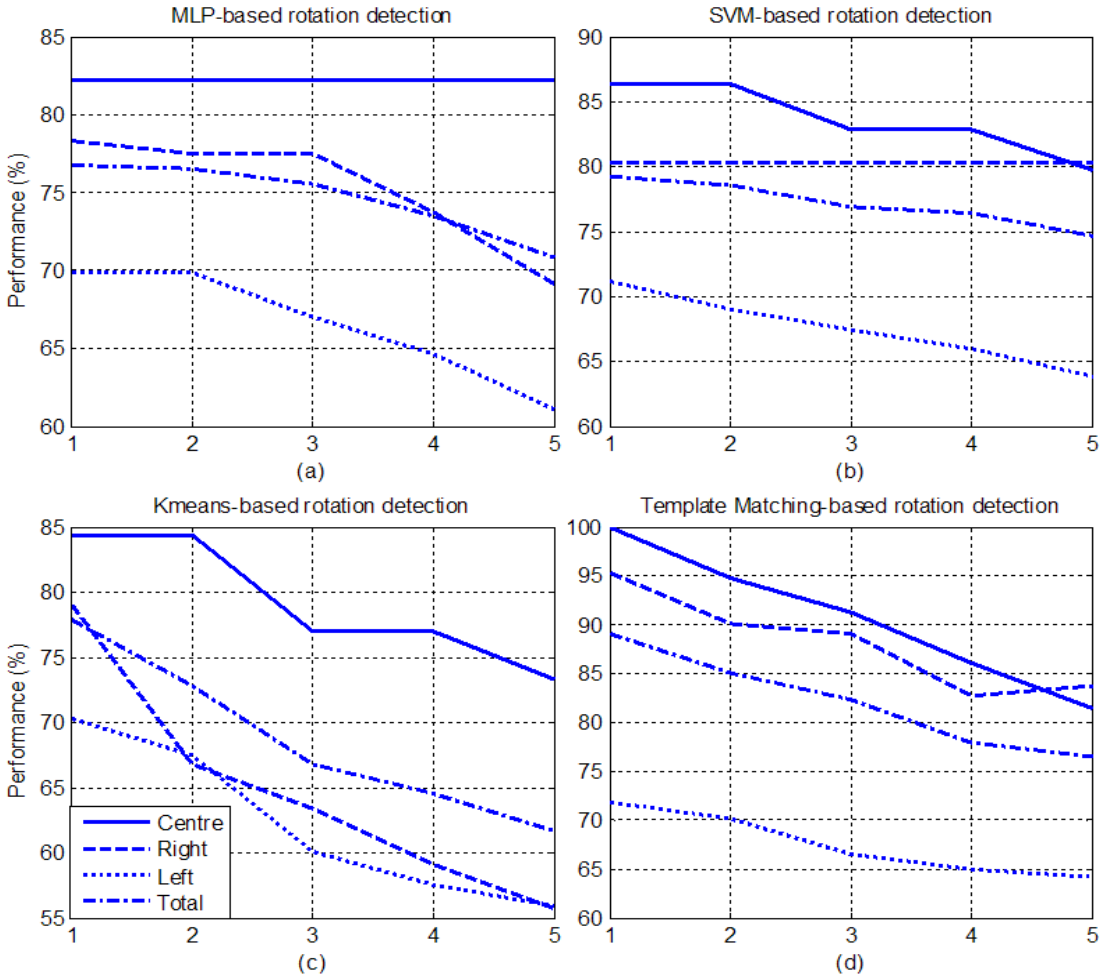

Figure 5-9: Recognition rates for hands in rotation for different numbers of frames skipped before selection

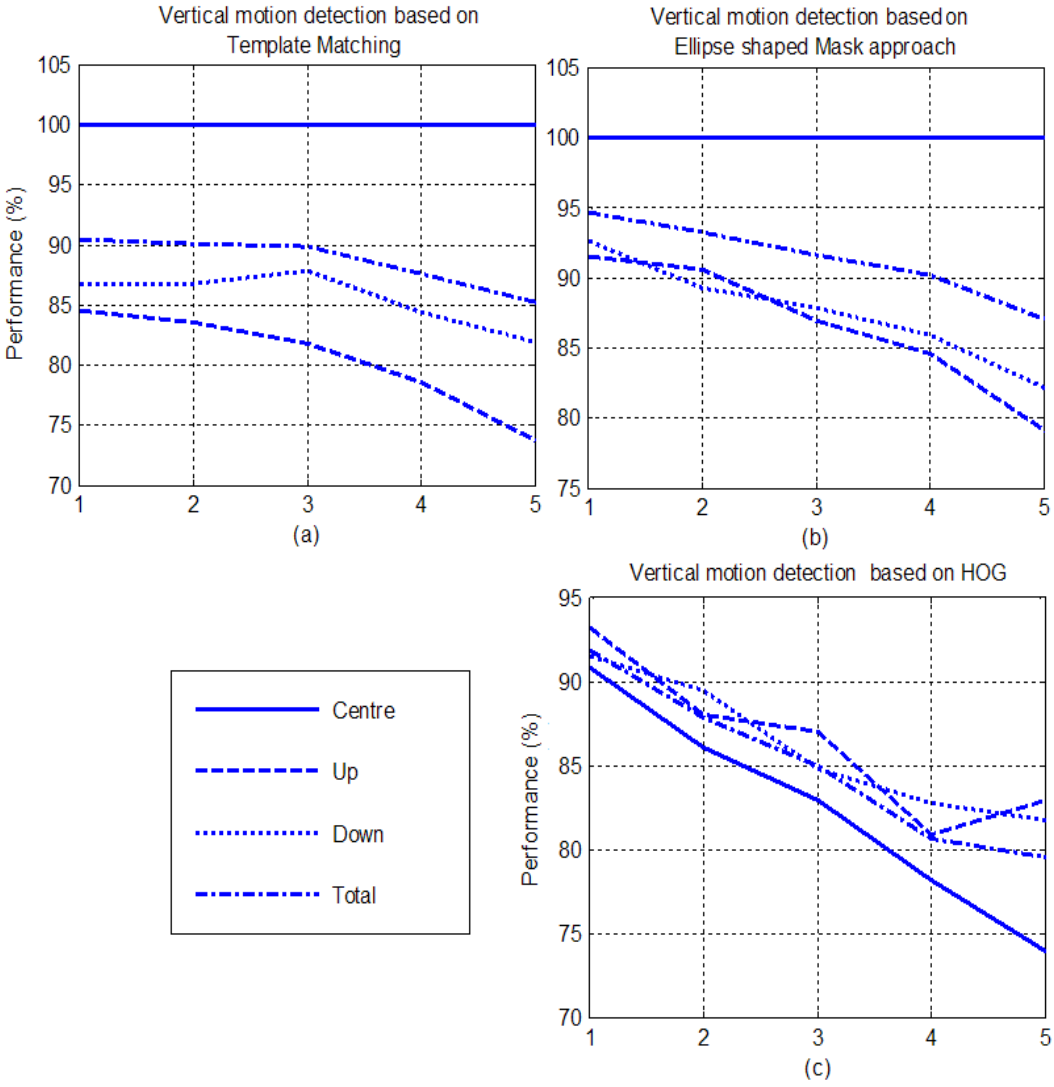

Figure 5-10: Recognition rates for hands in vertical motion for different numbers of frames skipped before selection

5.5 Concluding remarks

Overall, the methods proposed in this work yield good results. Three sets of results are given for each proposed and implemented method. The first set depicts the performance for single frame head and hand pose classification, the second set illustrates the performance for intent recognition through classification of intention curves obtained by processing 10 frames, and the third set demonstrates the performance of each method when fewer frames are used for recognition within a 10-frame video sequence.

