A Fuzzy Probabilistic Inference Methodology for Constrained 3D Human Motion Classification

Mehdi Khoury

Department of Creative Technologies

University of Portsmouth

A thesis submitted for the degree of

Doctor of Philosophy

06/10/2010

Whilst registered as a candidate for the above degree, I have not been registered for any other research award. The results and conclusions embodied in this thesis are the work of the named candidate and have not been submitted for any other academic award.

To my father who taught me about persistence. To my mother for her understanding and endless love. And a very special thank you to Carol.

Acknowledgements

The author is deeply grateful to Dr Honghai Liu for his invaluable guidance, assistance and support through the years. He played a very important role in leading this work towards scientific maturity. Many thanks to Dr David J. Brown, who has enabled attendance to international conferences and has offered precious advice. This research also benefited greatly from colleagues willing to share their knowledge and expertise: Mr. Zhaojie Ju, Mrs. Xiaofei Ji, Mr. Geoffrey Samuel, Mr. Edward Smart, Dr. Ian Morgan, Dr Chee Seng Chan, Mr Ping Liu. The experimental setup would not have been the same without the help from the Portsmouth University Boxing Club and from the Motion capture Team: Alex Counsell, Geoffrey Samuel, Ollie Seymour, Ian Sedgebeer, David McNab, David Shipway and Maxim Mitrofanov. This project has been funded by EPSRC Industrial CASE studentship and MM2G Ltd under grant No 07002034.

Abstract

Enormous uncertainties in unconstrained human motions lead to a fundamental challenge that many recognising algorithms have to face in practice: efficient and correct motion recognition is a demanding task, especially when human kinematic motions are subject to variations of execution in the spatial and temporal domains, heavily overlap with each other, and are occluded. Due to the lack of a good solution to these problems, many existing methods tend to be either effective but computationally intensive or efficient but vulnerable to misclassification.

This thesis presents a novel inference engine for recognising occluded 3D human motion assisted by the recognition context. First, uncertainties are wrapped into a fuzzy membership function via a novel method called Fuzzy Quantile Generation which employs metrics derived from the probabilistic quantile function. Then, time-dependent and context-aware rules are produced via a genetic programming to smooth the qualitative outputs represented by fuzzy membership functions. Finally, occlusion in motion recognition is taken care of by introducing new procedures for feature selection and feature reconstruction.

Experimental results demonstrate the effectiveness of the proposed framework on motion capture data from real boxers in terms of fuzzy membership generation, context-aware rule generation, and motion occlusion. Future work might involve the extension of Fuzzy Quantile Generation in order to automate the choice of a probability distribution, the enhancement of temporal pattern recognition with probabilistic paradigms, the optimisation of the occlusion module, and the adaptation of the present framework to different application domains.

Contents

Glossary

API Application Programming Interface is an abstraction that provides functions used by components of a software system. 45

CRF Conditional Random Fields produce discriminative models based on the conditional probability of a label given a specific observation. 22

DTW Dynamic Time Warping measures similarity between sequences independently of variations in the speed of performance and changes such as accelerations or decelerations. 28

FMF Fuzzy Membership Function from fuzzy set theory is a function that maps a crisp number such as a joint rotation to a membership score to, for example, a known motion. 3

FQG Fuzzy Quantile Generation is a novel way to generate Fuzzy Membership Functions using metrics derived from the probabilistic quantile function. 3

FRFS Fuzzy Rough Feature Selection is a feature selection method based on fuzzy and rough set theories. 3

FSM Finite State Machine models gestures as ordered sequences of states. 23

GP Genetic Programming is a branch of evolutionary computation that evolves programs represented as tree structures. 3

HMM Hidden Markov Models is a Markov process where states are not directly visible but outputs dependent on these states are observable. 22

IB1 is the simplest instance-based learning algorithm (K-Nearest Neighbour scenario for $k = 1$). 74

J48 is a Waikato Environment for Knowledge Analysis implementation of the C4.5 top-down decision-tree classifier. 76

PySTEP Python Strongly Typed gEnetic Programming is a light Genetic Programming API using the Python programming language. 45

RBM Restricted Boltzmann Machines are stochastic multi-layer neural networks where layers are learned greedily and stacked to create a hierarchy of features in an undirected graph. 26

RVM Relevance Vector Machines has a kernel based functional form similar to Support Vector Machines, but uses Bayesian inference to provide probabilistic classification. 25

SMC Sequential Monte Carlo methods include particle filtering techniques. 24

SMO Sequential Minimal Optimization algorithm used to train the Support Vector Machines classifier in the comparative study. 74

SVM Support Vector Machines is a kernel based method that maps examples of different categories to "points" or hyperplanes in a high dimensional space. 25

TDNN Time Delay Neural Network receives input over several time steps and is conceived to work with continuous data. 26

TSK Takagi-Sugeno-Kang fuzzy inference model is composed by fuzzy generator, fuzzy rule base, and fuzzy inference engine and defuzzification. It outputs membership functions that are either linear or constant. 5

VFI Voting Feature Interval is a classifier that use a voting strategy to build intervals for attributes inside classes. 29

WEKA Waikato Environment for Knowledge Analysis is an open-source collection of machine learning algorithms implemented in the JAVA programming language. 74

List of Figures

List of Tables

Chapter 1

Introduction

Human behaviour understanding can be defined as the recognition and description of actions and activities from the observation of human motions. It is usually performed by comparing observations to models inferred from examples. This process requires learning algorithms that can perform under enormous uncertainties from complex human kinematic structures, occlusion and environmental factors. Research in motion recognition first appeared in the mid nineties as a secondary topic mainly linked to computer vision based experiments. The relatively recent advent of 3d motion capture technology and subsequent spread of view invariant representation models contributed to its emergence as a field of research of its own more concerned with the qualitative analysis of motion itself, than with the extraction of a human pose estimation from videos or images. The rise of gesture based human interface devices in the entertainment industry, the need for automated detection of abnormal behaviour in security surveillance and health care, and the existence of a growing market for computer assisted sport performance analysis are some of the catalysts that accelerate the present growth of the field. However, due to the relative novelty of 3d motion capture technology, the creation of learning data sets for different types of motions is still computationally expensive, labour intensive and costly. This situation creates a niche for algorithms that can classify human motion in constrained scenarios and deal with challenges such as learning samples of sub-optimal size, high dimensionality, noise, imprecision, and incomplete data. The present work has been developed to address these issues in theoretical terms while taking into account the substantial need to provide a feasible solution for related practical applications.

1.1 Problem formulation

A compact definition by Weinland (2008) presents an action as "a 4D event performed by an agent in space and time" where space dimensions are the x,y and z axes. When chained into sequences of body motions, these actions can form the building blocks used to identify more complex behaviours. The recognition of 3D motions is a challenging task which requires the following problems to be addressed accurately.

1. A specific action has to be recognised independently from differences in execution in the spatial and temporal domains. The noise and imprecision in the space domain call for an adapted representation system that can deal with overlapping classes. The fact that an action can be performed at varying speeds also adds the need to model motions as series of time invariant discrete events that are chronologically related.

2. Motions should be recognised using learning samples of sub-optimal size. The representation system must be able to cope with the scarcity of input data that could result from 3D motion capture based experiments.

3. Prior knowledge is required to assist learning in a context-aware and time-sensitive fashion. By taking into account previous motions and their time duration, the aim is to smooth the qualitative output of the classifier by generating and applying context-aware fuzzy-rules.

4. Partial occlusion is one of the bottlenecks for human motion recognition nowadays since it corrupts the input data. This means the system must be able to classify motions from not just scarce, but also insufficient data.

In the context of this thesis, a novel method producing a standalone classifier has been developed to deal with the first two problems. A second contribution has also been added in order to address the third problem. Finally, a specific technique has been devised to specifically answer to the fourth problem.

1.2 Contributions

The contributions to the problem areas described in Section 1.1 are combined into a motion classification framework. Notwithstanding the fact that the design and the implementation of such a framework contains an element of novelty, these individual contributions take the shape of three distinct techniques that can be described as follows:

1. **A standalone classifier using Fuzzy Quantile Generation.**

   Fuzzy Quantile Generation (FQG) is a novel way to generate Fuzzy Membership Functions (FMF) using metrics derived from the probabilistic quantile function. This allows a Fuzzy Membership Function to directly map to the estimated probability distribution behind a data sample. This method demonstrates its effectiveness on the classification of noisy, imprecise and complex motions while using learning samples of sub-optimal size with motion capture data from real boxers. FQG outperforms other time-invariant classifiers in a comparative study made on the boxing data set.

2. **A context-aware Strongly-Typed Genetic Programming filter.**

   The Strongly-Typed Genetic Programming (GP) engine produces time-dependent and context-aware rules to smooth the qualitative outputs of the classifier as required by the third problem (the need for prior knowledge). Various factors such as speed, previous and next movements, and best ranked membership scores are taken into account to generate a complex and subtle network of conditional statements that would otherwise be difficult to identify in an empirical fashion for a human observer. Experimental results on the boxing motion capture data show that the filter consistently improves the accuracy of the FQG classifier.

3. **An occlusion module based on feature selection and reconstruction.**

   The feature selection mechanism introduces a new fuzzy similarity relation based on Laplace distributions has been developed in the context of Fuzzy Rough Feature Selection (FRFS) to identify important joints in case of occlusion. The feature reconstruction scheme proposes a novel way to reduce the uncertainty

caused by occlusion by building plausible rotational data from hidden joints using Fuzzy Qualitative Euler Angles, a modified version of the Fuzzy Qualitative Trigonometry representation system expounded in Liu & Coghill (2005) and Liu (2008). Results show that the system correctly guesses around half of the initially intractable occluded data in the context of the boxing motion capture data experiment.

The novelty of each of these three contributions can be justified with respect to the relevant literature background as follows.

Regarding FQG, the absence of a way to map probability distributions to fuzzy representation for the automated generation of FMF can be emphasized. While probabilistic methods express degrees of belief in a non-compositional way, fuzzy set theory described by Zadeh in Zadeh (2008) introduces degrees of truth which have the advantage of allowing overlapping classes. Despite their differences, probability theory and fuzzy logic can be seen as complementary as explained by Zadeh (1995) and Dubois & Prade (2001). When facing the fundamental problem of Fuzzy Membership Function generation from data, there seem to be no design methodology that allows the direct mapping from a fuzzy representation to a probability distribution. One notable exception is possibility theory presented in Zadeh (1978) that allows the transformation of probability distributions to possibility distributions. However, possibility theory does not allow compositionality as argued by Dubois & Prade (2001). A practical mapping from a Fuzzy Membership Function to a Normal Distribution is hinted at by Frantti (2001) in the context of mobile network engineering. Unfortunately, this work has its shortcomings as this a system does ignore motions which are over the extrema of the range of the learning sample. One possible way to overcome this problem would be to introduce a function that maps a degree of membership to the probability that values fall within a given cumulative probabilistic distribution. As a consequence, there seems to be a theoretical possible gap that leaves open the advent of a method such as FQG.

The suitability of a time-dependent and context-aware Genetic Programming filter can be discussed by briefly reviewing existing fuzzy inference engines. In order to improve the initial classification, a system that can generate fuzzy-rules that smooth the qualitative output in a context-aware fashion needs to be built. These rules might

combine very specific operators such as logical functions, measurements of speed, and multiple input labels ranked as first, second or third best choices for a motion. No initial prior knowledge about how motion can be smoothed is assumed. As the solution to this very specific and demanding problem takes the shape of nested logical structures of arbitrary complexity, standard approaches like the Takagi-Sugeno-Kang (TSK) or Mamdani models become unsuitable. Mainstream inference methods such as Fuzzy Neural Networks in Gobi & Pedrycz (2007) or standard Evolutionary Algorithms presented by Yu *et al.* (2003), Bastian (2000) and Belarbi *et al.* (2005) might also be difficult to reuse as the high dimensionality of the problem implies a significant increase in the size of the training sample in the case of Neural Networks, while the standard evolutionary approaches would struggle to generate syntactically correct rules. There is therefore a possible niche for a specifically adapted variation of Genetic Programming. For the purpose of this research, a Strongly-Typed Genetic Programming open-source distribution was built (see Khoury (2009)).

Regarding the occlusion module, the usefulness of a new fuzzy similarity relation can be explained in the context of Fuzzy Rough Feature Selection, while novel aspects of the feature reconstruction method are emphasized in light of previous work linked to Fuzzy Robot Kinematics. Firstly, regarding the feature selection scheme, Fuzzy Rough Feature Selection (FRFS) introduced by Jensen & Shen (2007, 2009) is chosen as an elegant solution that allows real-valued noisy data to be reduced without the need of user supplied information. FRFS estimates the dependency between attributes by measuring the similarity between two objects x and y for a feature a. The Gaussian, Triangular, and Cornelis fuzzy similarity relations presented in Jensen & Shen (2007) are the techniques used in FRFS. This situation supports the introduction of an improved measure called the Laplace fuzzy similarity relation. Secondly, the novelty of the feature reconstruction mechanism can be established by the fact that the majority of the existing studies done on occlusion focus on object tracking in video sequences. These approaches generally use Kalman Filter for tracking markers of interest that can take the form of blobs in Gabriel *et al.* (2003), image features Utsumi & Ohya (1999), or silhouette images Ueda *et al.* (2003) derived from video sequences. The use of 3d based representation can be found in Kakadiaris & Metaxas (2000) where a three-dimensional pose of the subject's upper and lower arms is recovered and computed in order to create video animation sequences, and Utsumi & Ohya (1999) where a small

number of reliable image features is needed to estimate 3d hand postures with a Fourier descriptor. In this study, 3d motion capture data are used to infer plausible 3 dimensional rotations defined by Euler Angle combinations for each occluded joint. The granularity of the search space is increased by using Fuzzy Qualitative Euler Angles, a modified version of the Fuzzy Qualitative Templates representation system exposed in Liu (2008) and Liu *et al.* (2008) that does not focus on end-effectors trajectories and Denavit-Hartenberg kinematics structures as few joint are occluded. Hence, the reconstruction of plausible rotational data from occluded joints based on a modified version of Fuzzy Qualitative Templates presents an novel approach in the context of 3d motion.

1.3 Outline of thesis

The dissertation consists of five chapters. Chapter 2 introduces background information regarding the representation systems and machine learning techniques in use. Chapter 3 presents a motion recognition framework composed of three components: the FQG layer, the context-aware filter, and the occlusion module. In chapter 4, the performance of the framework is evaluated by putting to the test its different components in several experiments. Chapter 5 concludes this study with future work pointed out. The content of the thesis is outlined below.

Chapter 2 details background information regarding existing motion representation systems and presents a survey of the state of the art machine learning techniques used in behaviour understanding. Firstly, existing representation systems used to capture human motion are introduced, and in light of the specific requirements of this study, the subsequent modelling choices and assumptions are explained. Then, a review of the state-of-the-art of machine learning techniques in use in this area of research is conducted in order to identify potential shortcomings and locate a functional niche that can be targeted by the presented framework.

Chapter 3 describes the framework by introducing the Fuzzy Quantile Generation layer, the context-aware filter, and the occlusion module. Firstly, a detailed and formal description of the Fuzzy Quantile Generation modelling process is first presented, as

well as an analysis of FQG flexibility as a machine learning technique. Secondly, the Genetic Programming filter that smoothes the qualitative output using context-aware fuzzy-rules is detailed. These rules might combine very specific operators such as logical functions, measurements of speed, and multiple input labels ranked as first, second or third best choices for a motion. Finally, occlusion being a de-facto standard problem when dealing with the classification of real human motion data, two different procedures are combined in order to deal with occlusion: feature selection and feature reconstruction. The former deals with the optimisation of the feature selection phase via the introduction of an improved measure of similarity. The latter is about the reconstruction of plausible rotational data from occluded joints using a modified version of Fuzzy Robot Kinematics.

Chapter 4 shows experiments and results where the performance of the framework is evaluated by putting to the test its different components. The challenges posed by the nature of this dataset are, among others: biologically "noisy" data, cross-gait differentials from one individual to another, and high dimensionality caused by the complexity of the skeletal representation (57 degrees of freedom for nineteen joints). It is assumed that being successful at the non-trivial exercise of classification of such complex data might give the presented techniques stronger credentials as a contender in the field of motion recognition. The motion recognition framework is therefore put to the test in an experiment involving the classification of real natural 3d motion capture data in the context of boxing. In the first part of this section, the experimental method and setup are described. Secondly, the performance of FQG as a standalone learning paradigm applicable to behaviour recognition is presented. Thirdly, experimental results of the context-aware GP filter are shown. Finally, the feature selection and feature reconstruction aspects of the occlusion module are evaluated.

Chapter 5 concludes the dissertation with an overview of the present framework, leading to suggestions regarding future work. These involve modifying FQG in order to automate and extend the choice of probability distributions, the enhancement of temporal pattern recognition with probabilistic paradigms, the optimisation of the occlusion module, and the adaptation of the present framework to different application domains.

Chapter 2

Background Research

2.1 Introduction

Intensive study has emphasized the emergence of motion recognition in the mid nineties as a secondary topic mainly linked to computer vision experiments as seen in Cedras & Shah (1995); Gavrila (1999); Hu *et al.* (2004a); Ju (1996); Mitra & Acharya (2007); Moeslund *et al.* (2006); Moeslund & Granum (2001); Wang *et al.* (2003). The early availability of video monitoring systems explains the preeminence of research focusing on computer vision-based human motion capture. The subsequent advent around 1994 of affordable commercial magnetic and optical motion capture systems (at the time, 40000 USD in average for a magnetic motion capture system) gave researchers the ability to measure motions in 3-dimensional space with a precision unheard of. The early availability of video monitoring systems and subsequent advent of motion capture systems contributes to explain why the relevant literature has added to its vision based perspective focusing on areas such as initialisation, tracking and pose estimation another point of interest: action recognition (see Figure 2.1). Initialisation describes pre-processing problems such as the estimation of parameters linked to camera calibration and appearance thresholds, the segmentation of the data into capture reference images, the definition a model appropriate to represent a subject, and the setting an initial pose. Tracking implies a way of segmenting the subject from the background and finding correspondences between segments in consecutive frames. Pose estimation uses some higher level knowledge of the domain to further process the output of the system so that it ensures consistency with the existing representation model, i.e.