In summary, it was observed that for single frame head pose classification the best method is the symmetry-based approach based on the difference of means of the resulting symmetry curve's COG; therefore the COG is a better pose indicator than the *y*-intercept of the line approximating that symmetry curve. For head direction intent recognition, the best approach is the proposed method based on the statistics (mean and standard deviation) in a Gaussian distribution of COG-based intention curves. For head in vertical motion, the proposed PCA-based approach performs better than the adaboost-based method [66], [67], both for single frame head pose classification and for intent recognition. For hand rotation our proposed template matching-based approach performs the best for both single frame head pose classification and intent recognition. Finally, for hands in vertical motion, our proposed ellipse shaped mask approach yields the best recognition rate.

It was also demonstrated that an attempt could be made for the proposed approaches to execute faster and be more data efficient by selecting only a few frames within the 10-frame video sequence, and extrapolating the 10-point intention curve

used for intent recognition. However, the recognition rate decreases revealing the necessity of a trade off that can be decided between recognition rate and data efficiency. It must also be noted that as the results reveal, better performance in pose classification does not necessarily mean better performance in intent recognition.

The next chapter furnishes a conclusion and propose some avenues that can be explored for future work.

Chapter 6

Conclusion

This thesis proposes an alternative visual solution for head and hand motion recognition aimed at intent recognition, intended to be applied to assistive living as a substitute to conventional wheelchair control devices such as joysticks, pneumatic switches or other control devices for wheelchair mobility. The input to the solution is a video sequence comprising 576×768 image frames captured from a charge-coupled device (CCD) camera (Hi-Resolution Dome Camera - 1/3" CCD, 470 TV lines, 0.8 lux, 3.6mm (F2.0) Lens) and a “25 frames per second” E-PICOLO-PRO-2 frame grabber. Results are obtained by collecting two sets of video sequences of 20 different subjects with five long sequences each, with the head and the hand in motion as objects of interest respectively. The video sequences of 10 subjects are used for all the training tasks, and the video sequences of the 10 others are used for validation. These long video sequences are divided into several 10-frame sequences for intention inference. As intent indicators, the objects of interest are the frontal view of the head in motion and the dorsal view (as opposed to the palm) of the hand in motion. Intent recognition is obtained using 10 frames with the head and the hand as the object of interest, and through the algorithms proposed in this thesis, these 10 frames are mapped to a 10-point intention curve that is subsequently classified using an appropriate decision rule. This work provides a contribution to the task of realising an

enabled environment allowing people with disabilities and the elderly to be more independent and as a result more active in society.

6.1 Summary of contributions

For head rotation recognition (refer to Section 3.3), a symmetry-based approach that maps a face to a symmetry curve is used. This symmetry curve is subsequently used in 4 different proposed methods according to the feature selection process (centre of gravity of the symmetry curve or *y*-intercept of the line approximating the symmetry curve) and the decision rule (based on difference of means or statistics in a Gaussian distribution) used for pose and intent recognition. For vertical motion recognition of the head (refer to Section 3.4), PCA is used for pose and intent recognition. A method proposed by Jia and Hu [66], [67] is also implemented for the purpose of comparison. The approach uses adaboost for face detection and profile pose estimation, camshift for tracking, and nose template matching for vertical pose detection.

For hand rotation recognition (refer to Section 4.3), a vertical symmetry-based approach is used where symmetry is calculated vertically rather than horizontally. The statistics (mean and standard deviation) of the resulting symmetry curves are subsequently used as 2D data features inputs to three different machine learning methods for classification: a neural network, a support vector machine and *k*-means clustering. Another method is proposed based on a normalised cross-correlation template matching. For the vertical motion of the hand (refer to Section 4.4), a mask in the shape of an ellipse is used to determine the vertical position of the hand. The other proposed approach is based on a normalised cross-correlation template matching for pose detection and a decision rule is used for intent recognition. For comparison of

the proposed methods for vertical motion of the hand, a feature selection approach found in the literature known as histogram of the oriented gradient is implemented for pose estimation. All these methods are combined with a decision rule to classify the resulting intention curve for intent classification.

Three sets of results are given for each proposed and implemented algorithm in this thesis. The first set focuses on the performance for single frame head and hand pose classification, the second set illustrates the performance for intent recognition through classification of intention curves obtained by processing 10 frames, and the third set shows the performance of each method when less frames are used for recognition within a 10-frame video sequence choosing every 2,3,4 or 5 frames rather than every frame. Overall, these techniques are simple to implement and yield very good results on the given validation set, indicating their merits.

In summary, it was observed that for single frame head pose classification for the head in rotation, the best method is the symmetry-based approach based on the difference of means of the resulting symmetry curve's COG with 95.5% recognition rate in the validation set. The COG is therefore a better pose indicator than the *y*-intercept of the line approximating that symmetry curve. For head direction intent recognition, the best approach is the proposed method based on the statistics (mean and standard deviation) in a Gaussian distribution of COG-based intention curves (93.7%). For head in vertical motion our proposed PCA-based approach performs better than the adaboost-based method [66], [67] for both single frame head pose classification and intent recognition (97.8% and 91.2% respectively). For hand rotation our proposed template matching-based approach performs the best for both single frame head pose classification and intent recognition (93.4% and 89% respectively). And finally for hand in vertical motion, our proposed ellipse shaped

mask approach yields the best recognition rates for both single frame hand pose classification and intent recognition (97.2% and 94.7% respectively).

It was also demonstrated that an attempt can be made to be more data efficient by selecting only a few frames within the 10-frame video sequence, and extrapolate the intention curve used for intent recognition. However, the recognition rate decreases revealing the necessity of a trade off that can be decided between recognition rate and data efficiency.

6.2 Concluding remarks

The head and the hand in motion are therefore useful intent indicators, and the proposed methods are able to recognise their motions as defined in this thesis for intent recognition. When compared to the work in [66] and [67], implemented in this work, it can be observed that overall, the proposed methods perform better and are therefore suitable alternative intention detection methods that can be applied to wheelchair motion.

Similarly to the above mentioned work in the literature the assumptions guiding the data acquisition process does not constitute a problem since the user who still retains the full use of her head and hand motion, should be trained to use the solution.

As mentioned in Section 1.6, this alternative visual solution is an important contribution because as shown in the literature, one of the most promising sensor technologies associated with assistive living application is machine vision and thus a successful implementation of visual solutions is increasingly favoured.

This thesis also shows how the symmetry property of the head can be used for motion understanding through a symmetry-based approach that is simple as opposed

to head pose estimation found in the literature that require sophisticated machine learning algorithm for recognition.

As shown in the literature, many head pose estimation and hand gesture recognition solutions used in different applications including wheelchair motion allow symbolic commands based on individual postures. The solution proposed in this thesis on the other hand, recognizes intents based on the motion contained in a specific number of frames (10 in this work) rather than the posture in a single frame. This brings the advantage that even if the position of the head and the hand is only loosely detectable, that is the exact pose cannot be quantified to determine which pose is left, right, up, down or centre; the different kinds of motion can still be robustly detected. The other advantage is that the misdetection of a single frame is less costly on the overall performance.