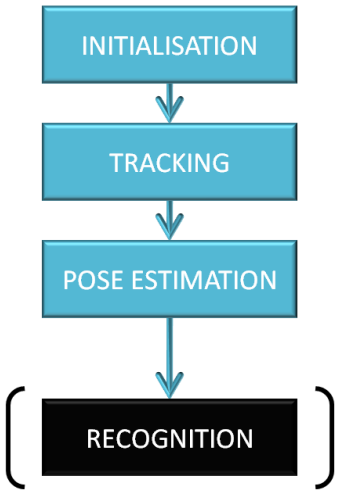


Figure 2.1: The four different areas of research linked to human motion : initialisation, tracking, pose estimation, and - the focus of this thesis - motion recognition.

refine poses based on the constraints of the human model. The motion recognition process is becoming a field of research of its own more concerned with the qualitative classification of motion itself, than with the extraction of human pose estimation from videos or images. Figure 2.2 summarizes the compressive surveys by Moeslund *et al.* (2006); Moeslund & Granum (2001) and presents a chronology from 1980 to 2006 that shows the numbers of papers that focus on initialisation, tracking, pose estimation or motion recognition. In 2006, motion recognition represents 25% of the relevant literature, and presents the second fastest growth after pose estimation when looking at the linear trends. The fast development of motion recognition underlines a shift to a higher level description of actions and interactions based on view-invariant perspective due to the spread of 3d motion capture technology. One present downside of this technology is that the creation of learning data sets for different types of motions is computationally expensive, labour intensive and costly. This calls in a timely fashion for Machine Learning methods that can classify motions from learning samples of reduced size.

This chapter aims at giving a general overview of related work. This is first achieved by reviewing existing formats used to capture human motion, modelling paradigms, and the resulting human skeletal representation chosen for this study. Secondly, a review of the state-of-the-art of the machine learning techniques in use in this area of research is conducted. Thirdly, the novelty of this work is investigated with

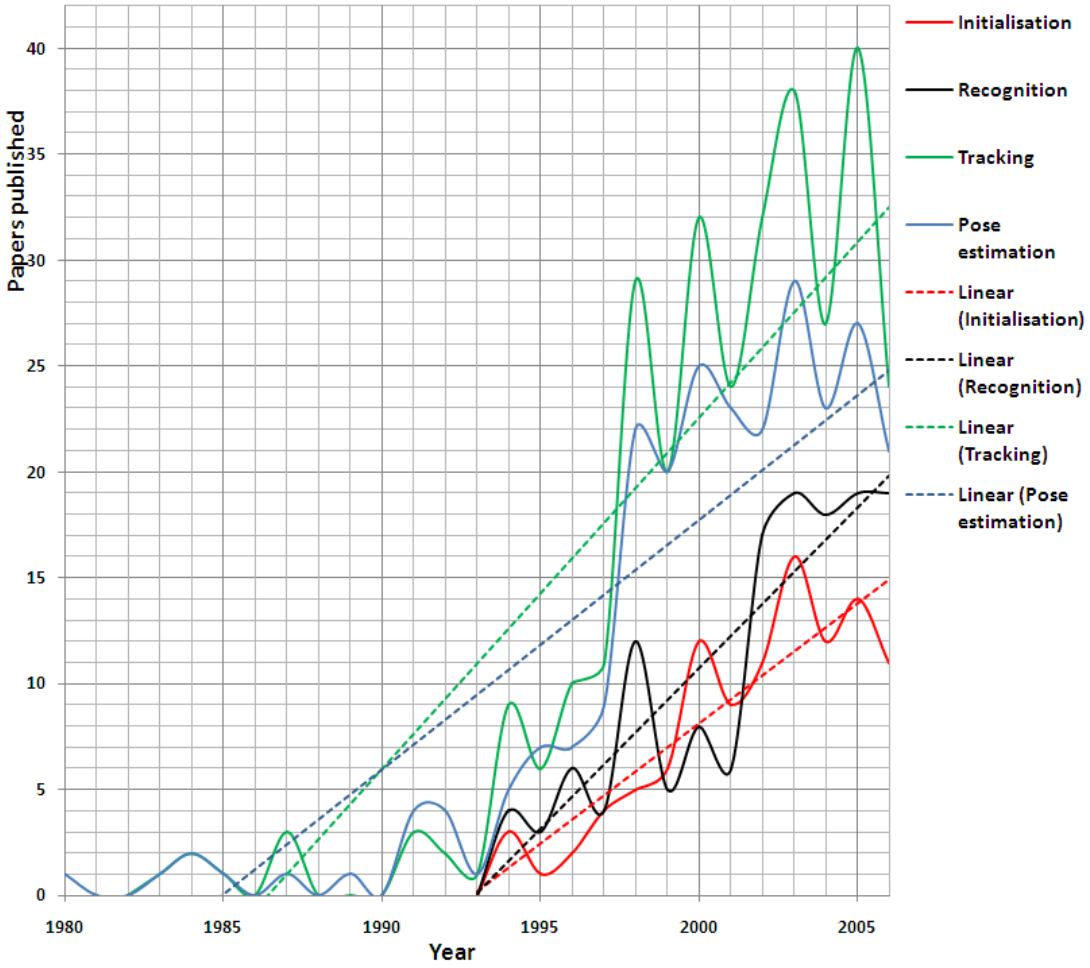


Figure 2.2: Classification of 488 papers published between 1980 and 2006 by relevance to initialisation, tracking, pose estimation and motion recognition . Linear trends are indicated by dashed lines. The graph has been constructed from raw data available in Moeslund *et al.* (2006); Moeslund & Granum (2001).

respect to potential gaps identified in the relevant literature background.

2.2 Human skeletal representation

Motion behaviour understanding is usually performed by comparing observations to models inferred from examples. Before the inference process can take place, raw information has first to be adequately formatted for the purpose of the analysis and then has to be captured in a semantically meaningful way via a model. Existing representation systems in use to model human motion are first introduced. In light of the specific requirements of this experiment, motion capture systems available are reviewed and some of the subsequent modelling choices and assumptions made in this study are explained.

2.2.1 Human motion representations

Motion recognition representation paradigms can be broadly divided into two types: template matching and state-spaces approaches.

2.2.1.1 Template matching

In template matching, observed values of the features that constitute a given motion are converted into a pattern that is compared with other templates stored in a knowledge base. Templates can simply encode static shapes with only spatial information or they can be composite spatio-temporal representations that can describe not only trajectories, but also speed, and acceleration. The most used templates are static poses, optical flow, Motion-History Images, manifolds, mean poses, motion-history volumes (MHV), and scale-space of Spatio-Temporal curves. Static poses templates presented in Freeman *et al.* (1996), Haritaoglu *et al.* (2000) and Jojic *et al.* (2000) focus exclusively on spatial information. Optical flow presented in Polana *et al.* (1994) and Efros *et al.* (2003) is based on the motion-features of points constituting 2D meshes of the subject in sequences of images (see Figure 2.3). Motion-History Images (MHI) Bobick *et al.* (2001) detail images by using pixel intensities as a function of motion recency (see Figure 2.4). Manifolds of recursively filtered images presented by Masoud & Papanikolopoulos (2003), are groups of ordered images similar to MHI. Gonzalez

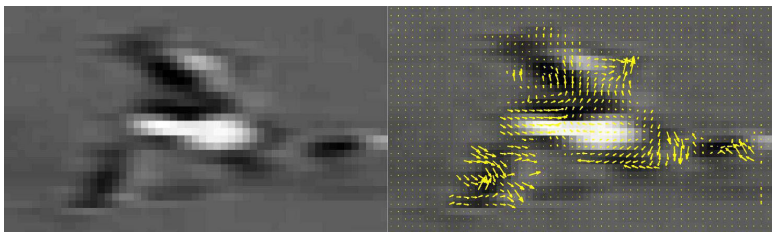


Figure 2.3: Image and corresponding 2D optical flow field of a footballer running in Efros *et al.* (2003)

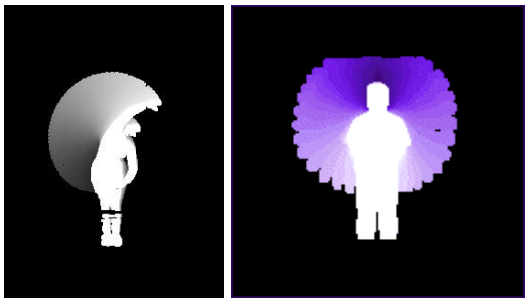


Figure 2.4: Motion-History Images of aerobic motions in Bobick *et al.* (2001)

(2004) expresses manifolds in Principal Component Analysis (PCA) space where key-frames identify the most characteristic poses of an action. Rahman & Robles-kelly (2006) model actions by computing the mean poses from different performances in PCA space and in a normalized time scale. Motion-History Volumes (MHV) presented in Weinland *et al.* (2005) are a view independent 3d version of MHI based on the visual hull of the subject (see Figure 2.5). The scale-space representation is based on Spatio Temporal curves Allmen & Dyer (1990), trajectories Rangarajan *et al.* (1993) or silhouettes Roh *et al.* (2006). It is a kernel-based approach that represent motions as signals in hyperspace and is especially good at recognising cyclic behaviors. Spatio Temporal volumes (STV) that have been proposed by Yilmaz & Shah (2005) contain information of human silhouettes tracked along a normalized time scale and are treated as solid objects for further comparison to other known objects in the database. A similar approach was proposed by Blank *et al.* (2005) (see Figure 2.6). The main advantage of templates is their low computational complexity. However, they are sensitive to the time variability of the performed actions, and therefore are more aimed to classify simple actions.

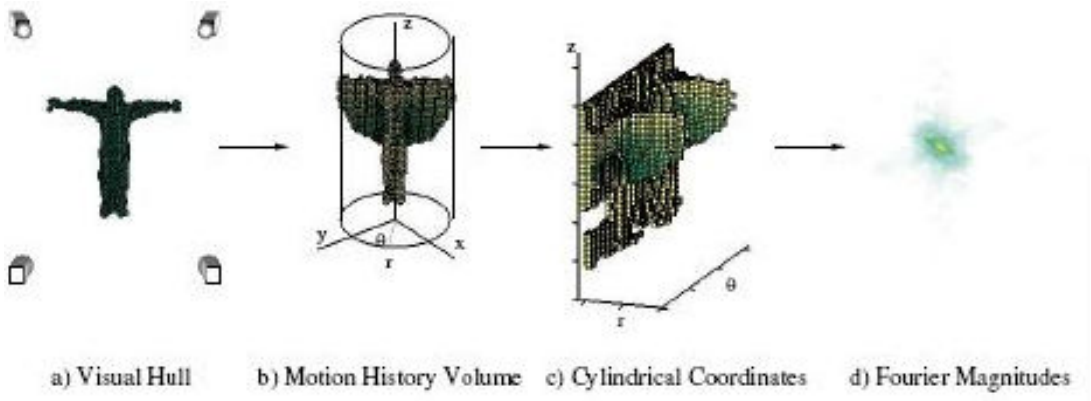


Figure 2.5: View invariant Motion-History Volume representation in Weinland *et al.* (2005)

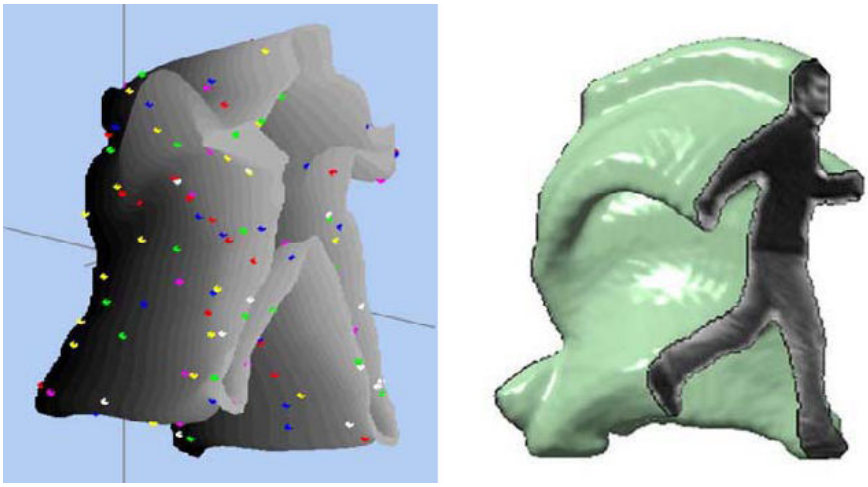


Figure 2.6: Examples of spatio-temporal volumes: *(left)* representation of a tennis move based on work by Yilmaz & Shah (2005) and *(right)* a striding motion by Blank *et al.* (2005)

2.2.1.2 State space models

State-space models transform a motion as a dynamic ordered sequence of discrete states. States are built from representations such as silhouettes, blobs displacements, optical flow and body shape components, XTslices, and composite models where motion features of the subject are completed by contextual information. Human silhouettes are used to build a view point dependent state-space approach in Yamato *et al.* (1992). Translation and rotational speed of blobs representing body parts are used for each state in Bregler (1997). Features of the optical flow are combined with elements of the human body shape in PCA space extracted from multiple views in Ahmad & Lee (2006). XTslices are space-time volumes presented in Ricquebourg & Bouthemy (2000) and Rittscher *et al.* (2002) that represent motions with trajectory patterns. A more complex approach by Ren & Xu (2002) and Ren *et al.* (2004), create states from attributes such as motion-features and contextual information. These attributes are weighted in order to optimise the recognition of specific actions. In order to differentiate subtle motion changes with a view invariant 3D skeletal representation, and take contextual information into account, the space state approach is chosen in conjunction with an adapted raw input that takes the shape of 3D motion capture data. This leads to a review of different motion capture devices.

2.2.2 Motion capture devices

The choice of a holistic (using a human figure as a whole) or non-holistic (focusing on body parts) representation scheme depends on several factors such as the complexity of the actions to be recognised, the level of detail and precision required to correctly identify such motions, and the level of focus between an individual action by a single actor or interactions between multiple subjects. Considering that this study is applied to the classification boxing motions, a non-holistic approach based on 3d motion capture is chosen. 3d motion capture systems can be divided into marker-based and markerless systems.

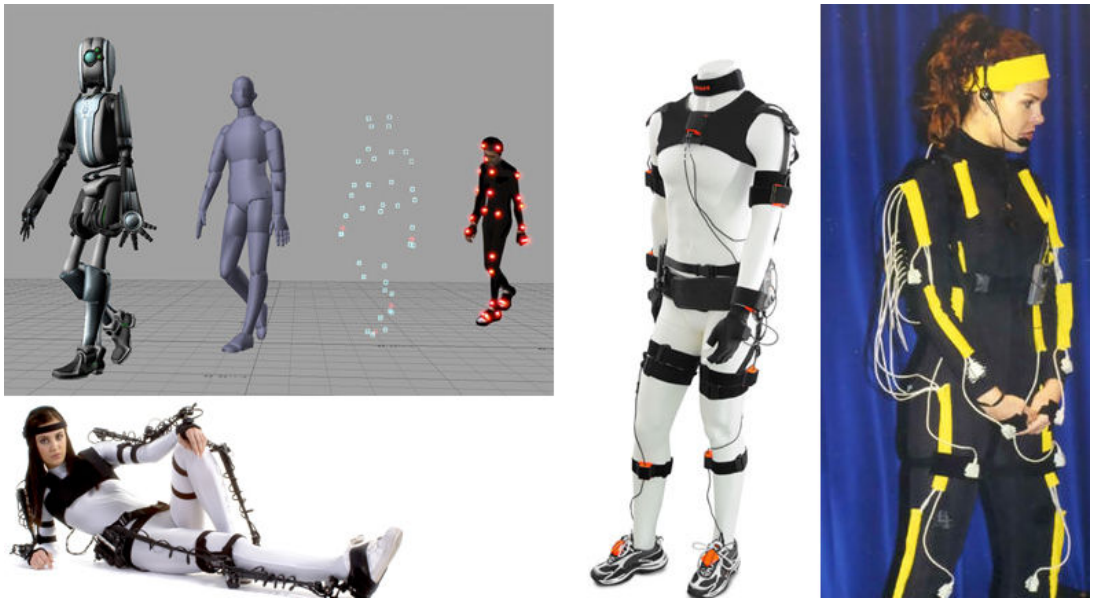


Figure 2.7: Optical (*top left*), mechanical (*bottom left*), inertial (*middle*), and magnetic (*right*) motion capture systems

2.2.2.1 Marker based motion capture

Marker-based motion capture systems are widely used in the industry. Markers placed all around the body in order to track the position of body parts during a motion. We can distinguish mainly among optical, magnetic, mechanical and inertial mocap systems (Figure 2.7).

In optical motion capture, several cameras placed all around the subject track the displacements of reflective or luminous markers. This type of system is able to capture extremely fast motions with the best accuracy. One potential problem might be occlusion during a motion if there are not enough cameras or if there are several subjects. Magnetic systems compute spatial coordinates and orientation from the variations of the magnetic flux between orthogonal coils on both the transmitter and each of the receivers. This system has the advantage of ignoring occlusion, but it is highly sensitive to interferences resulting from metal objects and electrical sources. Mechanical systems take the shape of articulated exoskeletons. Joints rotations can be tracked without occlusion and spatial limits. Inertial systems are able to capture the positions, orientation and velocities of markers in large capture areas, also free-of-occlusion. The recent

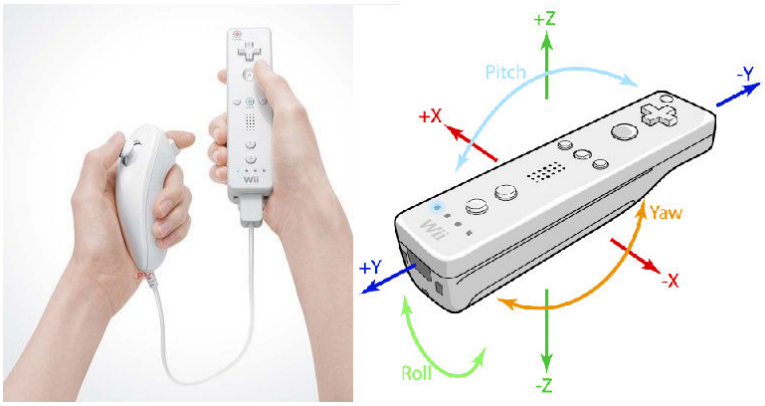


Figure 2.8: Nintendo Wiimote: Nintendo (2006)

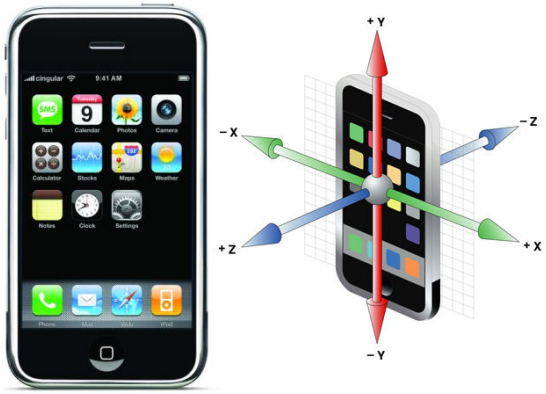


Figure 2.9: iPhone: Apple (2007)

spread of portable devices including accelerometers in the video game industry (see the console remote Wiimote in Figure 2.8), and the mobile market (see the Apple iPhone in Figure 2.9), has made inertial motion sensing affordable in the context of Human Computer Interaction.

2.2.2.2 Markerless motion capture

Recently, there have been remarkable advances in markerless motion capture which, in the near future, may render marker-based systems obsolete for many HCI applications due to their lower cost and easier use. In this approach, human movements are intended to be captured directly from images obtained by cameras where each pixel in the image not only has a standard colour value i.e. RGB, but also has a depth value. Depths of the

Figure 2.10: Mixing stereo vision and time of flight: images and corresponding depth maps in Zhu *et al.* (2008)

Figure 2.11: Kinect gesture based video game recognition system: Microsoft (2010)

observed scene can be achievable with new technologies that have recently emerged, such as the stereo (Figure 2.10 presented in Birchfield & Tomasi (1996) and Time-of-Flight cameras in May *et al.* (2006), or a combination of both as seen in Zhu *et al.* (2008) (a similar approach seems to be taken by Microsoft in the context of the Microsoft Kinect project - see Figure 2.8).

Considering the high accuracy of optical motion capture and its capacity to capture fast movement, this technique seems ideally suited for a boxing data set. A Vicon optical motion capture system was used to capture the raw data (further details are available regarding the equipment and setup used in section 4). The motion capture data format and modelling representation now need to be defined.

2.2.3 Motion capture format and corresponding model

Several choices and assumptions are made. First, considering that, as argued by Favre *et al.* (2006), motion capture data cannot give absolutely exact skeletal displacements of joints due to soft tissues movements, it is considered sufficient to obtain an approximation which would be good enough to characterize a motion. Secondly, the body is simplified to nineteen main joints which is the standard number of joints used to animate human representations. It is assumed that this number is sufficient to characterize and understand the general motions of a human skeleton performing boxing combinations. Finally, each joint having three degrees of freedom, the rotations are represented by Euler ZXY angles characterized by three rotation angles (in order Z, X and Y) given in degrees by the widely spread BVH motion capture format presented in Thingvold (1999) and Meredith & Maddock (2000) in which a human skeleton is formed of skeletal limbs linked by rotational joints (see figure 2.12 and 2.13).

In practice, for every frame, the raw observable data takes the shape of a nineteen-by-three matrix describing ZXY Euler Angles for all nineteen joints in a simplified human skeletal representation. In other words, multiple continuous variables between 0 and 360 characterize a stance at any time. The BVH format uses Euler angles to quantify rotations of joints having three Degrees of Freedom. This system is not perfect (Gimbal Lock is a possible issue), but allows one to gather data easily when using motion capture while keeping track of subcomponents such as the rotations of individual joints. Exposure to Gimbal Lock is reduced as the rotation angles of the skeleton are first computed through Motion Builder using quaternions before being transformed into Euler Angles approximations. Future work might involve keeping track of subcomponents of the rotations of individual joints in order to analyse and suggest corrections to a motion. In this context, the Euler Angles representation system seems acceptable because it is simple and intuitive enough to enable roll-yaw-pitch analysis and facilitates the suggestion of qualitative corrections for a rotation. If quaternions are not subject to Gimbal Lock, they are not as intuitive and lead to a greater multiplicity of interpretation for one single rotation as they are in 4 dimensions. This increases greatly the number of possible qualitative states for one rotation and would make the suggestion of plausible rotations of occluded joints more difficult.

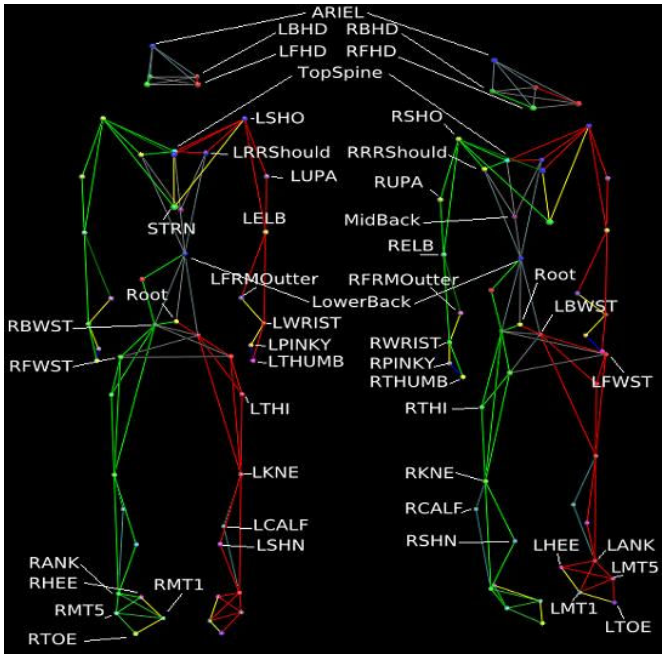


Figure 2.12: Optical markers - spatial data used to compute the BVH representation

Fuzzy membership functions introduced by Zadeh (1986) can deal with imprecision and overlapping classes, which make them ideal for the classification of such data. Therefore, in this study, a motion is represented as state that is composed of a set of 57 Fuzzy membership functions that each express a membership score to an extended range of fixed angular rotations (see Figure 2.14).

2.3 Machine learning techniques used in motion recognition

The machine learning techniques used in motion recognition can be presented as belonging to nine categories as detailed in Figure 2.15: probabilistic graphical models, Finite State Machine, Kalman filter and Sequential Monte Carlo methods, Kernel based methods, connectionist approaches, syntactic techniques, and hybrid approaches including soft computing. All the methods described in blue are pretty distinct and have their own respective merits and shortcomings. This thesis work can be classified under the “hybrid methods” category (coloured in black) which is a more recent area of

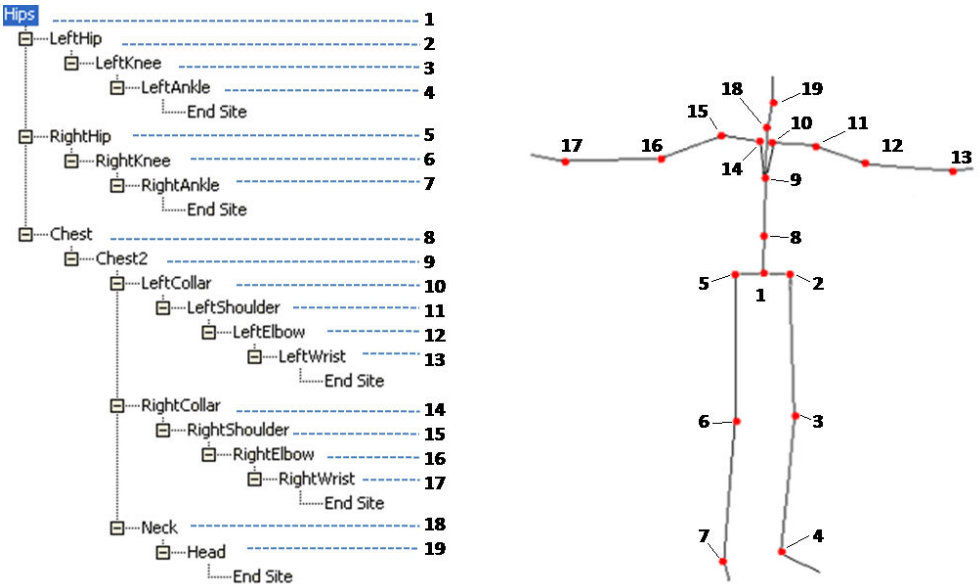


Figure 2.13: BVH Motion Capture format encoding human skeletal representation into rotational data - 19 joints with 3 degrees of freedom each

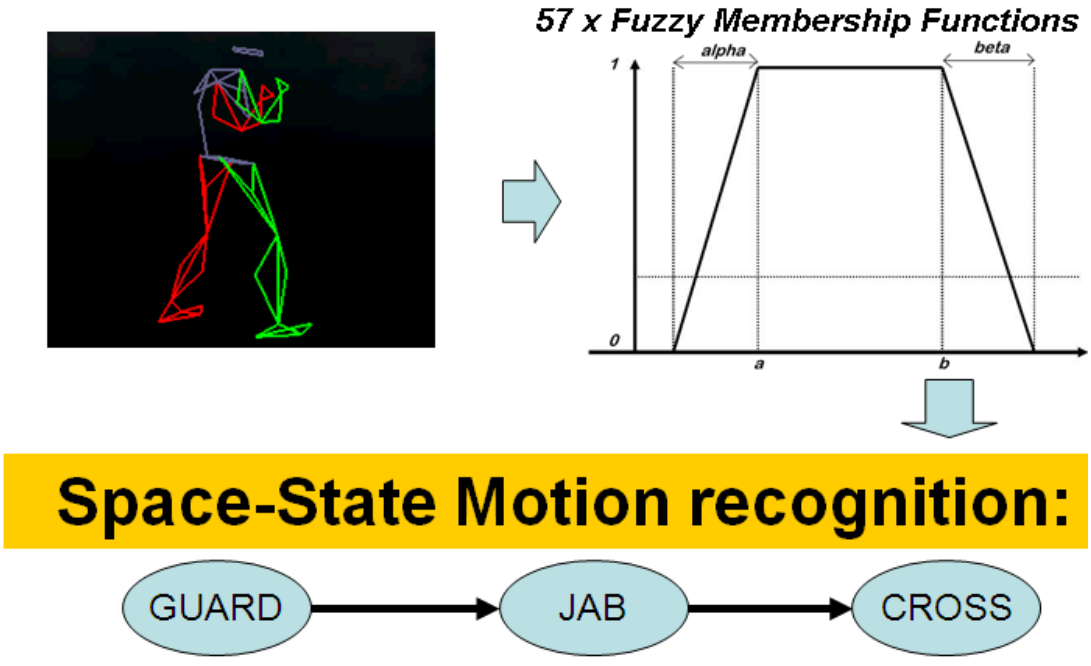


Figure 2.14: An overview of the modelling process: 3D motion capture input data using fuzzy membership functions in order to derive a space state recognition

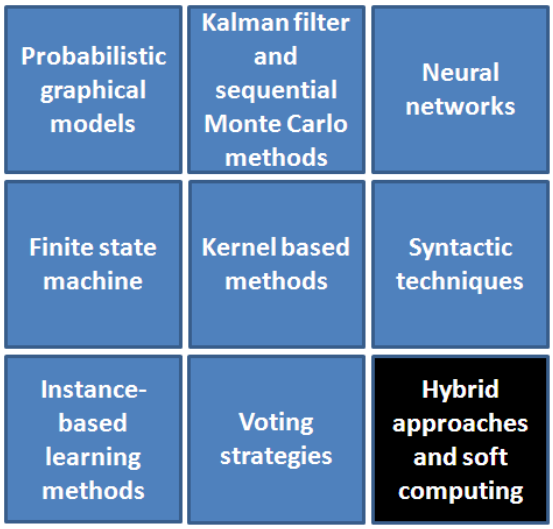


Figure 2.15: Machine learning techniques used in motion recognition. This thesis work is part of "Hybrid approaches" that combine some of the existing techniques shown in blue on the graph.

research in constant expansion. It encompasses a wide range of combinations of established techniques and sometimes various representation paradigms, created in order to minimise known shortcomings. In this section, the different categories of machine learning methods used for motion recognition are individually described and discussed.

2.3.1 Probabilistic graphical models

Probabilistic graphical models are the most widely spread paradigm. These graphs provide a compact representation of joint probability distributions where nodes represent random variables, and arcs represent conditional dependencies. Probabilistic graphical models can be directed (i.e. graphs where directional arcs can capture causality and encode deterministic relationships), or undirected (i.e. a graph where two nodes are conditionally independent if they are not directly connected).

Bayes Net Takahashi *et al.* (1994) uses a Bayesian network, that is to say a directed acyclic graph where nodes represent variables, and edges represent conditional dependencies. Probability functions associate a set of input value from a node's parent variables to the probability of the variable represented by the node. Bayes Net can be used to compute the conditional probability of one node, given values assigned to the

other nodes. This gives the posterior probability distribution of the classification node given the values of other attributes. Bayes Net has been used for recognition of head gestures by Lu *et al.* (2005), general motion understanding from video sequences by Leventon & Freeman (1998), and similarly, has been used by Sidenbladh *et al.* (2002) in conjunction with optical flow. Bayes Net is theoretically well equipped to deal with classification problems in the sense that it maximizes the expected utility of choices and its performance does not drop dramatically when models are slightly modified by small alterations. Bayes Net can also handle incomplete data by taking into account dependencies between variables. On the other hand, it builds networks which depend greatly on the accuracy of a prior model of beliefs. Furthermore, the entire network has to be computed to get the probability of any node, which is a NP-hard (nondeterministic polynomial-time hard) problem. This results in Bayes Net to be computationally expensive.

Hidden Markov Models (HMM) is a double stochastic process that associates an underlying Markov chain with a finite number of states to a set of random functions. As a generative model, it captures the joint probabilities of observations and corresponding states. Being an established temporal classification technique, HMM has found many application in areas like speech, handwriting and gesture recognition. It is presently the most popular method in motion recognition, and has been applied to various problems such as the recognition of human actions from time sequential images of sport scenes Yamato *et al.* (1992), the identification of complex social interactions based on two-handed actions Pentland *et al.* (1996) or from the analysis of relative trajectories Oliver *et al.* (2000), and sign language recognition Starner & Pentland (1997). HMM presents distinct advantages such as a solid statistical foundation, efficiency, flexibility, generality (various knowledge sources can be combined into a single HMM), and a capability for unsupervised learning from variable-length sequences (there is no need to manually set gestures boundaries). However, in order to remain tractable, HMM requires observations to be conditionally independent, which does not scale well in real-world conditions where features are often linked by multiple dependencies and interactions. Furthermore, HMM needs learning sample of quite significant size to work efficiently, which is not ideal for 3D motion capture.

Conditional Random Fields (CRF) are introduced by Lafferty *et al.* (2001) as "models for structured classification". As opposed to generative models like in HMM,

CRF produce discriminative models based on the best conditional probability over a sequence of labels given specific observations. CRF have been recently applied to motion recognition by Sminchisescu *et al.* (2006) and are considered by Vail (2008) to be well suited for activity recognition from sensor data. CRF facilitate the inference of sequences of activities by modelling relationships between labels because, unlike HMM, they do not rely on the assumption of independence of observations. However, one limitation of CRF is that it cannot use hidden-state variables in order to capture intermediate structures.

2.3.2 Finite state machine

Finite State Machine (FSM) is a method that models gestures as ordered sequences of states where each state represents a prototype trajectory modelled as a set of points in space and time.

Parameters determining the state transitions are built during training. The process of recognition consists in feeding input data under the form of feature vectors representing trajectories to the FSM in order to change the current state. A gesture is recognized when a final state is reached. Davis & Shah (1994) used FSM to model displacements of fingertips for hand gesture recognition. Bobick & Wilson (1997) uses dynamic programming to compute the average combined membership to a fuzzy state of the FSM. This approach is successfully tested on various input data such as 2D movements from a mouse input device, hand motions measured by magnetic sensors, and changing eigenvector projection coefficients computed from an image sequence. Other studies by Yeasin & Chaudhuri (2000) and Hong *et al.* (2000) focus on the recognition of symbolic hand gestures from video images.

A variation of FSM called Non-deterministic finite automaton (NFA) has properties such as instantaneousness and pure-nondeterminism and allows to build large FSMs in a piecewise and compact fashion. It is used by Wada & Matsuyama (2000) for the demanding task of multi-object behaviour recognition. The two main shortcomings FSM are their inherently synchronous nature (i.e. only one global state at any time), and the state space explosion that occurs when facing intricate motions with numerous variations distributed over a relatively high number of features.

2.3.3 Kalman filter and sequential Monte Carlo methods

Although they are different paradigms, Kalman filter and Sequential Monte Carlo methods are mostly used when motion recognition involves tracking and pose estimation from noisy sensors data.

Kalman filter introduced by Welch & Bishop (1995) is a method that recursively produces estimations of the true values from time based measurements containing random variations. Estimated values and their uncertainty are computed in order to produce a weighted average that favors values with the least uncertainty. The resulting filtered values have a better estimated uncertainty and tend to be closer to the true values than the original measurements. The basic Kalman filter relies on the linear assumption of a unimodal Gaussian density function to produce estimations. This can become a limitation in the case of complex nonlinear systems.

Sequential Monte Carlo methods (SMC) such as particle filtering techniques use a set of random samples or particles to represent the propagation of arbitrary probability densities over time. Particle filtering can focus in a stochastic way on probable regions of state-space, deal with non-Gaussian noise, and build multiple models (when tracking multiple moving targets). However, this method generally exhibits a high computational complexity (the number of particles needed increase drastically with the dimension of the model) and might result in a loss of diversity in the solution space. Condensation algorithms (Conditional Density Propagation) Isard & Blake (1996) propagate at each time step by representing the distribution of possible interpretations as a set of random samples instead of a unimodal density function (such as the Gaussian density function used in Kalman filter). Due to the fact that it is a mapping to a parallel architecture, condensation can be very effective when dealing with the representation of simultaneous alternative hypotheses. On the other hand, it does require a dynamic network with random communication patterns where decisions about connections are unknown before run time. This can greatly affect the speed of the algorithm. Other particle filters used in the context of motion recognition are described by Arulampalam *et al.* (2002) and Kwok *et al.* (2003).

2.3.4 Kernel based methods

Kernel based methods such as Support Vector Machines (SVM) and Relevance Vector Machines (RVM) are computationally efficient and give good results in high dimensional space.

Support Vector Machine is a kernel based method that maps examples of different categories to "points" or hyperplanes in a high dimensional space. This hyperplane-based representation amplifies the differences between examples that belong to different categories. This gap called functional margin is maximized to be the largest distance to the nearest training data points of any class. The wider and the clearer the gap, the easier the separation that leads to the classification of new examples, and the lower the generalisation error of the classifier. Finding the parameters that define the hyperplane with the maximum margin is a non trivial optimisation problem. The most popular algorithm used to train the SVM is the SMO algorithm that attempts to solve this problem by scaling it down into 2-dimensional sub-units. Further details about SMO can be found in Schölkopf *et al.* (1999) and Keerthi *et al.* (2001). SVM has been used to recognize human actions from video samples by Schldt *et al.* (2004), and in conjunction with optical flow by Danafar & Gheissari (2007). Mori *et al.* (2004) also used SVM to discover remarkable motion features. Kapur *et al.* (2005) used it as a comparison method in motion classification. Shawe-Taylor & Cristianini (2004) underlined some advantages such as: "the absence of local minima, the sparseness of the solution and the capacity control obtained by optimising the margin". However, SVM also presents extensive memory requirements, and a delicate and computationally expensive hyperparameter tuning process.