Gesture recognition solutions found in the literature are made possible looking at a change in the hand's contour shape and it is typically applied to sign language applications. The literature doesn't contain gesture recognition solutions where the motion of the hand is a micro-operation such as the rotation and vertical motion described in this thesis, for which the approaches found in the literature are typically invariant or unusable for robust classification. Furthermore this thesis proposes two novel methods for motion recognition of the hand in vertical motion. The first one makes use of an ellipse shaped mask for pose estimation and intention curves generation and the second one uses a HOG descriptor.

6.3 Future work

As the scope (refer to Section 1.5) of this thesis reveals, there are still some avenues that can be explored in this work:

Proposed solutions applied to people with disabilities: Though intended for a wheelchair mobility application, the algorithm has not been tested on actual people with disabilities. This work is limited to the implementation of intent recognition algorithms using recorded video sequences of subjects sited on an office chair (to mimic a person with a physical disability in a wheelchair) and performing the four types of motion described in this thesis. With the proper legal requirements, consisting of approvals from the university ethical committee, that of the research ethic group at provincial and district level as well as the hospital's, more work can consist of testing these algorithms on real people with disabilities whose mobility is made possible by the use of a wheelchair.

Real-time implementation: The algorithm was tested on recorded video sequences. Real-time implementation is also an important extension of this work.

Data fusion: The head and the hand are independent indicators for the same type of motions. No data fusion scheme was used to combine these two motion indicators. A data fusion scheme looking at both motions rather than one at a time can also be the object of further investigation.

Performance comparison between a joystick and the proposed intent indicators: A study can be conducted to compare the proposed solutions that are head-based and hand-based indicators for wheelchair motion, with a joystick. This comparison can be conducted from the perspective of performance, ease of use and speed of response.

List of Publications

Conference Proceedings:

1. Luhandjula T., Monacelli E., Hamam Y., van Wyk B.J., Williams Q., 2009. Visual Intention Detection for Wheelchair Motion. *In: Proceedings of the 5th International Symposium on Visual Computing (ISVC), Springer-Verlag Las Vegas, USA,* 407-416.
2. Luhandjula T., Hamam Y., van Wyk B.J., Williams Q., 2009. Symmetry-based head pose estimation for intention detection. *In: Proceedings of the 20th Annual Symposium of the Pattern Recognition Association of South Africa, Stellenbosch, South Africa,* 93-98.
3. Luhandjula T., Djouani K., Hamam Y., van Wyk B.J., Williams Q, 2010. A hand-based visual intent recognition algorithm for wheelchair motion. *In: Proceedings of the 3rd International IEEE Conference on Human System Interactions, Rzeszow, Poland,* 749-756.
4. Luhandjula T., Williams Q., Hamam Y., Djouani K., van Wyk B.J, 2010. Visual head pose estimation algorithm for fast intent recognition. *In: Proceedings of the 21st Annual Symposium of the Pattern Recognition Association of South Africa, Stellenbosch, South Africa,* 165-170.

Book Chapter:

Luhandjula T., Djouani K., Hamam Y., van Wyk B.J., Williams Q, 2011. A visual hand motion detection algorithm for wheelchair motion. *Z.S. Hippe, J.L. Kulikowski*

(Editors), Human-Computer Systems Interaction. Backgrounds and Applications 2, Rzeszow, Poland Springer-Verlag Co. in the series Advances in Soft Computing.

Bibliography

1. Fai Y.C., Amin H.M., Fisal N., Su, E.L.M., 2005. Development and evaluation of various modes of human robot interface for mobile robot. *In: Proceedings of the 9th International Conference on Mechatronics Technology, Kuala Lumpur, Malaysia.*
2. Jaimes A., Sebe N., 2007. Multimodal human–computer interaction: a survey. *Computer Vision and Image Understanding*, 108:116-134.
3. Mei T., Hua X.S., 2005. Intention-based home browsing. *In: Proceedings of the 13th annual ACM International Conference on Multimedia, Singapore.*
4. Bell B., Franke J., Mendenhall H., 2000. Leveraging task models for team intent inference. *In: Proceedings of the International Conference on Artificial Intelligence.*
5. Lesh N., Rich C., Sidner C.L., 1999. Using plan recognition in human–computer collaboration. *In: Proceedings of the 7th International Conference on User Modelling, Banff, Canada.*
6. Aarno D.K.E., 2007. Intention recognition in human machine collaborative systems. *Licentiate Thesis, Stockholm Sweden.*
7. Grauman K., Betke M., Lombardi J., Gips J., Bradski G., 2003. Communication via eye blinks and eyebrow raises: video-based human-computer interfaces. *Universal Access in the Information Society*, 2(4):359-373.
8. Burgoon J., Adkins M., Kruse J., *et al*., 2005. An approach for intent identification by building on deception detection. *In: Proceedings of the 38th Hawaii International Conference on System Sciences.* 1:21a.

9. Baklouti M., Monacelli E., Guitteny V., Couvet S., 2008. Intelligent assistive exoskeleton with vision based interface. *Lecture Notes in Computer Science*, 5120:123-135.
10. Junior V.G., Parikh S.P., Okamoto J., 2006. Hybrid deliberative/reactive architecture for human-robot interaction. *In: Proceedings of the ABCM Symposium Series in Mechatronics*. 2:563-570.
11. Schmidt C.F., Sridharan N.S., Goodson J.L., 1978. The plan recognition problem: an intersection of psychology and artificial intelligence. *Artificial Intelligence*, 11(1-2):45-83.
12. Allen J.F., Perrault R., 1980. Analyzing intention in utterances. *Artificial Intelligence*, 15(3):143-178.
13. Carberry S., 1990. Plan recognition on natural language dialogue, *MIT Press*.
14. Pynadath D.V., Wellman M.P., 1995. Accounting for context in plan recognition, with application to traffic monitoring. *In: Proceedings of the 11th International Conference on Uncertainty in Artificial Intelligence.*
15. Charniak E., Goldman R., 1989. A semantic for probabilistic quantifier-free first-order languages, with particular application to story understanding. *In: Proceedings of the 11th International Joint Conference on Artificial Intelligence, Detroit, Michigan, USA*, 1074-1079.
16. Charniak E., Goldman R., 1993. A Bayesian model of plan recognition. *Artificial Intelligence*, 64(1):53-79.
17. Albrecht D.W., Zukerman I., Nicholson A.E., 1998. Bayesian models for keyhole plan recognition in an Adventure Game. *User Modelling and Use-Adapted Interaction*, 8(1-2):5-47.

18. Geib C.W., Goldman R.P., 2001. Plan recognition in intrusion detection systems. *In: Proceedings of the 2nd DARPA Information Survivability Conference and Exposition, Anaheim, California, USA,* 1:46-55.

19. Huber, M.J., Durfee E.H., Wellman M.P., 1994. The automated mapping of plans for plan recognition. *In: Proceedings of the 10th International Conference on Uncertainty in Artificial Intelligence.*

20. Kaminka G., Pynadath D.V., Tambe M., 2002. Monitoring teams by overhearing: a multi-agent plan recognition approach. *Journal of Artificial Intelligence Research*, 17:83-135.

21. Kautz H.A., Allen J.F., 1986. Generalized plan recognition. *In: Proceedings of the 5th National Conference on Artificial Intelligence, San Mateo, CA, USA,* 32-37.