Relevance Vector Machines (RVM) has a kernel based functional form similar to SVM, but uses Bayesian inference to provide probabilistic classification. In this method, a set of probabilistic weights linked to hyperparameters are iteratively refined using a learning process similar to Expectation Maximization (EM). While RVM performances are similar to SVM, it achieves greater sparcity by using fewer kernel functions. RVM are used by Oikonomopoulos & Pantic (2007) to recognise aerobic exercises from image sequences and Guo & Qian (2006) for motor action recognition. Compared to the SVM, the Bayesian formulation avoids the set of free parameters of the SVM (that usually require cross-validation-based post-optimizations). However,

as RVM use an Expectation Maximization learning method, there are at risk of local minima, unlike the standard SMO-based algorithms employed by SVMs which are guaranteed to find a global optimum.

2.3.5 Connectionist approaches

Different types of neural networks such as Multi-layer Perceptrons, Radial Basis Function networks, Time Delay Neural Network (TDNN), Self-Organizing Neural Networks, and Restricted Boltzmann Machines (RBM) are often used in the context of motion recognition.

The Multi-layer Perceptrons presented in Rumelhart *et al.* (2002) is a feed-forward artificial neural network model that uses at least three layers of nodes with nonlinear activation functions based on Sigmoids. This is one of the most popular types of neural networks, and it has been applied to many situations varying from activity recognition from video surveillance in Jan *et al.* (2003) to hand gesture recognition in Symeonidis (1996). The Radial Basis Function network presented by Bugmann (1998) can be defined as a statistical feed-forward two-layer artificial neural network that uses Gaussian radial basis functions as activation functions in its hidden units. Output units are weighted sums of the hidden unit results. A non-linear input is approximated into a linear output. This gives Radial Basis Networks the ability to model and approximate efficiently complex functions. It has been used for real-time gesture recognition from image sequences in the context of a human-computer interface system by Ng & Ranganath (2000, 2002).

The Time Delay Neural Network (TDNN) receives input over several time steps and is conceived to work with continuous data. Delay units are added to a general static network, and some of the preceding values in a time-varying sequence are used to predict the next value. As larger data sets become available, small groups of adjacent neurons are transformed into single cells in the following neuron layer in order to increase the granularity of the search space in the time domain. This divides time series data into smaller chunks on which the network can be trained. TDNN has been used by Yang & Ahuja (1998, 1999) to recognize gestures related to American Sign Language (ASL). This method was also used for various other applications such as lip-

reading Meier *et al.* (2000) and gestural control of music and sound Modler & Myatt (2008).

Self-Organizing Neural Networks are used for unsupervised learning where unrestricted motions are defined as sequences of flow vectors capturing the positions and velocities of the object in the image plane. In action recognition, Self Organizing Maps and Competitive Networks are two popular approaches. A Kohonen Self-Organising Map (SOM) defines a neural network that transforms the input space into a lower-dimensional output space called a map. This map is obtained through a dimensionality reduction process based on a neighbourhood function to preserves the topological properties of the input space. Owens & Hunter (2000) applies SOM to find patterns in the distribution of movements in order to determine whether a point on a trajectory is normal or abnormal. Competitive networks presented by Johnson & Hogg (1996) build statistical models of object trajectories by interconnecting two networks via a layer of leaky neurons that have a decaying memory of previous activations. Sumpter & Bulpitt (2000) improves this concept by introducing feedback to the second competitive network. Hu *et al.* (2004b) also builds competitive neural network structures with a smaller scale and a faster learning speed.

Restricted Boltzmann Machines (RBMs) are stochastic multi-layer neural networks where layers are learned greedily and stacked to create a hierarchy of features in an undirected graph. Taylor *et al.* (2006) uses them to model human motions such as walking, crouching, sitting and running. The efficiency of neural networks being greatly dependent on the completeness and quality of the training set, this type of method is generally difficult to use with learning samples of sub-optimal size. Furthermore, the model generated by a neural network of average complexity is by nature very difficult to interpret when trying to identify relations of causality.

2.3.6 Syntactic techniques

Syntactic techniques use a grammar-based parser to recognize sequences of discrete behaviours on top of a low level standard independent probabilistic temporal behaviour detector. This grammatical approach has been studied mostly in the context of pattern recognition from static images. Brand (1996) uses a simple non-probabilistic grammar to recognize sequences of discrete behaviours. Ivanov & Bobick (2000) describe

a probabilistic version of the syntactic approach applied to the classification of interactions between multiple agents. The division of the recognition problem between a low-level classifier that outputs temporal features and a stochastic context-free parser presents advantages such as the extension of constraints on a longer time scale, the disambiguation of uncertain low-level detection, and the inclusion of prior knowledge about the structure of temporal behaviours in a given domain. The problem of this approach is that it does not solve the problem of how to build a set of intricate grammar rules that can fit complex data.

2.3.7 Instance-based learning methods

Instance-based learning methods such as Nearest Neighbours classifiers or Dynamic Time Warping in the temporal domain are popular in the field of motion recognition.

Nearest Neighbours based techniques generalise and classify by finding the most similar instance (hence the "nearest neighbour") and labelling the next unknown instance with the same label as the known neighbour. One measures the Normalised Euclidean distance to find the training sample closest to an existing test sample and then integrate the latter into the same class. If several closest training instances are equally distant to the same test sample, the object being assigned to the class most common amongst its k nearest neighbors is classified by a majority vote. Learning is an encapsulation process during which training data are not generalised before the end, that is to say at classification time (hence the "lazy learners" denomination). Nearest Neighbours classifiers have the advantages of being fast and cope well with small learning samples. K-nearest Neighbours were recently used by Kollorz *et al.* (2008) for gesture recognition from time-of-flight cameras and by Kaâniche & Brémond (2009) uses for motion classification from histograms descriptors.

Dynamic Time Warping (DTW) measures similarity between sequences independently of variations in the speed of performance and changes such as accelerations or decelerations. DTW warps sequences using time alignment and normalisation in order to obtain a measure of their similarity which is independent of the non-linear variations of the time dimension. DTW does not rely on the continuity of data sequences. It is therefore particularly useful for matching sequences with missing data. It has been often used in application domains such as human-computer interaction (HCI) systems

in studies by Takahashi *et al.* (1994) and Corradini (2001) and sign language recognition in Kuzmanic & Zanchi (2007). Although instance-based learning methods are computationally very efficient, they generally do suffer from a high sensitivity to noise.

2.3.8 Classifiers based on voting strategies

Classifiers based on voting strategies such as HyperPipes and Voting Feature Interval(VFI) have been used in the context of gesture recognition.

HyperPipes presented in Frank *et al.* (2005) is an algorithm that, for each class, builds bounds for the attribute-values found in the examples belonging to this class. Each hyperpipe contains the attribute-values found in the examples from the class it was built to cover. A test example is classified by finding the hyperpipe that most contains the instance. Hyperpipes has the advantages of speed, simplicity, and can cope well with large numbers of attributes. Eisenstein & Davis (2004) attempted to develop a human gesture classifier based on HyperPipes.

Voting Feature Interval(VFI) presented by Demiröz & Güvenir (1997), builds intervals for each attribute inside each class. Class counts are recorded for each interval on each attribute. The predicted class is the one with the highest count. Falco *et al.* (2008) used VFI as a comparison in a benchmark test done in the context of gesture recognition. Other studies by Kaâniche & Brémond (2010) and Zhang *et al.* (2009) use other variations of such voting mechanisms. These techniques are extremely fast, but their simplicity limits their accuracy when the interdependency between attributes gets stronger.

2.3.9 Hybrid approaches and soft computing

There is a growing trend of hybrid approaches in motion recognition such as, among others, multilayer perceptron and HMM in Bourlard & Wellekens (1990), K-Nearest Neighbors Algorithm and naive Bayes in Ziaie *et al.* (2009), PCA and HMM in Coogan *et al.* (2006), and Hidden Conditional Random Field in Wang *et al.* (2006) that combines the ability of CRFs to use dependent input features and the ability of HMMs to learn latent structure. More and more of the approaches described above are successfully integrating concept of soft computing such as Fuzzy Set Theory. The work described in this thesis fits into this category.

Fuzzy C-means clustering has been used in Korde & Jondhale (2008) and combined with HMM in Zhang & Naghdy (2005), or FSM in Verma & Dev (2009). Fuzzy neural networks are popular and presented in studies such as in Juang & Ku (2005) and Su (2000). Fuzzy decision tree in Fang *et al.* (2004) which is a mixture of "top-down" rule based approach and fuzzy representation is also quite well represented. Combinations of Finite State Automata with interval fuzzy logic in Callejas Bedregal *et al.* (2006) or with fuzzy partitioning and HMM in Kim *et al.* (1996) are not uncommon either. Methods mixing HMM and fuzzy neural networks are presented in Wang & Dai (2007) or techniques merging Possibility distribution and Dynamic Programming in Benoit *et al.* (2003) are some of the many new hybrid methods appearing in the field.

2.4 Discussion

The chronological emergence of motion recognition in the wider context of motion capture systems underlines a shift to a higher level description of actions and interactions based on view-invariant perspective due to the spread of 3d motion capture technology. In this study, an optical motion capture system is used in conjunction with the BVH motion capture format in order to quantify rotations of joints and provide raw input data. A state-space modelling paradigm is then employed to represent a motion as a discrete state that is composed of a set of 57 Fuzzy membership functions that each express a membership score to an extended range of fixed angular rotations. A review of the state-of-the-art of machine learning techniques shows that conventional methods such as probabilistic graphical models and neural networks need learning sample of very significant size to work efficiently, which is not ideal for 3D motion capture. Similarly, instance-based classifiers generally do suffer from a high sensitivity to noise, which is a problem when dealing with the naturally imprecise data set. Methods such as finite state machine, voting strategies and syntactic techniques are difficult to apply to 3d motion capture data due to the complexity and high dimensionality of the problem. The same applies for Kernel-based methods because of the delicate and computationally expensive hyperparameter tuning process. Hybrid approaches which are based on clustering methods do present a specific problem in that they do not overlap well with the sets of moves distinguished by human in the continuous spatio temporal domain. Furthermore, other hybrid techniques based on fuzzy neural networks or

probabilistic graphical models still present significant constraints regarding the size of the training set. Due to the novelty of the relatively recent availability of 3d motion capture data sets, hybrid methods have yet to satisfy the very demanding and specific constraints of this type of data in the context of the recognition of human activity. The demand for such techniques is now great. As noted by Moeslund *et al.* (2006), there is at present a quantitative explosion of various hybrid approaches in the field, triggered by the need to address motion recognition problems at an unprecedented scale in areas such as the entertainment industry, security surveillance and health care. This situation confirms a new growing trend of hybrid Machine Learning methods that seek to fit into a niche satisfying the following functional requirements: the ability to classify from learning samples of sub-optimal size, a low sensitivity to noise, and simplicity regarding the parameter tuning process. The motion classification framework presented in this thesis is aiming a addressing these specific problems and is detailed in the next chapter.

Chapter 3

Motion recognition framework

3.1 Introduction

The presented framework aims at classifying 3D human motions while using learning samples of sub-optimal size and taking into account time, contextual information and occlusion. As illustrated in Figure 3.1, a standalone time invariant classifier based on Fuzzy Quantile Generation is built and connected to a post-processing filter that deals with context sensitive/time variant information and a pre-processing module that deals with occlusion. The standalone classifier at the heart of the system must be able to integrate information from the occlusion module during the classification process, while delivering a discrete output exploitable by the context-aware post-processing filter. The required level of coupling between the different components of the framework is ensured by choosing a type of input/output that can be shared by all three modules, i.e. the fuzzy membership scores of observed data to different known motions. The occlusion module produces estimations of rotations for hidden joints that can be integrated seamlessly in the classification process by correcting these fuzzy membership scores, while the post-processing filter can use them as discrete inputs by ranking them. Each of the detailed modules deals with specific problems of a different nature. The occlusion module must deal with insufficient information, while the FQG classifier has to face challenges linked to the nature of motion capture data: spatial and temporal variations, cross-gait differentials from one individual to another, relatively high dimensionality of the representation, and large number of learning samples of suboptimal size. Similarly, the post-processing filter must be able to generate rules processing

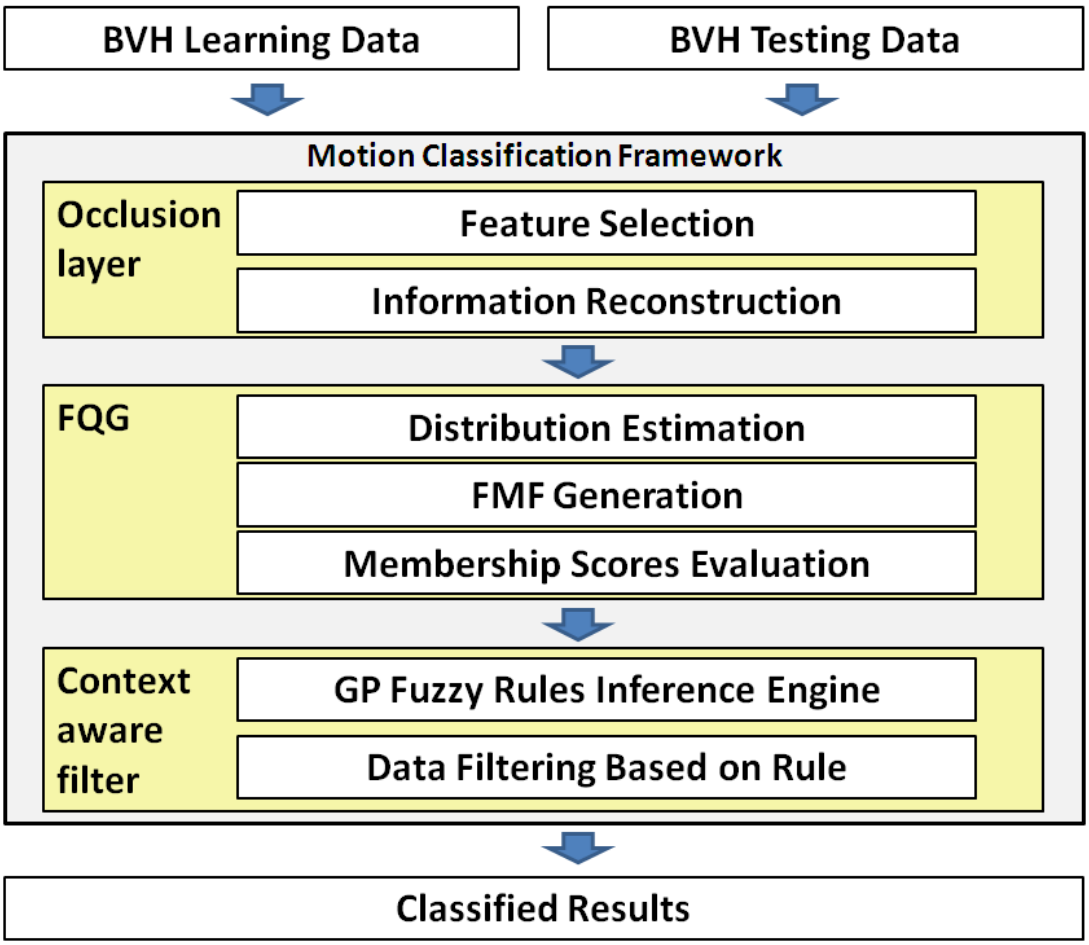


Figure 3.1: A block diagram of the motion classification framework

discrete qualitative input in a time dependent and context-aware fashion. As no initial prior knowledge is assumed, the high dimensionality and specificity of the problem might cause these rules to form nested logical structures of arbitrary complexity. In this section, FQG as a standalone modelling method is first introduced. Secondly, the context-aware post-processing filter is presented. Finally, the pre-processing module that deals with occlusion is detailed.

3.2 Fuzzy quantile generation

The novelty of FQG comes from the fact that it generates Fuzzy Membership Functions by building a simple and efficient connective between probabilistic and fuzzy paradigms that allows the classification of noisy, imprecise and complex motions while using learning samples of sub-optimal size. A detailed and formal description of the FQG modelling process is presented, and is then followed by a qualitative analysis of its flexibility as a machine learning technique.

3.2.1 Building a connection between a fuzzy membership function and a probability distribution

FQG is a method that builds and maps Fuzzy Membership Functions to probability distributions. A practical mapping from a Fuzzy Membership Function to a Normal Distribution is hinted at by Frantti (2001) in the context of mobile network engineering. Unfortunately, this work has its shortcomings as this a system does ignore motions which are over the extrema of the range of the learning sample. One solution to this problem would be to introduce a function that maps a degree of membership to the probability that values fall within a given cumulative probabilistic distribution. As a consequence, FQG presents such a method in four steps. First, probabilistic and fuzzy models in use must be identified. Secondly, the upper and lower bases of a Fuzzy Membership Function are initially estimated. Thirdly, the FMF shape is modified in order to follow a shift of the probability distribution inferred from the mean of the learning sample. Finally, the resulting model is used to classify instances by evaluating their membership scores.

3.2.1.1 Choosing the type of probability distribution and fuzzy membership function

Human motions are non-trivial to model because nobody moves exactly the same way, and even the same person, when repeating the same motion, does not perform it in a strictly identical fashion. Despite of this, we seem to be able to recognize a motion when performed by different actors at various speeds and in very different conditions.

From this observation, and the existence of the Central Limit Theorem, it is assumed that motions and by extension joints rotations, are somehow distributed.

The proposed method does not automate the choice of a distribution, so it is left to the user to decide which distribution best represents the domain the sample is extracted from. The normal distribution being commonly encountered in practice is used extensively throughout statistics as a simple model for complex phenomena. In the context of this experiment, it is chosen as the most likely base case and is used as a proof of concept. However, there are times when other distributions might offer a better alternative for different application domains. That is why others such as the Log-Normal, Exponential, and Laplace distributions are presented as possible alternatives in order to give a wider range of choices from a theoretical perspective. These can be used as a starting point to later map more complex distributions with multiple modes such as Beta and Sine distributions.