22. Jiang Y.F., Ma N., 2002. Plan recognition algorithm based on plan knowledge graph. *Journal of Software*, 13(4):686-692.

23. Ivanov Y., Bobick A., 2000. Recognition of visual activities and interactions by stochastic parsing. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 22(8):852-872.

24. Austin K.B., Rose G.M., 1997. Automated behaviour recognition using continuous-wave Doppler radar and neural networks. *In: Proceedings of the 19th Annual International Conference of the IEEE Engineering in Medicine and Biology Society: Magnificent Milestones and Emerging Opportunities in Medical Engineering, Chicago, Illinois, USA,* 4:1458-1461.

25. Pynadath D.V., Wellman M.P., 2000. Probabilistic state-dependent grammars for plan recognitions. *In: Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence*, 507-514.

26. Patterson D., Liao L., Fox D., Kautz H., 2003. Inferring high level behaviours from low level sensors. *In: Proceedings of the 5th Annual Conference on Ubiquitous Computing (UBICOMP), Seattle, Washington, USA,* 73-89.

27. Intille S.S., Bobick A.F., 2001. Recognising planned multi-person action. *Computer Vision and Image Understanding,* 81(3):414-445.

28. Gonzalez A.J., Gerber W.J., DeMara R.F., Georgiopoulos M., 2004. Context-driven near-term intention recognition. *Journal of Defence Modelling and Simulation,* 1(3):122-143.

29. Erhard M.J., 2007. Visual intent recognition in a multiple camera environment. *A Thesis Submitted in Partial Fulfilment of the Requirements for the Degree of Master of Science in Computer Engineering*.

30. Yasuhiro I., Hirosi S., Hideyuki A., Shuitiro O., Tatsuo U., 1995. Behaviour-based intention inference for intelligent robots cooperating with human. *In: Proceedings of the 4th IEEE International Conference on Fuzzy Systems and the 2nd International Fuzzy Engineering Symposium, Yokohama , Japan,* 3:1695-1700.

31. Tavakkoli A., Kelley R., King C., Nicolescu M., Bebis G., 2008. A Visual tracking framework for intent recognition in videos. *Advances in Visual Computing*. 450-459.

32. Kelley R., Nicolescu M., Tavakkoli A., King C., Bebis G., 2008. Understanding human intentions via Hidden Markov Models in autonomous mobile robots. *In: Proceedings of the 3rd ACM/IEEE International Conference on Human robot interaction, Amsterdam, Netherlands*. 367-374.

33. Starner T., Pentland A., 1995. Real-time American Sign Language recognition from video using Hidden Markov Models. *In: Proceedings of International Symposium on Computer Vision, Coral Gables, Florida, USA,* 265-270.

34. Wilson A., Bobick A., 1998. Recognition and interpretation of parametric gesture. *In: Proceedings of the International Conference on Computer Vision, Bombay, India,* 329-336.

35. Brand M., Oliver N., Pentland A., 1997. Coupled Hidden Markov Models for complex action recognition. *In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Juan, Puerto Rico,* 994-999.

36. Bobick A., Ivanov Y.A., 1998. Action recognition using probabilistic parsing. *In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Santa Barbara, California,* 196-202.

37. Bui H.H., Venkatesh S., West G., 2002. Policy recognition in the abstract Hidden Markov Model. *Journal of Artificial Intelligence Research,* 17:451-499.

38. Bui H.H., 2003. A general model for online probabilistic plan recognition. *In: Proceedings of the 8th International Joint Conference on Artificial Intelligence,* 18:1309-1318.

39. Kiefer P., Schlieder C., 2007. Exploring context-sensitivity in spatial intention recognition. *In: Proceedings of the Workshop on Behaviour Monitoring and Interpretation, Osnabrück, Germany,* 102-116.

40. Nakauchi Y., Noguchi K., Somwong P., Matsubara T., Namatame A., 2003. Vivid room: human intention detection and activity support environment ubiquitous autonomy. *In: Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems, Las Vegas, Nevada,* 1:773-778.

41. Mei T., Hua X.S., Zhou H.Q., 2005. Tracking user's capture intention: a novel complementary view for home video content analysis. *In: Proceedings of the 13th annual ACM International Conference on Multimedia, Singapore.*

42. Mei T., Hua X.S., Zhou H.Q., *et al*. 2005. To mine capture intention of camcorder users. *Visual Communications and Image Processing, 5960, Bellingham, WA: SPIE.*

43. Wu C., Aghajan H., 2008. Head pose and trajectory recovery in uncalibrated camera networks – region of interest tracking in smart home applications. *In: Proceedings of the ACM/IEEE International Conference on distributed smart cameras, Stanford, CA, USA,* 1-7.

44. Khezri M., Jahed M., 2007. Real-Time intelligent pattern recognition algorithm for surface EMG. *Biomedical Engineering Online,* 6(45).

45. Salvucci D.D., 2004. Inferring driver intent: A case study in lane-change detection. *In: Proceedings of the 48th Annual Meeting of the Human Factors Ergonomics Society, Santa Monica, CA, USA.*

46. Geib C.W., 2002. Problems with intent recognition for elder care. *In: Proceedings of the Conference of the Association for the Advancement of Artificial Intelligence Conference.* 13-17.

47. Bauer M., Dempster-Shafer A., 1995. Approach to modelling agent preferences for plan recognition. *User Modelling and User-Adapted Interaction*. 5(3-4):317-348.

48. Bauer M., 1996. Acquisition of user preferences for plan recognition. *In: Proceedings of the 5th International Conference on User Modelling, Hawaii, USA.*

49. Weiss S.M., Indurkhya N., 1995. Rule-based machine learning methods for functional prediction. *Journal of Artificial Intelligence Research.* 3:383-403.

50. Pentland A., Liu A., 1999. Modelling and prediction of human behaviour. *Neural computation*, 11(1):229-42.

51. Bodor R., Morlok R., Papanikolopoulos N., 2004. Dual-camera system for multi-level activity recognition. *In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1:643-648.

52. Lin L., Patterson D.J., Fox D., Kautz H., 2004. Behaviour recognition in assisted cognition. *In: Proceedings of the 9th National Conference on Artificial Intelligence, San Jose, California.*

53. Braga R.A.M., Petry M., Moreira A.P., Reis L.P., 2008. Intellwheels: a development platform for intelligent wheelchairs for disabled people. *In: Proceedings of the 5th International Conference on Informatics in Control, Automation and Robotics, Madeira, Portugal,* 115-121.

54. Tzafestas S.G., 2001. Reinventing the wheelchair: autonomous robotic wheelchair projects in Europe improve mobility and safety. *IEEE Robotics and Automation Magazine.*8:1.

55. Simpson R.C., 2005. Smart wheelchairs: A literature review. *Journal of Rehabilitation Research & Development.* 42(4):423-436.

56. Yu H., Spenko M., Dubowsky S., 2003. An adaptive shared control system for an intelligent mobility aid for the elderly. *Autonomous Robots.* 15(1):53-66.

57. Aigner P., McCarragher B.J., 2000. Modelling and constraining human interactions in shared control utilizing a discrete event framework. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans.* 30(3):369-379.

58. Martens C., Ruchel N., Lang O., Ivlev O., GrÄaser A., 2001. A friend for assisting handicapped people. *IEEE Robotics & Automation Magazine.* 8(1):57-65.