The Fuzzy Membership Function that maps to a distribution must be computationally efficient and suitable for noisy data. Fuzzy membership functions introduced by Zadeh (1986) can deal with imprecision and overlapping classes, which make them ideal for the classification of such data. The trapezoidal fuzzy membership is chosen here as a mean of representation primarily for its simplicity and efficiency with respect to computability. It offers a bit more flexibility than a triangular membership function, and set a membership score to a known move to 1 for an extended range of values. This makes it more practical for this data set, as a motion can be characterized by a range representing a set of fixed angular rotations. A standard trapezoid fuzzy-four-tuple (a, b, α, β) which defines a function that returns a degree of membership in [0,1] is defined in equation 3.1 and Figure 3.2.

$$\mu(x) = \begin{cases} 0 & x < a - \alpha \\ \alpha^{-1}(x - a + \alpha) & x \in [a - \alpha \quad a] \\ 1 & x \in [a \quad b] \\ \beta^{-1}(b + \beta - x) & x \in [b \quad b + \beta] \\ 0 & x > b + \beta \end{cases} \tag{3.1}$$

Once specific types of probabilistic and fuzzy representations are defined as suitable to the application domain, they need to be connected via the FQG framework. A simple connective is built between a Fuzzy Membership Function and a probability distribution that is based on Inverse Cumulative Distribution Function or Quantile

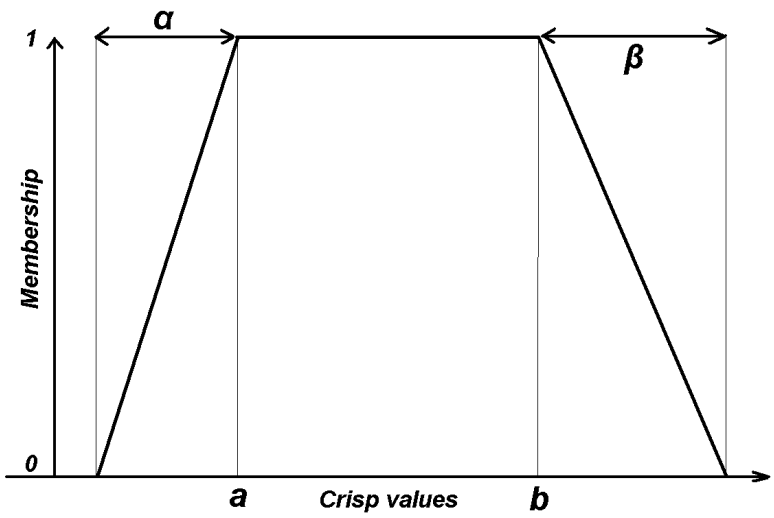


Figure 3.2: A trapezoid Fuzzy Membership Function

Function which is a standard probabilistic function (see Warren (2000)). Intuitively, FQG uses the quantile function to evaluate the shape of a Fuzzy Membership Function using metrics expressed in numbers of standard deviations. The shape of the fuzzy membership function is defined by:

- Seeing how the sample range fits a distribution and build a Fuzzy Membership Function accordingly.
- Shifting the Fuzzy Membership Function so that the mean of the sample overlaps with the true mean of the distribution.

Hence, the general shape of the Fuzzy Membership Function is defined and then adjusted in the next two steps.

3.2.1.2 Building the upper/lower base of the Fuzzy Membership Function

Let n_{min} and n_{max} be the minimum and maximum in number of standard deviations covered by the Fuzzy Membership Function. Then, $|n_{max} - n_{min}|$ quantifies in standard deviations the length of $|(b + \beta) - (a - \alpha)|$ in the Fuzzy Membership Function mapping the distribution. Let $s \in [0, 1]$ be a parameter describing the proportion of the population whose values are in the interval $[a, b]$, with a and b being respectively the

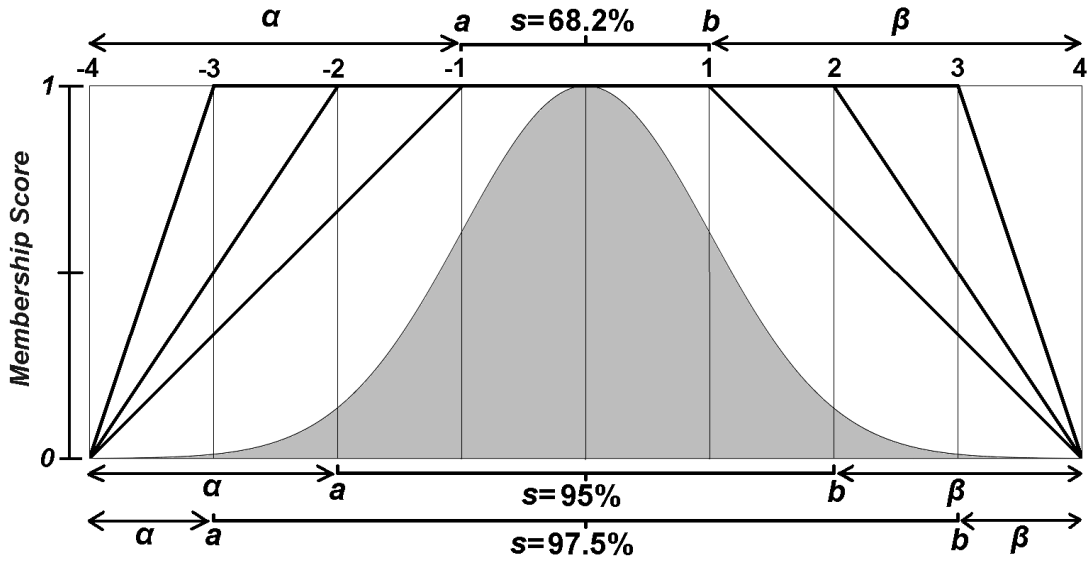


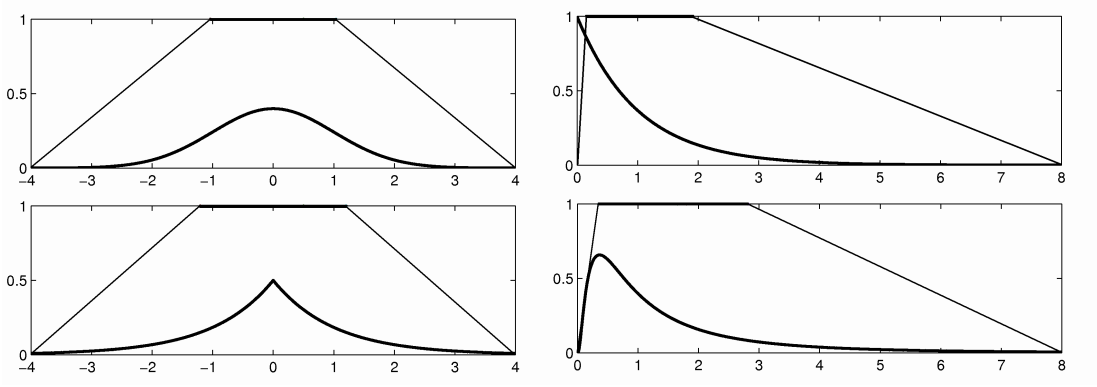
Figure 3.3: Influence of the s parameter (expressed in %) on the shape of the fuzzy membership function mapping to a Normal distribution

min and max values of the learning sample. Let $CDF^{-1}(x)$ be the Quantile Function also known as the Inverse Cumulative Distribution function of the applied distribution. Let $CDF^{-1}(0.5)$ be the position of the median of the distribution. Then, a, the lowest value of the learning sample, can be defined in term of the underlying distribution by $a = CDF^{-1}(0.5 - (s/2))$ and b, the highest value of the learning sample, can be defined in term of the underlying distribution by $b = CDF^{-1}(0.5 + (s/2))$. Also $a - \alpha$ can be defined in term of the underlying distribution by $a - \alpha = n_{min}$ and $b + \beta$ can be defined in term of the underlying distribution by $b + \beta = n_{max}$. Then, as illustrated in Figure 3.3 and 3.4, initially, α and β are defined in term of the underlying distribution by:

$$\begin{cases} \alpha = |CDF^{-1}(0.5 - (s/2)) - n_{min}| \\ \beta = |n_{max} - CDF^{-1}(0.5 + (s/2))| \end{cases} \tag{3.2}$$

Examples of a Normal, but also Laplace, Exponential and Log-Normal distributions are considered in Figure 3.4. For each of these distributions, the Quantile Function CDF^{-1} can be expressed as follows. In the case of a Normal distribution, of mean μ and standard deviation σ, $CDF^{-1} = \{x : CDF^{-1}(x|\mu, \sigma) = p\}$ where:

$$p = F(x|\mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{x} \exp(\frac{-(t-\mu)^2}{2\sigma^2})dt \tag{3.3}$$



(a) Normal and Laplace

(b) Exponential and Log-Normal

Figure 3.4: Building the upper/lower base of the FMF by mapping to selected distributions with $s = 0.7$

The result, x, is the solution of the integral equation above where you supply the desired probability, p. In the case of a Laplace distribution, of mean μ and standard deviation σ:

$$CDF^{-1} = \begin{cases} x = \mu + \sigma \cdot log(2p) & \text{if } p \leqslant 0.5 \\ x = \mu - \sigma \cdot log(2(1-p)) & \text{otherwise} \end{cases} \tag{3.4}$$

In the case of an exponential distribution, of parameter λ:

$$CDF^{-1}(p, \lambda) = \frac{-\ln(1-p)}{\lambda} \tag{3.5}$$

In the case of a log-normal distribution, of mean parameter μ and standard deviation σ:

$$CDF^{-1}(p, \sigma) = \exp(\sigma \Phi^{-1}(p)) \tag{3.6}$$

where Φ is the quantile function of the normal distribution.

This analysis is valid for distributions with only one mode (the present work excludes saddle shaped distributions). The ratio between the upper part and the lower part of the Fuzzy Membership Function being found, the initial characteristics of the Fuzzy Membership Function are set. However this is done under the assumption that the mean of the learning sample overlaps with the midpoint between its min and max ranges. As this is rarely the case, the FMF needs to be re-evaluated in order to follow a shift resulting from a discrepancy between the mean and the midpoint of the learning sample.

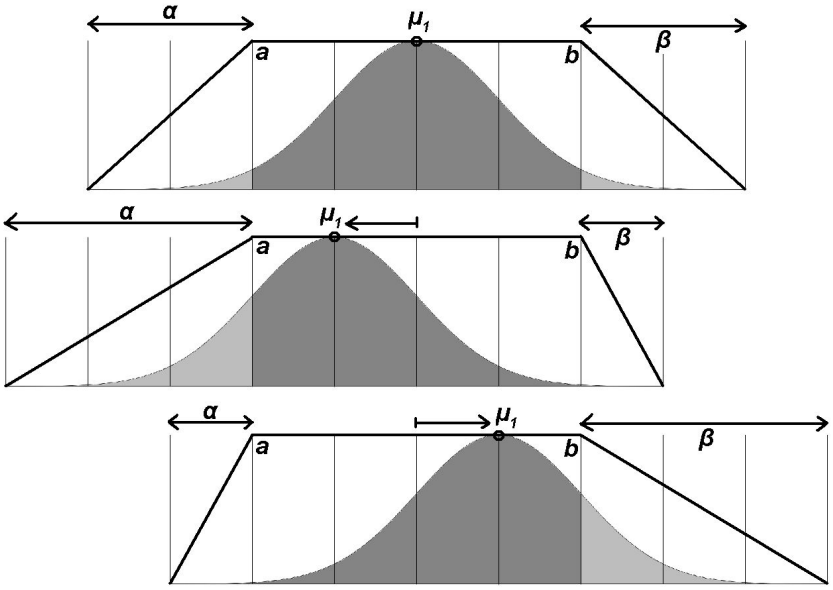
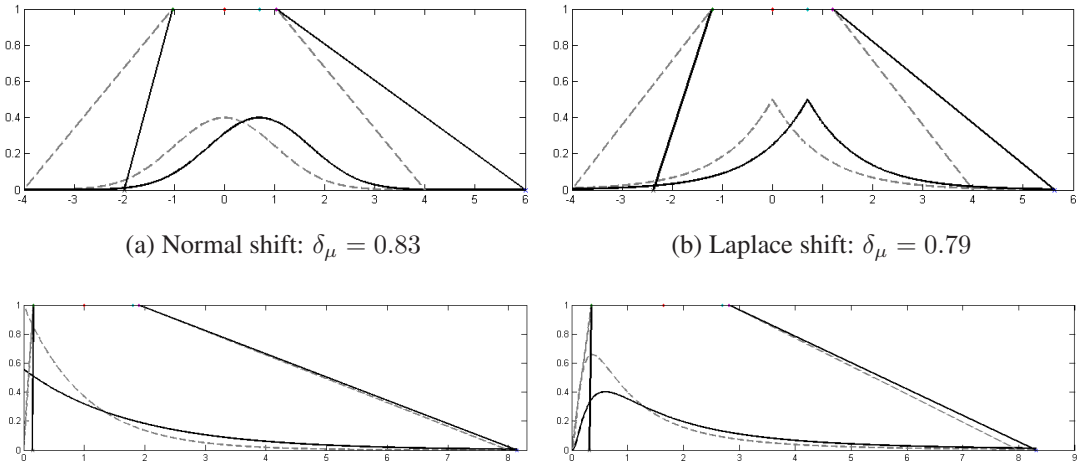


Figure 3.5: Displacing the mean of the learning sample shifts the distribution and deforms the fuzzy membership function

3.2.1.3 Deforming the fuzzy membership function to follow a shift of distribution

The Central Limit Theorem states that sampling distributions drawn from a uniformly distributed population will tend to form nearly perfect normal distribution when the sample size is large enough. This guarantees that a mean based on a randomly chosen sample of sufficiently large size will be remarkably close to the true mean of the population. Therefore, the mean of the "theoretical" underlying distribution is shifted to overlap with the mean of the learning sample. The shape of the corresponding Fuzzy Membership Function deformed in kind (see Figure 3.5). This shift of the distribution δ_μ is a ratio expressing how far the mean of the learning sample is from its midpoint which represents the mean of the underlying distribution in number of standard deviations. The FMF is deformed to follow the shift δ_μ in different ways depending on the type of distribution in use (see Figure 3.6. Let μ_0 be the initial mean of the distribution and μ_1 be the mean of the learning sample respectively expressed in number of standard deviations by $CDF^{-1}(\mu_0)$ and $CDF^{-1}(\mu_1)$. Let a and b the extrema of the



(a) Normal shift: $\delta_\mu = 0.83$

(b) Laplace shift: $\delta_\mu = 0.79$

(c) Exponential shift: $\delta_\mu = 0.89$

(d) Log-Normal shift: $\delta_\mu = 0.89$

Figure 3.6: Shifting distributions depending on δ_μ (plain lines are the shifted distributions and parameter $s = 0.7$)

learning sample such that: $a = CDF^{-1}(0.5 - (s/2))$, and $b = CDF^{-1}(0.5 + (s/2))$. Let α_0, and β_0 be the initial values of α, and β as defined in the previous step in equation 3.2.

If the chosen distribution has only one mode and is central symmetric, then δ_μ, and the new values of α and β can be computed as follows:

$$\begin{cases} \delta_\mu = (CDF^{-1}(\mu_1) - a)/(b - a) \\ \alpha = (1 - \delta_\mu)(\alpha_0 + \beta_0) \\ \beta = (\delta_\mu)(\alpha_0 + \beta_0) \end{cases} \tag{3.7}$$

In the case of right-skewed distributions of the type Exponential or Log-Normal, δ_μ, and the new values of α and β can be computed as follows:

$$\begin{cases} \delta_\mu = (CDF^{-1}(\mu_1) - CDF^{-1}(\mu_0))/(b - CDF^{-1}(\mu_0)) \\ \alpha = \alpha_0 - (\delta_\mu \cdot \alpha_0) \\ \beta = \beta_0 + (\delta_\mu \cdot \alpha_0) \end{cases} \tag{3.8}$$

Having presented a fuzzy membership function design methodology using metrics based on the probabilistic quantile function, the next step is to use it for classification.