59. Carlson T., Demiris Y., 2008. Human-wheelchair collaboration through prediction of intention and adaptive assistance. *In: Proceedings of the IEEE International Conference on Robotics and Automation, Pasadena, CA.* 3926-3931.

60. Benfold B., Reid I., 2008. Colour invariant head pose classification in low resolution video. *In: Proceedings of the 19th British Machine Vision Conference, Leeds, UK.*

61. Vatahska T., Bennewitz M., Behnke S., 2007. Feature-based head pose estimation from images. *In: Proceedings of the IEEE-RAS 7th International Conference on Humanoid Robots (Humanoids), Pittsburgh, Pennsylvania, USA,* 330–335.

62. Gourier N., Maisonnasse J., Hall D., Crowley J.L., 2007. Head pose estimation on low resolution images. *Multimodal Technologies for Perception of Human. Springer Berlin/Heidelberg.*

63. Hoey J., Gunn D., Mihailidis A., *et al*. 2006. Obstacle avoidance wheelchair system. *In*: *Proceedings of the International Conference on Robotics and Automation, Orlando, Florida, USA.*

64. Felzer T., Nordman R., 2007. Alternative wheelchair control. *In: Proceedings of the International IEEE-BAIS Symposium on Research on Assistive Technologies, Dayton, OH, USA.* 67-74.

65. Demeester E., Hüntemann A., Vanhooydonck D., Vanacker G., Van Brussel H., Nuttin M., 2008. User-adapted plan recognition and user-adapted shared control:

a Bayesian approach to semi-autonomous wheelchair driving. *Autonomous Robots*, 24(2):193-211.

66. Jia P., Hu H., 2005. Head gesture based control of an intelligent wheelchair. *In: Proceedings of the Conference of the Chinese Automation and Computing Society in the UK.*
67. Hu H.H., Jia P., Lu T., Yuan K., 2007. Head gesture recognition for hands-free control of an intelligent wheelchair. *Industrial Robot: An International Journal*, 34(1):60-68.
68. Bell D.A., Levine S.P., Koren Y., Jaros L., Borenstein J., 1993. An identification technique for adaptive shared control in human-machine systems. *In: Proceedings of the 15th Annual International Conference of the IEEE.* 1299-1300.
69. Bell D.A., Borenstein J., Levine S.P., Koren Y., Jaros L., 1994. An assistive navigation system for wheelchairs based upon mobile robot obstacle avoidance. *In*: *Proceedings of the IEEE transaction on Robotics and Automation, San Diego CA*, 3:2018-2022.
70. Simpson R., Levine S.P., Bell D.A., Koren Y., Borenstein J., Jaros L.A., 1995. The NavChair assistive navigation system. *In: Proceedings of the International Joint Conference on Artificial Intelligence, Montréal, Québec, Canada.*
71. Ju J.S., Shin Y., Kim E.Y., 2009. Vision based interface system for hands free control of an intelligent wheelchair. *Journal of NeuroEngineering and Rehabilitation*, 6:33.
72. Demeester E., Nuttin M., Vanhooydonck D., Brussel, H.V., 2003. Assessing the user's intent using Bayes' rule: application to wheelchair control. *In*: *Proceedings*

of the first International Workshop on Advanced in Service Robotics, Bardolino, Italy, 117-124.

73. Braga R.A.M., Petry M., Oliveira E., Reis L.P., 2008. Multi-level control of an intelligent wheelchair in a hospital environment using a cyber-mouse simulation system. *In*: *Proceedings of the International Conference on Informatics in Control Automation and Robotics, Madeira, Portugal*. 179-182.
74. Urdiales C., Peula J.M., Barrué C., Pérez E.J., Sánchez-Tato I., *et al*. 2008. A new collaborative-shared control strategy for continuous elder/robot assisted navigation. *Gerontechnology*, 7(2):229.
75. Carlson T., 2006. Adaptive shared control for an intelligent wheelchair. *Initial Research Plan*.
76. Nelisse M.W., 1998. Integration strategies using a modular architecture for mobile robots in the rehabilitation field. *Journal of Intelligent and Robotic Systems, Journal of Intelligent and Robotic Systems*, 22(3-4):181-190.
77. Mazo M., *et al*., 2001. An integral system for assisted mobility. *IEEE Robotics & Automation Magazine*, 8:46-56.
78. García J.C., Mazo M., Bergasa L.M., Ureña J., Lazaro J.L., Escudero M., Marron M., Sebastian E., 2005. Human-machine interfaces and sensory systems for an autonomous wheelchair. *Assistive Technology on the Threshold of the New Millennium. Assistive Technology Research Series*, 6:272-277.
79. Bergasa L.M., Mazo M., Gardel A., Berea R., Boquete L., 2000. Commands generation by face movements applied to the guidance of a wheelchair for handicapped people. *In: Proceedings of the 15th International Conference*, 4:660-663.

80. Taylor P.B., Nguyen H.T., 2003. Performance of a head-movement interface for wheelchair control. *In: Engineering in Medicine and Biology Society. Proceedings of the 25th Annual International Conference of the IEEE,* 17-21(2): 1590-1593.

81. Evans D.G., Drew R., Blenkhorn P., 2000. Controlling mouse pointer position using an infrared head-operated joystick. *IEEE Transactions on Rehabilitation Engineering,* 8:107-117.

82. Ma B., Zhang W., Shan S., Chen X., Gao W., 2006. Robust head pose estimation using LGBP. *In: Proceedings of the 18th International conference on pattern recognition, Hong Kong, China,* 2:512-515.

83. Oka K., Sato Y., Nakanishi Y., Koike H., 2005. Head pose estimation system based on particle filtering with adaptive diffusion control. *IEICE Transactions on Information and Systems,* 8:1601-1613.

84. Cootes T.F., Walker K., Taylor C.J., 2000. View-based active appearance models. *In: Proceedings of the 4th International Conference on Automatic Face and Gesture Recognition, Grenoble, France*, 227-232.

85. Li S.Z., Lu X., Zhang H. 2001. View-Based Clustering of Object Appearances Based on Independent Subspace Analysis. *In: Proceeding of the 8th IEEE International Conference on Computer Vision, Vancouver, Canada*, 9-12.

86. Brown L.M., Tian Y., 2002. Comparative studies of coarse head pose estimation. *In: Proceedings of the IEEE Workshop on Motion and Video Computing, Washington, DC, USA,* 125.

87. Wei Y., Fradet L., Tan T., 2001. Head pose estimation using Gabor eigenspace modelling. *Technical report, National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Science,* 7.

88. Gee A., Cipolla R., 1994. Determining the gaze of faces in images. *Image and Vision Computing.* 12(10).

89. Chen Q., Wu H., *et al*. 1998. 3D head pose estimation without feature tracking. *In: Proceedings of the 3rd IEEE International Conference on Automatic Face and Gesture Recognition, Nara Japan.* 88-93.

90. Tian Y., Brown L., Connell J.H., Pankanti S., Hampapur A., Senior A.W., Bolle R.M., 2003. Absolute head pose estimation from overhead wide-angle cameras. *In: Proceedings of the IEEE International Workshop on Analysis and Modelling of Faces and Gestures, Nice, France,* 92-99.

91. Voit M., Nickel K., Stiefelhagen R., 2006. Bayesian approaches for multi-view head pose estimation. *In: Proceedings of the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems,* 31-34.

92. Wang C., Brandstein M., 2000. Robust head pose estimation by machine learning. *In: Proceedings of the International Conference on Image Processing (ICIP), Vancouver, BC, Canada,* (3)210-213.

93. Robertson N., Reid I., 2006. Estimating gaze direction from low-resolution faces in video. *Lecture Notes in Computer Science,* 3952:402-415.

94. Niyogi S., Freeman W.T., 1996. Example-based head tracking. *In: Proceedings 2nd IEEE International Conference on Automatic Face and Gesture Recognition, Killington, Vermont, USA,* 374-378.

95. Ba S.O., Odobez J.M., 2005. Evaluation of multiple cue head poses estimation algorithms in natural environments. *In: Proceedings of the IEEE International Conference on Multimedia and Expo, Amsterdam, Netherlands,* 1330-1333.

96. Wu Y., Toyama K., 2000. Wide-range, person- and illumination-insensitive head orientation estimation. *In: Proceedings of the 4th IEEE International Conference on Automatic Face and Gesture Recognition. Grenoble, France,* 183-188.

97. Pappu R., Beardsley P.A., 1998. A qualitative approach to classifying gaze direction. *In: Proceedings of the 3rd International Conference on Face & Gesture, Nara, Japan,* 160-165.

98. Birchfield S., 1998. Elliptical head tracking using intensity gradients and colour histograms. *In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Santa Barbara, CA, USA,* 232-237.

99. Russakoff D.B., Herman M., 2002. Head tracking using stereo. *Machine Vision Applications,* 13(3):164.173.

100. Matthews I., Baker S., 2004. Active appearance models revisited. *International Journal of Computer Vision,* 60(2):135-164.

101. Rowley H., Baluja S., Kanade T., 1998. Neural network based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 20(1):23-38

102. Voit M., Nickel K., Stiefelhagen R., 2007. Head pose estimation in single- and multi-view environments – results on the CLEAR'07 Benchmarks. *In: Proceedings of the 2nd International Evaluation Workshop on Classification of Events, Activities and Relationships, Baltimore, MD, USA,* 4625:307-316.

103. Turk M., Pentland A.P., 1991. Face recognition using eigenfaces. *In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Maui, Hawaii, USA,* 586-591.

104. Huang K.S., Trivedi M.M., Gandhi T., 2003. Driver head pose and view estimation with single omnidirectional video stream. *In: Proceedings of the IEEE Intelligent Vehicle Symposium, Columbus, Ohio, USA.*

105. Turk M., Pentland A.P., 1991. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71-86.

106. Belhumeur P.N., Hespanha J.P., Kriegman D.J., 1997. Eigenfaces vs. Fisherfaces: recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711-720.

107. Fu Y., Huang T.S., 2006. Graph embedded analysis for head pose estimation. *In: Proceedings of the 7th IEEE International Conference on Automatic Face and Gesture Recognition, Southampton, UK*, 3-8.

108. Tu J.L., Fu Y., Hu Y.X., Huang T.S., 2006. Evaluation of head pose estimation in studio data. *In: Proceedings of the Classification of Events, Activities and Relationships Evaluation Workshop, England.*

109. Chen L.B., Zhang L., Hu Y.X., Li M.J., Zhang H.J., 2003. Head pose estimation using fisher manifold learning. *In: Proceedings of the IEEE International Workshop on Analysis and Modelling of Faces and Gestures, Nice, France*, 203-207.

110. Roweis S.T., Saul L.K., 2000. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323-2326.

111. Belkin M., Niyogi P., 2001. Laplacian eigenmap and spectral techniques for embedding and clustering. *In: Proceedings of Advances in Neural Information Processing Systems, Vancouver, British Columbia, Canada.*

112. Müller K.R., Mika S., Rätsch G., Tsuda K., Schölkopf B., An introduction to kernel-based learning algorithms. *IEEE Transactions on Neural Networks*, 12(2):181-201.

113. Li S.Z., Fu Q.D., Gu L., Scholkopf B., Cheng Y.M., Zhang H.J., 2001. Kernel machine based learning for multi-view face detection and pose estimation. *In:*

Proceedings of the 8th IEEE International Conference on Computer Vision. Vancouver, Canada, 2:674-679.

114. Li Z., Gao L., Katsaggelos A.K., 2006. Locally embedded linear subspaces for efficient video indexing and retrieval. *In: Proceedings of IEEE International Conference on Multimedia and Expo, Toronto, Ontario, Canada,* 1765-1768.
115. Yang R., Zhang Z., 2002. Model-based head pose tracking with stereovision. *In: Proceedings of the 5th International Conference on Automatic Face and Gesture Recognition, Washington, D.C., USA,* 255-260.
116. La Cascia M., Isidoro J., Sclaroff S., Head tracking via robust registration in texture map images. *In: Proceedings of the Conference on Computer Vision and Pattern Recognition, Santa Barbara, California, USA,* 508-514.
117. Brown L.M., 2001. 3D head tracking using motion adaptive texture-mapping. *In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Kauai, Hawaii, USA,* 1:998-1003.
118. Ke W., Yanlai W., Baocai Y., Dehui K., 2003. Face pose estimation with a knowledge-based model. *In: Proceedings of the IEEE International Conference Neural Networks and Signal Processing, Nanjing, China,* 2:1131-1134.
119. Malassiotis S., Strintzis M.G., 2005. Robust real-time 3D head pose estimation from range data. *Pattern Recognition,* 38(8):1153-1165.
120. McKenna S.J., Gong S., 1998. Real-time face pose estimation. *Real-Time Imaging,* 4(5):333347.
121. Brolly X.L.C., Stratelos C., Mulligan J.B. 2003. Model-based head pose estimation for air-traffic controllers. *In: Proceedings of the IEEE International Conference on Image Processing, Barcelona, Catalonia, Spain,* 2:113-116.

122. Malciu M., Prieteux F.J., 2000. A robust model-based approach for 3D head tracking in video sequences. *In: Proceedings of the IEEE International Workshop on Analysis and Modelling of Faces and Gestures, Grenoble, France,* 169-175.

123. Zhang Y., Kambhamettu C., 2000. Robust 3D head tracking under partial occlusion. *In: Proceedings of the 4th IEEE International Workshop on Analysis and Modelling of Faces and Gestures, Grenoble, France,* 176-182.

124. Fitzpatrick P., 2000. Head pose estimation without manual initialization. *Technical report, AI Lab, MIT, Cambridge, USA.*

125. Gorodnichy D.O., 2002. On importance of nose for face tracking. *In: Proceedings of the 5th IEEE International Conference on Automatic Face and Gesture Recognition Washington, DC, USA,* 188-196.

126. Gong S., Mckenna S., Collins J., 1996. An Investigation into face pose distribution. *In: Proceedings of the 2nd IEEE International Conference on Automatic Face and Gesture Recognition, Killington, Vermont, USA,* 265-270.

127. Murase H., Nayar S.K., 1995. Illumination planning for objects recognition using parametric eigenfaces. *The IEEE Transactions on Pattern Analysis and Machine Intelligence,* 16(12):1219-1227.

128. Pentland A., Moghaddam B., Starner T., Oliyide O., Turk M., 1993. View-based and modular eigenspaces for face recognition. *Technical Report 245, M.I.T Media Lab.*

129. Li Y., Gong S., Liddell H., 2000. Support vector regression and classification based multi-view face detection and recognition. *In: Proceedings of the 4th IEEE International Conference on Face and Gesture Recognition, Grenoble, France,* 300-305.