3.2.1.4 Membership score evaluation

Instances of moves are classified by evaluating their membership scores to the Fuzzy Membership Functions generated by FQG. A posture being defined by a set of 57 Euler angles is modelled by 57 Fuzzy Membership Functions. Having no initial prior knowledge about the eventual predominance of some of these joints, the overall membership of a test instance to a known move is computed by calculating the average of all the 57 membership scores of the Euler Angles. This approach could probably be improved in the near future by introducing weighted average for certain joints (for example, the position of the elbow might be more important than the position of the knee when in guard). A parameter t expresses the membership threshold to a move. In practice, all frames with a membership score equal or greater than t are classified as belonging to that move. The lower the value of t, the lower is the selectivity of the classifier, and the higher the value of t, the more difficult it becomes for a move to be given a membership score of 1. When the same frame has a high membership score for several fuzzy sets representing different moves, an order of preference of these sets can be established by comparing the Euclidean distance of the observed data to the centroid of each fuzzy set. The existence of t allows the introduction of a convenient a-posteriori way to fine tune parameters in order to tailor the precision of every model to the quantity of available learning data for each move. If results show that the relative size s of a learning sample for given move was over-estimated, the membership scores m_i can be re-scaled by fine-tuning the thresholds t_i linked to each move i, effectively "truncating" the upper part of the Fuzzy Membership Functions without having to recalculate the model (see equation 3.9).

$$m_i = (m_i - t_i) \,/\, (1 - t_i) \tag{3.9}$$

3.2.2 Qualitative analysis: the flexibility of fuzzy quantile generation as a machine learning technique

Unlike methods such as PCA that sometimes lose crucial information in the process of dimensionality reduction, Fuzzy Quantile Generation keeps the initial number of dimensions when building the model. The method decomposes what would normally be a Gaussian Mixture of a number x of m-dimensional Normal distributions into $x \times m$ Fuzzy Membership Functions (see Figure 3.7). In this study, nineteen joints with

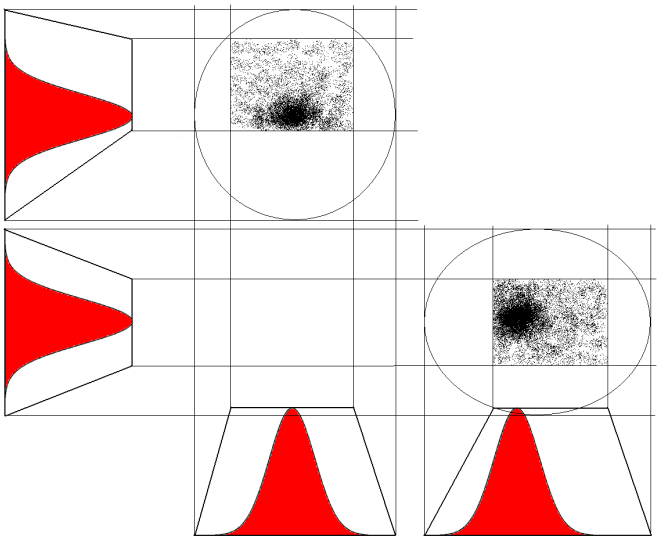


Figure 3.7: Decoupling feature distributions - a comparison between FQG and Gaussian Mixture

3 degrees of freedom each represent in total 57 continuous values expressing rotation angles are modelled by 57 fuzzy membership functions. It can be observed that FQG has the ability to decouple conditional feature distributions into m one-dimensional set, but it is also able to link these features together by estimating a measure that combines their memberships. Having no initial prior knowledge about the eventual predominance of some features, the measure is chosen to simply be an average of the membership scores of all Euler Angles. If this is not as fine grained as a network of probability distributions, this is enough to make features interdependent and it can also be made more complex by introducing weights depending on the importance of certain joints. The flexibility of a machine learning method is generally determined by how successfully it can be applied to different application domains. Empirically speaking, making use of supervised machine learning techniques generally involves testing a data sample with different parameter values in order to reach an optimal combination leading to a maximized performance of the given system. Two of the contributing factors to the degree of usability for such methods are the number of parameters in use and the sensitivity the system exhibits to slight variations in parameters values. In other words, if the classifier is parameter dependent like most machine learning techniques, it is interesting to know the relationship between these parameters, and

how do parameters variations influence the overall system performance. FQG is based on two parameters which, combined with input data, produce a classification with a certain degree of accuracy. The first parameter $\bar{s}$ evaluates the "relative size" of the sample. This ratio represents the average of different s values over m features as defined in section 3.2.1.2 and expresses the proportion of the population whose value is in the intervals defined by the learning sample. Intuitively, $\bar{s}$ tells how much of the population of correct moves the stances present in the learning sample represent. The second parameter t expresses the membership threshold in use with the classifier as described in section 3.2.1.4. A membership threshold of 0.95 means for example that all frames which have a membership score greater than 95% to a move are identified as belonging to this move. When classifying different types of movements for a given specific accuracy, there seems to be an empirical relationship between the parameter $\bar{s}$ and the membership threshold t. This relation can be generalised into an equation where a function g maps $\bar{s}$ to the membership threshold t for a given accuracy such that: $g(\bar{s}) = t$. Figure 3.8 shows different plots of the function g mapping the parameter $\bar{s}$ on the x-axis to the threshold t on the y-axis using three different data sets. One can observe that the slope of the function g is always negative. That is to say that for any parameter $\bar{s}$, $\dot{g}(\bar{s}) < 0$. For a given accuracy, the threshold t seems to vary as a function of $\bar{s}$ following a general curve with an equation of the form:

$$t = \delta + 1/(\gamma \times \log \bar{s}) \tag{3.10}$$

where δ and γ are constants linked to the dataset considered. In economics, the concept of elasticity presented in Case & Fair (2003) is used to measure the responsiveness of a function to changes in parameters in a relative way. In the context of machine learning, this same concept can be reused to evaluate if the threshold t is $\bar{s}$-elastic. It becomes noticeable that the elasticity is poor when using a very high $\bar{s}$ value (superior to 0.95). The maximum elasticity is obtained when $\bar{s}$ is between 40 and 95%. This means that in this data set, the variations of the $\bar{s}$ parameter are more likely to influence the threshold t if $\bar{s}$ is kept between 0.4 and 0.95. Regarding the relationship between accuracy and parameters, the higher the value of $\bar{s}$ is, the more likely the accuracy is to get lower. This is expected as the sample is of limited size, over-estimating its relative-size will damage the accuracy of the classifier. The loss in accuracy is determined as a function

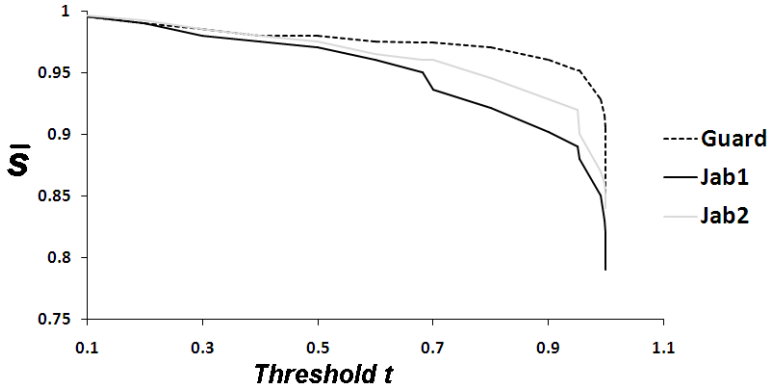


Figure 3.8: Empirical relationship between two parameters: function mapping the relative-size $\bar{s}$ to the threshold t

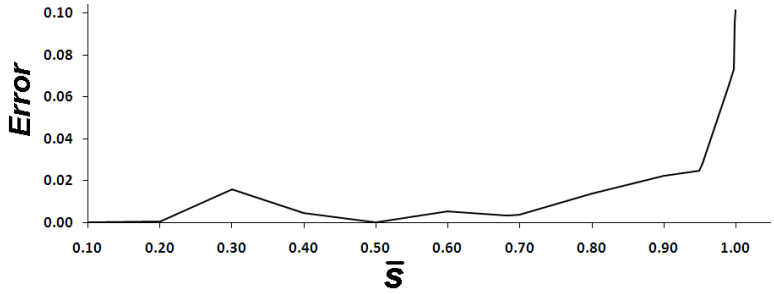


Figure 3.9: Function mapping the loss of accuracy when over-estimating the parameter $\bar{s}$ when recognising a "Guard" stance

of the relative-size parameter $\bar{s}$. When classifying a guard, the error is rising with over-estimation of $\bar{s}$ up to a maximum of 10% which is quite a small (see figure 3.9) error rate for parameters that take extreme values.

3.3 The context-aware genetic programming filter

The standalone classifier labels frames by computing membership scores to known motions. However, this classification is done frame by frame and is not time-based. It does not take into account previous or following motions. In this context, it becomes necessary to smooth the qualitative output using context-aware fuzzy-rules. These rules might combine very specific operators such as logical functions, measurements of speed, and ranked as first, second or third best choices for a motion.

No initial prior knowledge about how motion can be smoothed is assumed. Considering the fact that there is a relatively high number of input symbols (around 57), standard approaches like the Takagi-Sugeno-Kang (TSK) or Mamdani models become unsuitable because of the exponential growth of the number of rules. Similarly, the high dimensionality and specificity of the problem where results might form nested logical structures of arbitrary complexity make mainstream inference methods such as Fuzzy Neural Networks in Gobi & Pedrycz (2007) or standard Evolutionary Algorithms in Yu *et al.* (2003) and Belarbi *et al.* (2005) difficult to reuse in this context. Genetic Programming (GP) is a branch of evolutionary computation described in Koza (1992) that evolves programs represented as tree structures and can cut through vast search spaces to suggest solutions that optimise a fitness function. In existing Genetic Programming packages, the generation and recombination of individuals tend to focus on producing random trees, and the mechanisms involved in the production of syntactically correct individuals are generally limited due to computational and usability problems. For the purpose of this research, a Strongly-Typed Genetic Programming open-source distribution was built (see Khoury (2009) and appendix E) that allows the user to easily evolve populations of trees with precise grammatical and structural constraints.

3.3.1 Python Strongly Typed gEnetic Programming

PySTEP is a light Genetic Programming Application Programming Interface (API) using the Python programming language that focuses on facilitating the creation of building blocks and rules that define individuals during the evolutionary process. Although this "Strongly-Typed" aspect defined its originality as an evolutionary tool and defines specificities at the representational and algorithmic level, the general evolutionary process corresponds to a classic Genetic Programming flowchart as shown in Figure 3.10. The three distinctive features that characterise PySTEP are presented in this section: firstly, the specific representation scheme; then syntactic ordered grammar rule system; and finally, the modified operators.

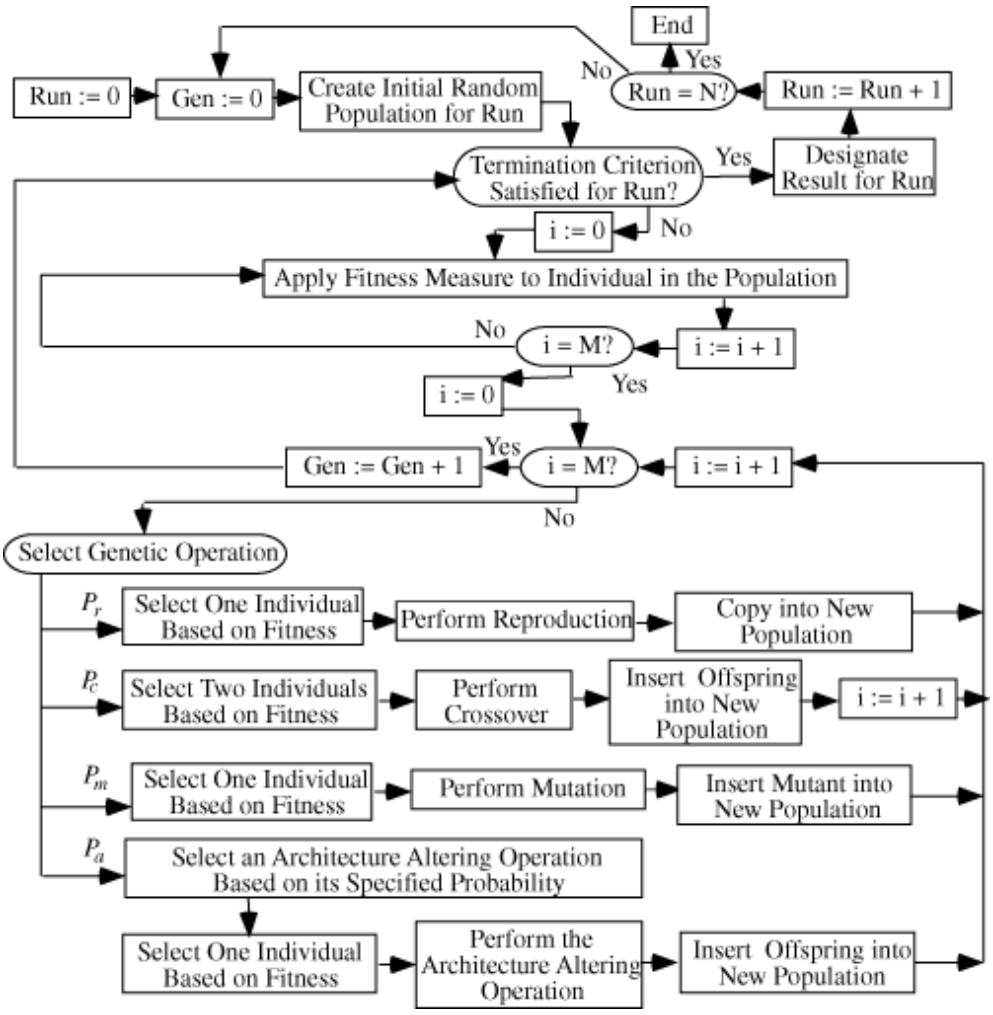


Figure 3.10: Flowchart presenting executional steps for Genetic Programming - image extracted from Koza (2007)

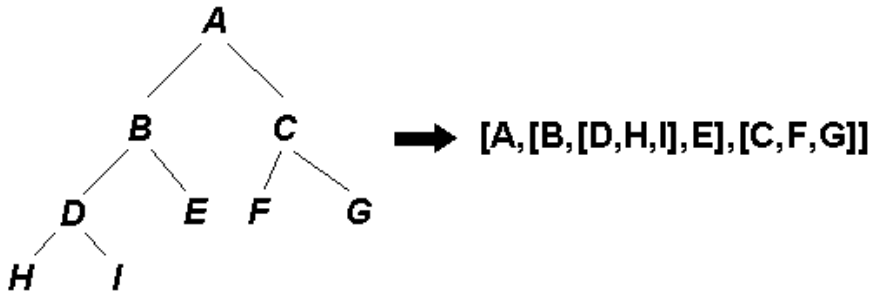


Figure 3.11: PySTEP representation of a basic tree in a nested list format - the first element of a list or a nested list is always a head node

3.3.1.1 A specific representation scheme

PySTEP represents tree data structures as nested lists where the first element of a list or a nested list is always a head node. A simple tree structure can be represented in PySTEP as shown in Figure 3.11 where each letter represents a different node that corresponds to an object in the Python programming language. In PySTEP, a node object is a tuple, e.g. (0,2,'root') that is composed of three elements. The first element represents the type of node:

- 0 is a root branch, i.e. the first node at the very top of the tree.
- 1 is a branch node that is a function, i.e. it can take as children variables, constant, terminal nodes, or other functions.
- 2 is an Automated Defined Function Defining Branch that is a value that maps to a sub tree that can be reused as a terminal.
- 3 is a terminal node that is a variable.
- 4 is a terminal node that is a constant.
- 5 is an Automated Defined Function Leaf that is a sub tree that can be reused as a terminal.

The second element is the arity of the node , i.e. the number of children. The third element is the name or unique identifier of the Node that refers to a specific operation

Table 3.1: Nested lists and corresponding mappings of tree depth structure.

Nested lists	Corresponding mappings of depth structure.
$[1, 2, 3, 4, 5, 6, 7, 8, 9]$	$[([0], 9)]$
$[1, [2, 3], 4, 5, 6, 7, 8, 9]$	$[([0], 8), ([1], 2)]$
$[1, [2, 3, 7], 4, 5, 6, 7, 8, 9]$	$[([0], 8), ([1], 3)]$
$[1, [2, 3, 7], 4, [5, [6, [7, 8, 9]]]]$	$[([0], 4), ([1], 3), ([3], 2), ([3, 1], 2), ([3, 1, 1], 3)]$

implemented in the programming language, e.g. $(1, 2, '+')$ is a function branch node with two children and unique identifier '+' that corresponds to the addition operation implemented as a function in the programming language. So, following this way of coding nodes, a basic equation like $x \times (x^2 + x)$ could be represented by the following nested list: *[(0,1,'root'),[(1,2,'*'), (3,0,'x'), [(1,2,'+'), [(1,1,'square'), (3,0,'x')], (3,0,'x')]]].*

There is no function included in the object-nodes to iterate through the tree, unlike other genetic programming based tree data structures. This is intended in order to keep the nodes-objects composing trees as light-weight as possible, so that the resulting memory space occupied by trees stays minimum. This simplifies the storage of populations of tens of thousands of trees of significant depth in one data base by lowering unreasonable space requirements. The downside is that a mechanism is needed to cope with the absence of node iterator functions in order to identify the indices of nodes at a given level of depth (this type of operation is used constantly during crossover or sexual recombination operations). In order to keep the representation relatively compact, a list of tuples containing integer numbers presenting the numbers of nodes per sub list by depth is generated while avoiding computationally expensive recursive tree parsing. The first element of each tuple contains information about the depth and position of a head node, while the second contains the number of elements. This occupies some storage space, but not as much as a heavily object oriented node. Examples of such structures are visible in Table 3.1.

During evolution, each individual is stored in a data base with the following elements: individual identifier number, nested list representing the tree data structure, list containing the depth structure of the tree, integer value expressing the tree final depth, Boolean value expressing if the fitness of the individual has been evaluated, and the

fitness score of the individual as computed from the fitness function.

3.3.1.2 A grammar rule system using ordered function and terminal sets

The grammar rules used to define the construction are defined by nested tuples where each member is represented by a function set (i.e. the set of branch nodes) and a terminal set (i.e. the set of leaf nodes). The order of terminal and function sets for children node can be pre-determined in order to build ordered structures i.e. an 'If' node must always precede 'Then' nodes. Every child of every node in the tree has a list of possible function nodes and terminal nodes. A given node will be described as a list of tuples where each tuple contains the function and terminal set for a child. e.g. a 'dot' node expressing a dot product will have two children, each one belong to a specific set. The first child must be chosen from specific function and terminal sets and the second child must be chosen from different sets. So a 'dot' product node will be described by: *[(FunctionSet1,TerminalSet1),(FunctionSet2,TerminalSet2)]*. When using the GP in the context of polynomial regression, a polynomial of the type: $x^3 + x^2 + cos(x)$ can be generated by using one variable x and the following mathematical operators: $'+','-','neg','*','square','cos','sin'$. The PySTEP rules used to generate such polynomial will take the form of a list of nodes names with corresponding function and terminal sets as shown in the very brief commented Python code below in table 3.2.

This representation allows ordered input to be given to function, which might give different results depending on the input combination. This feature is quite unique in Genetic Programming and makes the PySTEP evolutionary package competitive.

3.3.1.3 The modified generation and manipulation of trees

Conventional processes used to generate trees in Genetic Programming are also present in PySTEP. The tree building algorithms follow the standard Half, Full, and Ramped half-and-half procedures explained in Koza (1992) with one alteration: when a random function or terminal nodes is selected to build a tree structure, it has to be guaranteed compatible with its direct parent node.

Genetic Programming operators used to manipulate trees such as crossover, mutation, and reproduction also exist in PySTEP, but are modified in the case of crossover

Table 3.2: Example of PySTEP rules generating syntactically correct polynomials

1: $funcSet \Leftarrow [(1, 2,' +'), (1, 2,' *'), (1, 1,' square'), (1, 2,' -'), (1, 1,' cos'),$
$(1, 1,' sin'), (1, 1,' neg')]$ {default function set applicable for branches nodes}
2: $termSet \Leftarrow [(3, 0,' x')]$ {default terminal set applicable for leaf nodes}
3: $'root' \Leftarrow [(funcSet, termSet)]$ {a root node can have one child from either the default function or terminal sets}
4: $'+' \Leftarrow [(funcSet, termSet), (funcSet, termSet)]$ {these nodes can have two children from either the default function or terminal sets}
5: $'*' \Leftarrow [(funcSet, termSet), (funcSet, termSet)]$
6: $'-' \Leftarrow [(funcSet, termSet), (funcSet, termSet)]$
7: $'square' \Leftarrow [(3, 0,' x')]$ {default terminal set applicable for leaf nodes}
8: $'neg' \Leftarrow [([(1, 2,' +'), (1, 2,' *'), (1, 2,' -'), (1, 1,' cos'), (1, 1,' sin')], termSet)]$
{these nodes can have two children from either a specific function set or a default terminal set}
9: $'cos' \Leftarrow [([(1, 2,' +'), (1, 2,' *'), (1, 2,' -'), (1, 1,' sin'), (1, 1,' neg')], termSet)]$
10: $'sin' \Leftarrow [([(1, 2,' +'), (1, 2,' *'), (1, 2,' -'), (1, 1,' cos'), (1, 1,' neg')], termSet)]$

and mutation. In mutation, a single individual is selected based on fitness. At a randomly chosen mutation point, the subtree rooted at that point is replaced by the same modified process used for tree generation. The crossover or sexual recombination operation combine features from two parents which are trees of different shapes in order to produce two distinctive offsprings. This operator used generally around 90% of the time during evolutionary runs has been heavily modified in PySTEP. Making the new offsprings compliant with the syntactic rules defining a tree is delicate and computationally expensive because the parts of the parent trees which are swapped in order to produce an offspring are not always grammatically compatible between themselves. This incompatibility has to be a) detected by analysing the syntactic validity of the new tree, and b) corrected in order to generate potentially useful solutions. Depending on the complexity of the rules, this problem cannot always be solved with great speed. A balance is struck between valid syntax and fast computation by setting the following rule: once the system has tried 100 times without success to produce a rules-compliant tree using crossover, the unfit offsprings are substituted with a mutated tree. In order to facilitate the correction process, a mapping of function branches that can be

swapped between parents is established. The PySTEP crossover operator is detailed in the pseudocode in table 3.3.

3.3.2 Genetic programming settings in the context of the framework

Figure 3.12 shows a typical input obtained from the FQG classifier when identifying a sequence of Jab-Cross combinations separated by Guard positions. The first, second and third best membership scores corresponding to the three best estimated labels for each frame are used as multiple inputs by the GP filter. These membership scores are rescaled following the procedure detailed in section 3.2.1.4. On the right part of the figure, each frame is coloured corresponding to the move with the highest membership score (e.g. blue for jab, black for guard, orange for cross). These sequences of guards separated by jab-cross combinations are identified mostly correctly as the corresponding colours appear in the right order: black separating patches of blue and orange. The classification is not perfect, as some isolated frames are showing unexpected colours, e.g. some frames that should be orange (cross) are coloured in red (right hook). The context aware filter aims at improving these results by changing these misclassified anomalies. The GP system evolves logic rules (see Figure 3.13) that transform the qualitative output of each frame. The GP terminal and function sets are detailed as follows in Table 3.4 and the corresponding Python code that captures the syntactic constraints is visible in E. Some operators return the groups of frames that are identical for a given duration, e.g. *is_short* expresses a duration of less than 5 frames. Other operators return the groups of frames that belong to the first , second and third best membership scores of a motion e.g. *membership_2(left_hook)* return groups of frames with the second best membership score for a motion as a left hook. Others return the groups of frames that have specific previous motions e.g. *left_2(guard,jab)* returns groups of frames preceded in order by a guard and then a jab motion. Logical operators are present (e.g. and, or, not). Logical structures of the type *If Then Replace X by Y* use the previous operators to identify groups of frames and replace their best motion membership score by a different one, e.g. *If (membership_2(guard))Then Replace (jab) by (Cross)*. One individual or tree consists of four interconnected *If Then Replace X by Y* rules. The GP parameters are:

Table 3.3: pseudo code of the PySTEP crossover operator

let $C_{Depth_{P1}}$ be a random depth in first parent tree.
let $C_{Depth_{P2}}$ a random depth in second parent tree such that (depth of $parent1$-$C_{Depth_{P1}}$) + $C_{Depth_{P2}} \leqslant maxDepth$
randomly choose crossover nodes among those located at depth $C_{Depth_{P1}}$ and $C_{Depth_{P2}}$ in both parents
find the subtrees $fragmentP1$ and $fragmentP2$ associated with respective crossover nodes in each parent
get the parent nodes of each subtree
for each branch node in $fragmentP1$ and in $fragmentP2$ **do**
  if not present in the list of authorized branches nodes of its parent node **then**
    unauthorized branches nodes - set boolean values to false for each fragment
    if swapping branch nodes between fragments is permitted **then**
      for for each unauthorized branches node **do**
        try to swap branch nodes between fragments by following the list of authorised crossover swaps.
        if all swaps successful **then**
          return modified fragments and set boolean values to true
        else
          return unmodified fragments and set boolean values to false
        end if
      end for
    end if
  end if
end for
for terminal node in $fragmentP1$ and in $fragmentP2$ **do**
  if not present in the list of authorized terminal nodes of its parent node **then**
    set boolean values to false
  end if
end for
produce two offsprings that aggregates parents with modified fragments
return the offsprings resulting from the crossover with boolean values indicative of their structural compliance

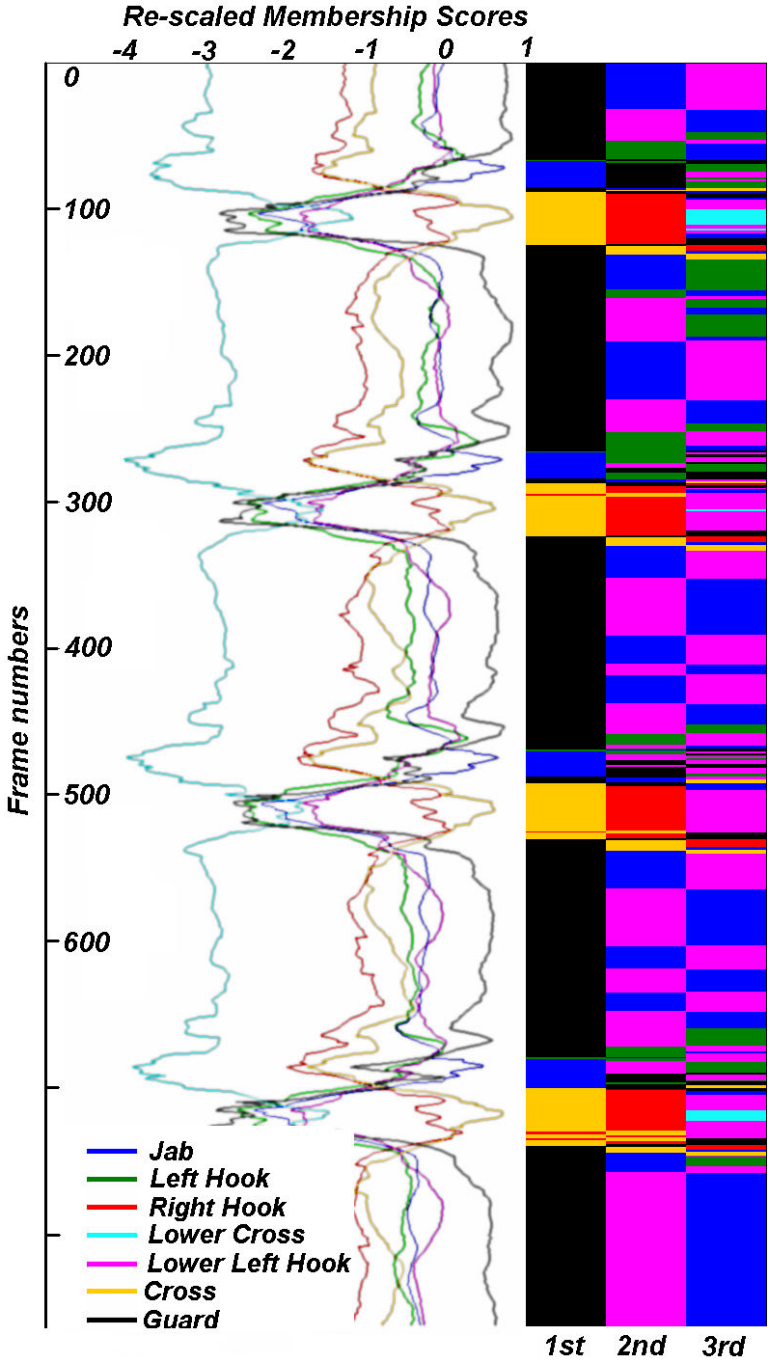


Figure 3.12: Example of a sequence of Jab-Cross combinations separated by Guard positions. The three labels with the highest membership scores are used as inputs for each frame by the filter.

Table 3.4: Grammar rules detailing possible function and terminal children nodes for each parent node.

Parent Nodes	Associated Children Nodes	
	Function nodes	Terminal Nodes
Root	If Then Replace X by Y	Empty
If Then Replace X by Y	Membership_1 Membership_2 Membership_3 Left_1 Left_2 Left_3 Right_1 Right_2 Right_3 And Or Not	Is_Short Is_Medium Is_long Mvt Type 1 Mvt Type 2 Mvt Type 3 Mvt Type 4 Mvt Type 5 Mvt Type 6 Mvt Type 7
Membership_1 Membership_2 Membership_3 Left_1 Left_2 Left_3 Right_1 Right_2 Right_3	Empty	Mvt Type 1 Mvt Type 2 Mvt Type 3 Mvt Type 4 Mvt Type 5 Mvt Type 6 Mvt Type 7
And Or	Membership_1 Membership_2 Membership_3 Left_1 Left_2 Left_3 Right_1 Right_2 Right_3 And Or Not	Is_Short Is_Medium Is_long
Not	Membership_1 Membership_2 Membership_3 Left_1 Left_2 Left_3 Right_1 Right_2 Right_3 And Or	Is_Short Is_Medium Is_long

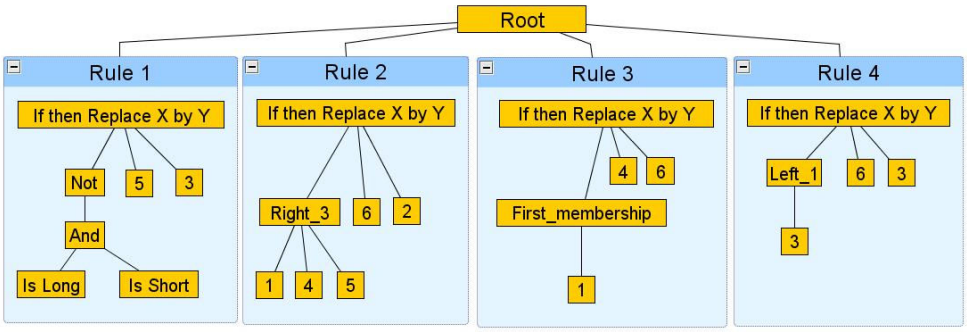


Figure 3.13: A Typical set of Rules Generated by PySTEP

- A population size of 1000 individuals, with a maximum number of 400 generations per evaluation.
- Tournament selection of size seven and selection probability 0.8.
- The probabilities of crossover, mutation, and reproduction are respectively 0.5, 0.49 and 0.01.

The fitness function simply sums the number of frames that have a different qualitative output from the frames present in a "perfect" sequence as defined by a human observer. So if n is the total number of frames observed, Δ is one unit that expresses a difference of classification on one frame between FQG and the human observer, then the fitness F is such that:

$$F = \sum_{r=1}^{n} \Delta_r \tag{3.11}$$

The Koza operators such as the tree building "ramped half and half" algorithm, and operators such as crossover, mutation are all modified with a Strongly-Typed flavour. In practice, this means that the structure of all the individuals generated will be defined by a set of rules. These rules associate for each parent node an ordered set of children nodes. Each parent node maps to a list of possible children nodes which can be either function nodes or terminal nodes. The performance of the context-aware GP filter is evaluated in section 4.3.

3.4 Dealing with partial or occluded data

Occlusion is a standard problem when dealing with the classification of real human motion data. In order to successfully classify a motion which has missing observable features, one needs to deal with the uncertainty introduced by the occluded data. Two different procedures are combined in order to deal with occlusion: feature selection and feature reconstruction. The former deals with the optimisation of the feature selection phase via the introduction of an improved measure of similarity. The latter is about the reconstruction of plausible rotational data from occluded joints using a modified version of Fuzzy Robot Kinematics.

3.4.1 Reducing uncertainty using fuzzy rough feature selection

A survey by Radzikowska & Kerre (2002) presents many possible attribute selection techniques for discrete data. For the purpose of this work, Fuzzy Rough Feature Selection (FRFS) introduced by Jensen & Shen (2007, 2009) is chosen as it seems to offer good performances compared to other techniques and allows real-valued noisy data to be reduced without the need of user supplied information. When working with FRFS, one notices that the quality of the results relies on the estimation of dependency degrees between attributes which itself is based on the measure of how similar two objects x and y can be for a feature a. That is to say, the fuzzy similarity relation between two objects for a given attribute can determine the efficiency of the whole attribute selection process. In this work, fuzzy rough feature selection is first introduced. Then, several fuzzy similarity relations are presented and compared to the fuzzy Laplace Similarity relation. Comparison is done by looking at the accuracy of a Naive Bayesian classifier over several datasets. It is important to know which joints are the most relevant to differentiate motions. Reducing the features to a reduct subset with minimal information loss (according to rough set theory) as described by Jensen & Shen (2009) provides a way to estimate which joints are the most important. If none of these essential joints are occluded, the uncertainty of the classification can be lowered. The importance of these joints when computing a distance between membership scores of motions can be taken into account. Fuzzy Rough Feature Selection is based on fuzzy lower and upper approximations $\mu_{\underline{R_P}}(x)$ and $\mu_{\overline{R_P}}(x)$ as defined in Jensen & Shen (2007), where

a T-transitive fuzzy similarity relation approximates a fuzzy concept X.

$$\mu_{\underline{R_P}}(x) = \inf_{y \in \mathbb{U}} I(\mu_{R_P}(x,y), \mu_X(y)) \tag{3.12}$$

$$\mu_{\overline{R_P}}(x) = \sup_{y \in \mathbb{U}} T(\mu_{R_P}(x,y), \mu_X(y)) \tag{3.13}$$

Here, I is a fuzzy implicator and T a t-norm. R_P is the fuzzy similarity relation induced by the subset of features P.

$$\mu_{R_P}(x,y) = \bigcap_{a \in P} \{\mu R_a(x,y)\} \tag{3.14}$$

where $\mu R_a(x,y)$ expresses the degree to which objects x and y are similar for a feature a. The fuzzy positive region can be defined as:

$$\mu_{POS_{R_P(Q)}}(x) = \sup_{X \in \mathbb{U}/Q} \mu_{\underline{R_P X}}(x) \tag{3.15}$$

The resulting degree of dependency is:

$$\gamma' P(Q) = \frac{\sum_{x \in \mathbb{U}} \mu_{POS_{R_P(Q)}(x)}}{|\mathbb{U}|} \tag{3.16}$$

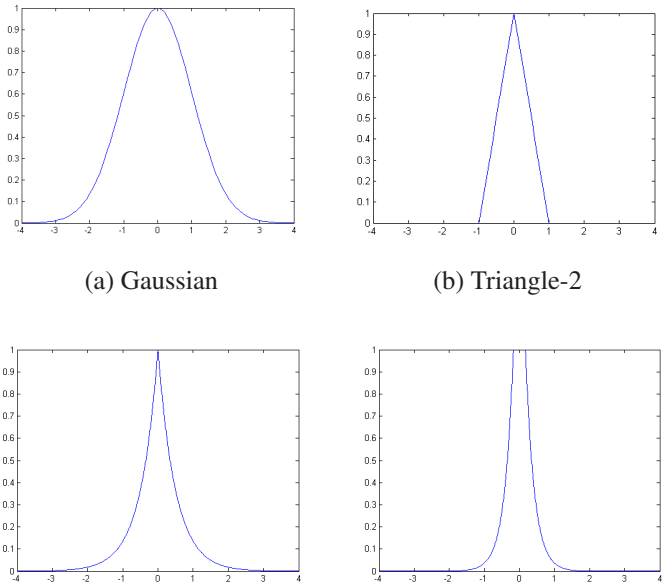
Core features Jensen & Shen (2007) may be determined by considering the change in dependency of the full set of conditional features when individual attributes are removed:

$$Core(\mathbb{C}) = \left\{ a \in \mathbb{C} | \gamma'_{\mathbb{C}-\{a\}}(Q) < \gamma'_{\mathbb{C}}(Q) \right\} \tag{3.17}$$

The degree of similarity $\mu R_a(x,y)$ can be expressed by different fuzzy similarity relations. Depending on which relation is used, a different subset of attributes will be selected, influencing in turn directly the performance of a classifier. The Gaussian (equation 3.18) and the Triangular-2 (equation 3.19) Fuzzy Similarity Relations shown in Figure 3.14 are used extensively Jensen & Shen (2007) by default in FRFS.

$$\mu_{R_a}(x,y) = \exp(-\frac{(a(x)-a(y))^2}{2{\sigma_a}^2}) \tag{3.18}$$

$$\mu_{R_a}(x,y) = max(min(\frac{(a(y)-(a(x)-\sigma_a))}{(a(x)-(a(x)-\sigma_a))}, \frac{((a(x)+\sigma_a)-a(y))}{((a(x)+\sigma_a)-a(x))}), 0) \tag{3.19}$$



(a) Gaussian (b) Triangle-2

(c) Laplace for $\lambda = 0.5$ (d) Laplace for $\lambda = 0.25$

Figure 3.14: Gaussian, triangle-2 and Laplace fuzzy similarity relations

The Fuzzy Laplace Similarity relation (see Figure 3.14) can be expressed in function of two parameters: the location parameter of the Laplace Distribution μ_0 and the scale parameter λ. Equation 3.20 shows the general case while equation 3.21 represents the most used case when $\mu_0 = 0$ and $\lambda = 0.5$.

$$\mu_{R_a} = \min\left(\left[\frac{\exp\left(\frac{-||a(x)-a(y)|-\mu_0|}{\lambda}\right)}{2\lambda}\right], 1\right) \tag{3.20}$$

$$\mu_{R_a} = \min(\exp(-2 \cdot |a(x) - a(y)|), 1) \tag{3.21}$$

It is worth noticing that, in this case, the Laplace similarity seems to combine the large base of the Gaussian similarity relation with the tight pick of the triangle-2 relation (see Figure 3.14). This specific shape can be modified when changing λ. The performance of Laplace similarity is compared with Gaussian and triangle-2 fuzzy similarity relations in section 4.4.1.