130. Krueger M.W., 1991. Artificial reality II, *Addison-Wesley*.

131. Yin X., Xie M., 2007. Finger identification and hand posture recognition for human–robot interaction. *Image and Vision Computing 25:1291-1300*.

132. Derpanis K.G., 2004. A review of vision-based hand gesture (Unpublished).

133. Wilson A.D., Bobick A.F., Cassell J., 1996. Recovering the temporal structure of natural gesture. *In: Proceedings of the 2nd IEEE International Conference on Automatic Face and Gesture Recognition, Killington, Vermont, USA,* 66-71.

134. O'Hagan R.G., Zalensky A., Rougeaux S., 2002. Visual gesture interfaces for virtual environments. *Interacting with Computers*, 14:231-250.

135. McAllister G., McKenna S.J., Ricketts I.W., 2002. Hand tracking for behaviour understanding. *Image and Vision Computing,* 20:827-840.

136. Ahmad T., Taylor C.J., Lanitis A., Cootes T.F., 1997. Tracking and recognising hand gestures using statistical shape models. *Image and Vision Computing*, 15(5):345-352.

137. Garg P., Aggarwal N., Sofat S., 2009. Vision-based hand gesture recognition. *World Academy of Science, Engineering and Technology*, 49:972-977.

138. Francke H., Ruiz-del-Solar J., Verschae R., 2007. Real-time hand gesture detection and recognition using boosted classifiers and active learning. *In: Proceedings of the 2nd Pacific Rim Conference on Advances in image and video technology, Santiago, Chile*, 553-547.

139. Xiong Y., Quek F., 2006. Hand motion gesture frequency properties and multimodal discourse analysis. *International Journal of Computer Vision*, 69(3):353-371.

140. Wu S., Hong L., 2005. Hand tracking in a natural conversational environment by the interacting multiple model and probabilistic data association (IMM-PDA) algorithm. *Pattern Recognition*, 38:2143-2158.

141. Chen F.S., Fu C.M., Huang C.L., 2003. Hand gesture recognition using a real-time tracking method and Hidden Markov Models. *Image and Vision Computing*, 21:745-758.

142. Coogan T., Awad G., Han J., Sutherland A., 2006. Real-time hand gesture recognition including hand segmentation and tracking. *Advances in Visual Computing*, 495-504.

143. Erol A., Bebis G., Nicolescu M., Boyle R.D., Twombly X., 2007. Vision-based hand pose estimation: A review. *Computer Vision and Image Understanding*, 108:52-73.

144. Ionescu B., Coquin D., Lambert P., Buzuloiu V., 2005. Dynamic hand gesture recognition using the skeleton of the hand. *Journal on Applied Signal Processing (EURASIP)*, 13:2101-2109.

145. Sturman D.J., Zeltzer D., 1994. A survey of glove-based input, *IEEE Computer Graphics and Applications*, 14:30-39.

146. Takahashi T., Kishino F., 1992. A hand gesture recognition method and its application, *Systems and Computers in Japan*, 23(3):38-48.

147. Bobick A.F., Wilson A.D., 1995. A state-based technique for the summarization and recognition of gesture. *In: Proceedings fifth international conference on computer vision*, 382-388.

148. Shamaie A., Sutherland A., 2005. Hand tracking in bimanual movements. *Image and Vision Computing*, 23:1131-1149.

149. Iannizzotto G., Villari M., Vita L., 2001. Hand tracking for human computer interaction with gray level visual glove: turning back to the simple way. *In: Proceedings of the workshop on Perceptive user interfaces, Orlando, Florida, USA.*

150. Pavlovic V.I., Sharma R., Huang T.S., 1997. Visual interpretation of hand gestures for human-computer interaction: a review. *IEEE Transactions on Pattern Analysis and Machine intelligence,* 19(7): 677-695.

151. Baudel T., Beaudouin-Lafom M., 1993. Charade: remote control of objects using gestures. *In: Proceedings of the 1st Conference on Computer Science* 1993*: Indianapolis, Indiana, USA,* 28-35.

152. Cipolla R., Okamoto Y., Kuno Y., 1993. Robust structure from motion using motion Parallax. *In: Proceedings of the IEEE International Conference on Computer Vision,* Berlin, Germany, 374-382.

153. Davis J., Shah M., 1994. Recognising hand gestures. *In: Proceedings of the 3rd European Conference on Computer Vision, Stockholm, Sweden,* 331-340.

154. Davis J., Shah M., 1994. Visual gesture recognition. *In: IEE Proceedings - Vision Image Signal Processing,* 141(2):101-106.

155. Darrell T., Pentland A., 1992. Recognition of space-time gestures using a distributed representation. *Technical Report, No.197, MIT Media Laboratory Perceptual Computing Group*.

156. Cui Y., Weng J.J., 1996. Hand sign recognition from intensity image sequences with complex backgrounds. *In: Proceedings of the 2nd IEEE International Conference on Automatic Face and Gesture Recognition, Killington, Vermont, USA,* 259-264.

157. Ohknishi A., Nishikawa A., 1997. Curvature-based segmentation and recognition of hand gestures. *In: Proceedings of the Annual Conference on Robotics Society of Japan,* 401-407.

158. Nagaya S., Seki S., Oka R., 1996. A theoretical consideration of pattern space trajectory for gesture spotting recognition. *In: Proceedings IEEE 2nd International Workshop on Automatic Face and Gesture Recognition, Killington, Vermont, USA,* 72-77.

159. Heap T., Hogg D., 1996. Towards 3D hand tracking using a deformable model. *In: Proceedings of the 2nd IEEE International Conference on Automatic Face and Gesture Recognition, Killington, Vermont, USA,* 140-145.

160. Zhu S.C., Yuille A.L., 1995. FORMS: A flexible object recognition and modelling system. *In: Proceedings of the 5th International Conference on Computer Vision,* 465-472.

161. Rehg J.M., Kanade T., 1994. Visual tracking of high DOF articulated structures: an application to human hand tracking. *In: Proceedings of the 3rd European Conference on Computer Vision, Stockholm, Sweden,* 2:35-45.

162. Blake A., Curwen R., Zisserman A., 1993. A framework for spatiotemporal control in the tracking of visual contours, *International Journal Computer Vision,* 11:127-145.

163. Kadous M.W., 1995. Machine recognition of Auslan Signs using power gloves: towards large lexicon recognition of Sign Language. *Thesis for Bachelor in Computer Science, University of new South Wales.*

164. Malima A., Özgür E., Çetin M., 2006. A fast algorithm for vision-based hand gesture recognition for robot control. *In: Proceeding of the IEEE 14th Conference on Signal Processing and Communications Applications, Antalya, Turkey,* 1-4.

165. Liang R.H., Ouhyoung M., 1998. A real-time continuous gesture recognition system for sign language. *In: Proceedings IEEE 3rd International Conference on Automatic Face and Gesture Recognition, Nara, Japan,* 558-565.

166. Starner T., Pentland A., 1995. Visual recognition of American Sign Language using Hidden Markov Models. *In: Proceedings of International Workshop on Automatic Face- and Gesture-Recognition, Zurich, Switzerland,* 189-194.

167. Campbell L.W., Becker D.A., Azarbayejani A., Bobick A.F., Pentland A., 1996. Invariant features for 3D gesture recognition. *In: Proceedings IEEE 2nd International Workshop on Automatic Face and Gesture Recognition, Killington, Vermont, USA,* 157-162.

168. Starner T., Weaver J., Pentland A., 1998. Real-time American Sign Language recognition using desk and wearable computer based video. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 20(12):1371-1375.

169. Murakami K., Taguchi H., 1991. Gesture recognition using recurrent neural networks. *In: Proceeding of the SIGCHI conference on Human factors in computing systems: Reaching through technology,* 2008(1-3)1915-1922.

170. Huang C.L., Huang W.Y., 1998. Sign language recognition using model-based tracking and a 3D hopfield neural network. *Machine Vision and Applications* 10:292-307.

171. Lockton R., Fitzgibbon A.W., 2002. Real-time gesture recognition using deterministic boosting. *In: Proceedings of British Machine Vision Conference, Cardiff, UK.*

172. Lee H.K., Kim J.H., 1999. An HMM-based threshold model approach for gesture recognition. *IEEE Transactions on Pattern Analysis and Machine intelligence,* 21(10):961-973.

173. Lien C.C., Huang C.L., 1998. Model-based articulated hand motion tracking for gesture recognition. *Image and Vision Computing* 16:121-134.

174. Elmezain M., Al-Hamadi A., Niese R., Michaelis B., 2010. A robust method for hand tracking using mean-shift algorithm and Kalman filter in stereo colour image sequences. *World Academy of Science, Engineering and Technology, WASET,* 3:131-135.

175. Sánchez-Nielsen E., Antón-Canalís L., Hernández-Tejera M., 2004. Hand gesture recognition for human-machine interaction. *In: Proceedings of the 12th International Conference in Central Europe on Computer Graphics, Visualization and Computer (WSCG), Plzen-Bory, Czech Republic,* 395-402.

176. Malik S., 2003. Real-time hand tracking and finger tracking for interaction. *CSC2503F Project Report*.

177. Huang D.Y., Hu W.C., Chang S.H., 2009. Vision-based hand gesture recognition using PCA + Gabor filters and SVM. *In: Proceeding of the 5th International conference on Intelligent Information Hiding and Multimedia Signal processing (IIH-MSP'09), Kyoto, Japan,* 1-4.

178. Cutler R., Turk M., 1998. View-based interpretation of real-time optical flow for gesture recognition. *In: Proceedings of the 3rd IEEE International Conference on Automatic Face and Gesture Recognition, Nara, Japan,* 416-421.

179. Freeman W.T., Roth M., 1994. Orientation histograms for hand gesture recognition. *In: Proceedings of the International Workshop on automatic face-and gesture-recognition, Bichsel, M., editor, Zurich, Switzerland,* 12:296-301.

180. Yang M.H., Kriegman D.J., Ahuja N., 2002. Detecting faces in images: a survey. *IEEE Transactions on pattern and Machine Intelligence*, 24(1):34-58.

181. Sandeep K., Rajagopalan A.N., 2002. Human face detection in cluttered colour images using skin colour and edge information. *In: Proceedings of the 3rd Indian Conference on Computer Vision, Graphics and Image Processing, Ahmadabad, India*.

182. Zhanjie W., Li T., 2008. A face detection system based skin colour and neural network. *In: Proceedings of the International Conference on Computer Science and Software Engineering, Wuhan Hubei, China*, 961-964.

183. Jones M.J., Rehg J.M., 1999. Statistical colour models with application to skin detection. *In: IEEE Computer Society Conference on Computer vision and Pattern Recognition, Fort Collins, CO, USA*, 274–280.

184. Zarit B.D., Super B.J., Quek F.K.H., 1999. Comparison of five colour models in skin pixel classification. *In: Proceedings of the International Workshop on Recognition, Analysis and Tracking of Faces and Gestures in Real-Time systems, Corfu, Greece,* 58-63.

185. Chai D., Ngan K.N., 1999. Face segmentation using skin-colour map in videophone applications. *IEEE Transactions on Circuits and Systems for Video Technology*, 9(4)551-564.

186. John C.R., 2002. The Image Processing Handbook (*4th Ed*). *CRC Press LLC*.

187. Gonzalez R., Woods R., 1992. Digital Image Processing. *Addison-Wesley Publishing Company Chap. 2*.

188. Luhandjula, K.T., van Wyk, B.J., Kith, K., van Wyk, M.A., 2006. Eye detection for fatigue assessment. *In: Proceedings of the 7th International Symposium of the Pattern Recognition Society of South Africa*. Parys, South Africa.

189. Luhandjula K.T., Monacelli E., Hamam Y., van Wyk B.J., Williams Q., 2009. Visual intention detection for wheelchair motion. *In: proceedings of the 5th International Symposium on Visual Computing, Las Vegas, USA,* 407-416.

190. Viola P., Jones M., 2001. Rapid object detection using a boosted cascade of simple features. *In: Proceedings of the IEEE Computer Society Conference on computer Vision and Pattern Recognition (CVPR'01),* 1:511-518.

191. Peng K., Chen L., Ruan S., Kukharev G., 2005. A robust and efficient algorithm for eye detection on grey intensity faces, *In: Proceedings of the International Workshop on Pattern Recognition for Crime Prevention, Security and Surveillance, Bath, UK,* 302-308.

192. Bradski G., Kaehler A. Pisarevsky V., 2005. Learning-based computer vision with Intel's open source computer vision library. *Intel Technology Journal,* 9(2):119-130.

193. Bradski G., 1998. Real-time face and object tracking as a component of a perceptual user interface. *In: Proceedings of the 4th IEEE Workshop on Applications of Computer Vision, Princeton, NJ, USA,* 214-219.

194. Bradski G., 1998. Computer vision faces tracking for use in a perceptual user interface. *Intel Technology Journal.*

195. Ikizler N., Duygulu P., 2009. Histogram of oriented rectangles: A new pose descriptor for human action recognition. *Image and Vision Computing,* 2:1515-1526.

196. Bishop C., 1995. Neural networks for pattern recognition. *Oxford University Press: New York.*

197. Cristianini N., Shawe-Taylor J., 2000. An introduction to support vector machines and other kernel-based learning methods. *Cambridge University Press: New York.*

198. Webb A.R., 2002. Statistical Pattern Recognition (*2nd Ed*). *Wiley*.

199. MacQueen J.B., 1967. Some Methods for classification and Analysis of Multivariate Observations. *In: 5th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, USA,* 281-297.

200. A tutorial on clustering algorithms**:** *k*-means clustering [online]. Available from: http://home.dei.polimi.it/matteucc/Clustering/tutorial_html/kmeans.html, [Accessed: 14-11-2011].

201. Ikizler N., Duygulu P., 2007. Human action recognition using distribution of oriented rectangular patches. *In: Proceedings of the Workshop on Human Motion-Understanding, Modelling, Capture and Animation*, 271-284.

202. Dalal N., Triggs B., 2005. Histograms of oriented gradients for human detection. *In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* I:886–I:893.

203. Micheli-Tzanakou E. 2000. Supervised and Unsupervised Pattern Recognition: Feature Extraction and Computational Intelligence. Industrial electronics series, *Rutgers University Chap.1*.

204. Kanno, T., Nakata, K., Furuta, K. 2003. Method for team intention inference. *Human-Computer Studies*, 58:393-413.