3.4.2 Reconstructing plausible rotational data from occluded joints using fuzzy robot kinematics

The novelty of the feature reconstruction mechanism can be established by the fact that the majority of the existing studies done on occlusion focus on object tracking in video sequences. These approaches generally use Kalman Filter for tracking markers of interest that can take the form of blobs in Gabriel *et al.* (2003), image features Utsumi & Ohya (1999), or silhouette images Ueda *et al.* (2003) derived from video sequences. The use of 3d based representation can be found in Kakadiaris & Metaxas (2000) where a three-dimensional pose of the subject's upper and lower arms is recovered and computed in order to create video animation sequences, and Utsumi & Ohya (1999) where a small number of reliable image features is needed to estimate 3d hand postures with a Fourier descriptor. Many possible Euler Angles combinations can define one given rotation. In order to reduce the search space, its granularity is increased by using Fuzzy Qualitative Euler Angles. This work is based on the Fuzzy Qualitative Trigonometry representation system exposed in Liu (2008) and Liu *et al.* (2008). It does not include Fuzzy Qualitative Denavit-Hartenberg kinematics structure as we consider here cases where one joint is occluded. First, this method by which possible fuzzy qualitative states representing plausible rotations are inferred is explained. Secondly, the way of ranking possible solutions using existing and reconstructed rotational data is detailed.

The skeletal representation is normalised, so the distances to the occluded joints are known. This means that the position of the occluded joint lies along a circle of centre O defined by these distances (see figure 3.15 for an example of occluded elbow joint). This circle is divided in $n = 16$ possible spheres where each sphere represent the volume of a possible position. Considering 8 different cameras placed evenly placed around the stick-figure at belt height, the occlusion of all the spheres is reconstructed in 3d for all 8 points of views. Only spheres which are occluded by other limbs are considered as possible joints positions. This significantly reduces the search space. The coordinates of the centre of each occluded sphere are extracted. For each centre, one crisp joint position is extracted and then converted into all possible corresponding Fuzzy Qualitative Euler Angles, giving a limited number of plausible suggestions for the Euler Angles of the hidden joint. This conversion of one Crisp Euler Angle rotation

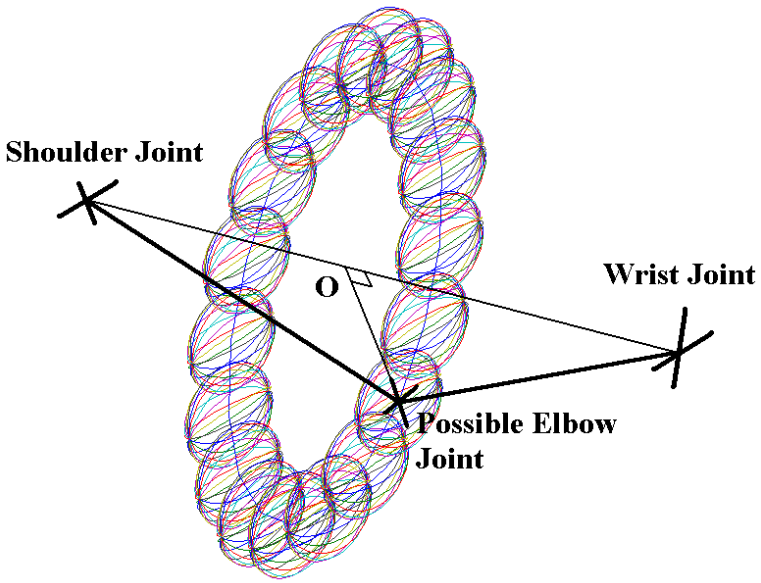


Figure 3.15: Possible qualitative locations for an occluded elbow joint

into the set of all equivalent fuzzy qualitative Euler angles is done via a pre-computed mapping table. Let the unit circle be divided into $n = 16$ fuzzy qualitative angles (figure 3.16). Each of the 3 fuzzy qualitative Euler Angle expressing a rotation will have n possible discrete states. Therefore, there exist $n!/(n-3)!$ possible fuzzy qualitative Euler Angles combinations, that is to say 3360 combinations. Each combination will correspond to a resulting fuzzy qualitative surface on the unit sphere (see figure 3.16) defined as follows.

First, the area of membership 1. After rotation R_Z around the Z axis, it is defined by the segment $P1_{R_Z} = [R_{Z2}, R_{Z3}]$. Let $R_{Yi} \circ R_{Zi}$ be the ordered function composition that designates combined rotations R_Z and R_Y around the Z and Y axes. Let $R_{Xi} \circ R_{Yi} \circ R_{Zi}$ be the ordered function composition that designates combined rotations R_X R_Z and R_Y around the X, Z and Y axes . The area of membership 1 after rotation R_Z and R_Y is described by the polygon $P1_{ZY}$with vertices $(R_{Y2} \circ R_{Z2}, R_{Y3} \circ R_{Z2}, R_{Y2} \circ R_{Z3}, R_{Y3} \circ R_{Z3})$. The area of membership 1 after rotation R_X R_Z and R_Y is : the area

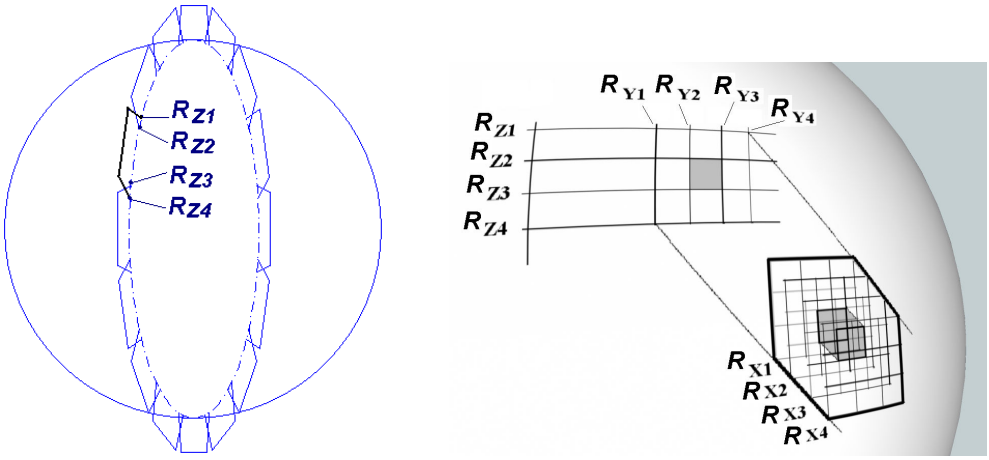


Figure 3.16: From the Fuzzy Qualitative Circle To Fuzzy Qualitative Euler Angles Positions

described by the polygon $P1_{ZYX}$ with vertices:

$$(R_{X2} \circ R_{Y2} \circ R_{Z2}, R_{X2} \circ R_{Y3} \circ R_{Z2}, R_{X2} \circ R_{Y2} \circ R_{Z3},$$
$$R_{X2} \circ R_{Y3} \circ R_{Z3}, R_{X3} \circ R_{Y2} \circ R_{Z2}, R_{X3} \circ R_{Y3} \circ R_{Z2},$$
$$R_{X3} \circ R_{Y2} \circ R_{Z3}, R_{X3} \circ R_{Y3} \circ R_{Z3})$$

Secondly, the area of membership smaller than 1. After rotation R_Z, it is defined by the segment $[R_{Z1}, R_{Z4}] - [R_{Z2}, R_{Z3}]$. The area of membership smaller than 1 after rotation R_Z and R_Y is described by the difference of polygons defined with vertices $(R_{Y1} \circ R_{Z1}, R_{Y1} \circ R_{Z4}, R_{Y4} \circ R_{Z1}, R_{Y4} \circ R_{Z4}) - P1_{ZY}$. The area of membership smaller than 1 after rotation R_X R_Z and R_Y is : the area described by difference of polygons with vertices:

$$(R_{X1} \circ R_{Y1} \circ R_{Z1}, R_{X1} \circ R_{Y1} \circ R_{Z4}, R_{X1} \circ R_{Y4} \circ R_{Z1},$$
$$R_{X1} \circ R_{Y4} \circ R_{Z4}, R_{X4} \circ R_{Y1} \circ R_{Z1}, R_{X4} \circ R_{Y1} \circ R_{Z4},$$
$$R_{X4} \circ R_{Y4} \circ R_{Z1}, R_{X4} \circ R_{Y4} \circ R_{Z4}) - P1_{ZYX}$$

A mapping of all possible Qualitative Euler transformations to surfaces defined by an Icosahedron based polygon is then done (see figure 3.17). A unit sphere is tessellated into triangle of similar areas using a regular Icosahedron where each initial triangle is divided into 3 equilateral sub-triangles. As the sphere is divided into 60

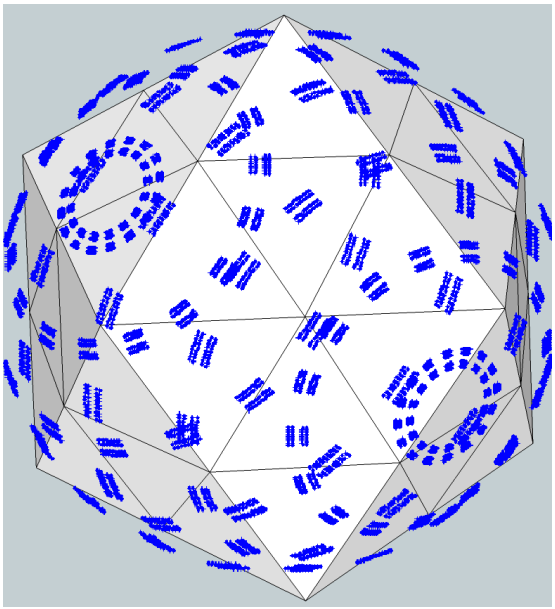


Figure 3.17: Mapping vertices of all possible Qualitative Euler transformations to triangles of an Icosahedron-based polygon

equal areas, there should be a mapping of minimum roughly $3360/60 = 56$ Euler combinations per triangular area (assuming one triangle covers in average one fuzzy qualitative surface). In practice, one triangle covers an average of 3 qualitative surfaces, so $56 \cdot 3 = 168$ Euler combinations can be expected per triangle.

When dealing with insufficient data, one generally has to first make the best of the precious little information available in order to then venture into making educated guesses. This idea is translated by implementing the ranking strategy in two steps. First, an initial evaluation is conducted from the available (or non-occluded) data. An overall average of the membership scores of visible joints to Fuzzy Membership Functions that model the rotations of corresponding joints in known stances is produced. Let $j = 1, .., 19$ be one of the 19 joints and M_i with $i = 1, .., 6$ be one of the six possible moves: Guard, Jab, Cross, Left Hook, Right Hook, and Lower Left Hook. Let $S_{j_{M_i}}$ be the membership score of a non-occluded joint j to a trapezoid Fuzzy Membership Function that models the rotation of that joint for a known move M_i. We end up with a set of six average membership scores: $\langle S_{M_1}, S_{M_2}, S_{M_3}, S_{M_4}, S_{M_5}, S_{M_6} \rangle$. If there is very little occlusion, the information from visible joints conveys more certainty. Inversely, if there are many occluded joints, the information from these visible

joints conveys a much greater uncertainty. If most of the joints (around 80%) are non-occluded, then the average membership scores of the visible joints is the most important information, while the plausible rotations estimated for the occluded joints is just here to give possible interpretations. In the inverse situation where 80% of joints would be occluded, the plausible estimations of rotations would have more weight on the classification. In this experiment, as around 80% the joints are visible at any given time, the move with the highest average membership scores from non-occluded joints tends to reflect the most likely estimation. Therefore, the best move from visible data satisfies the constraint: $M_{V_{1st}} = \max \langle S_{M_1}, S_{M_2}, S_{M_3}, S_{M_4}, S_{M_5}, S_{M_6} \rangle$. In this case, solutions generated by plausible joints are simply here to confirm or infirm this initial assessment. In the second step, for every occluded joint, plausible fuzzy qualitative Euler Angles are ranked by geometrical distance to the Fuzzy Membership Functions of known moves. The three closest moves are selected and ranked as possible solutions: $\langle M_{O_{1st}}, M_{O_{2nd}}, M_{O_{3rd}} \rangle$. If one of the most plausible solutions from occluded joints corresponds to the move estimated as most likely from visible data, the system confirms that the occluded frame can be classified as the latter. In other words, if: $M_{V_{1st}} \in \langle M_{O_{1st}}, M_{O_{2nd}}, M_{O_{3rd}} \rangle$ then $M_{V_{1st}}$ is considered the most plausible suggestion for the occluded frame. A case study using an occluded data sample is conducted as a proof of concept in section 4.4.2.

3.5 Summary

This thesis presents a novel inference engine that aims at classifying occluded 3d human motion assisted by the recognition context. First, uncertainties are wrapped into a fuzzy membership function via a novel method called fuzzy quantile generation which employs metrics derived from the probabilistic quantile function. Then, time-dependent and context-aware rules are produced via genetic programming to smooth the qualitative outputs represented by fuzzy membership functions. Finally, occlusion in motion recognition is taken care of by introducing new procedures for feature selection and feature reconstruction. The three components of this framework therefore address the problems stated in section 1.1, i.e. the recognition of actions independently from differences in execution in the spatial and temporal domains with learning samples of sub-optimal size, the generation and application of prior knowledge in a context-aware and time-sensitive fashion, and the need to deal with insufficient data.

Chapter 4

Experiments and results

4.1 Introduction

The aim of this chapter is to evaluate the framework presented in chapter 3 by putting to the test its different components with experiments and results. The challenges posed by the nature of the motion capture boxing dataset are, among others: biologically "noisy" data, cross-gait differentials from one individual to another, and high dimensionality caused by the complexity of the skeletal representation (57 degrees of freedom). It is assumed that being successful at the non-trivial exercise of classification of such complex data might give the presented techniques stronger credentials as a contender in the field of motion recognition. The motion recognition framework is therefore put to the test in an experiment involving the classification of real natural 3d motion capture data in the context of boxing. Boxing motions present the advantage of being well defined and involve a challenging degree of precision for spatio-temporal recognition. They are also relatively static and can therefore accommodate a motion capture studio of reduced surface. Finally, they can easily be reused in application domains linked to security surveillance. On the other hand, 3d boxing motion capture data can be quite challenging regarding the experimental design due to potential problems such as possible physical or psychological changes taking place through time, subjects with abnormal gaits, and instrumentation inconsistencies. In the first part of this section, the experimental method and setup are described. Secondly, the performance of FQG as a standalone learning paradigm applicable to behaviour recognition

is presented. The effectiveness of this method is demonstrated through the classification of motion capture data from real boxers and a comparative study. Thirdly, experimental results of the context-aware GP filter show that the filter consistently improves the accuracy of the FQG classifier. Finally, an evaluation of the feature selection and feature reconstruction aspects of the occlusion module shows that a) the Laplace fuzzy similarity relation outperforms other measures consistently, and b) the feature reconstruction mechanism allows the system to estimate correctly a significant portion the initially intractable occluded data present in the boxing experiment.

4.2 Experimental setup: recognizing boxing motion captures

The description of the experiment starts with a review of the equipment and apparatus. Then, an examination of the modalities in use for the selection of participants follows. Finally, the methods and procedures employed in this experiment are detailed step by step.

4.2.1 Apparatus

The motion capture data are obtained from a Vicon Motion Capture Studio with eight infra-red cameras. Each motion capture suits is set with a total of 49 optical passive reflective markers and recordings are sampled at the speed of 120 frames per second. The motion recognition is implemented in MATLAB on a standard PC with 2 Gigs of RAM. An additional MATLAB toolbox presented in Lawrence (2007) is also used for extracting Euler Angles from BVH files. The relative positions of the cameras and the subject are shown in figure 4.1.

Three male subjects, aged between 18 and 21, of light to medium-average size (167cm to 178cm) and weight (59 to 79kgs), all practicing boxing in competition at the national level. None of them presented any abnormal gait. Optical Markers were placed in a similar way on each subject to ensure a consistent motion capture (see Figure 4.2 and 4.3).

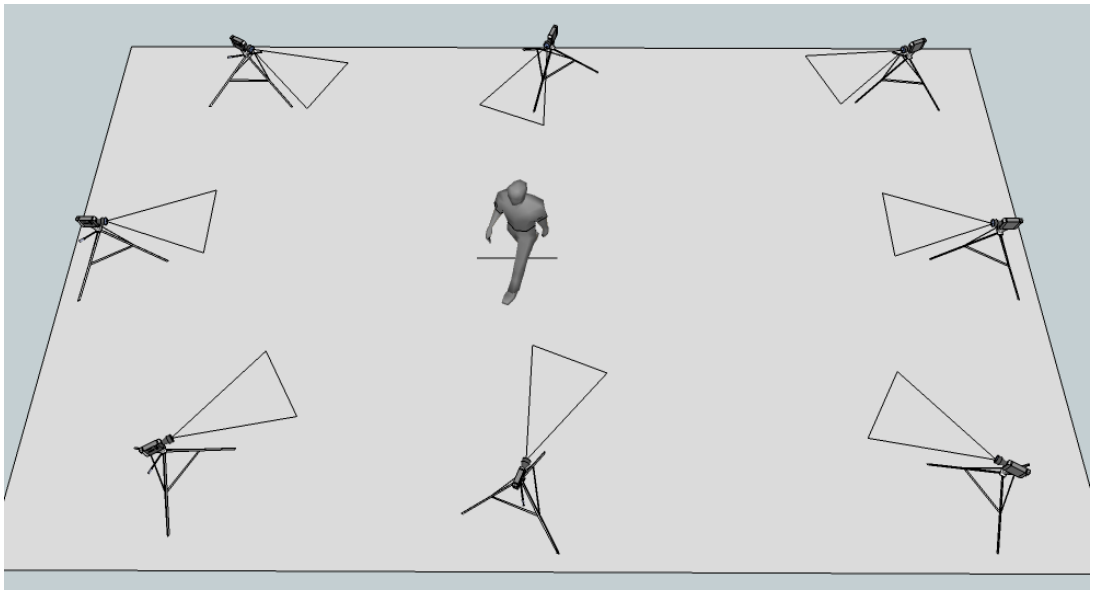


Figure 4.1: 3D motion capture studio - floor plan

4.2.2 Procedure

Participants selected at the national championship level are asked to perform modern boxing moves on their own, in a preordained and controlled fashion. The motion capture data is obtained from several subjects performing each boxing combination four times. There are twenty-one different boxing combinations, each separated by a guard stance. These are performed at two different speeds (medium-slow and medium fast). The boxing combinations are ordered sets of basic boxing stances. There are in total nine precisely defined basic stances: Guard, Jab, Lower Jab, Cross, Lower Cross, Left Hook, Right Hook, Lower Left Hook, Right Uppercut. The level of precision needed to identify such motions is non-negligible. These are well-known boxing stances and are accurately described for a right handed boxer - they should be reversed for a left-handed one - in Cokes (1980) as follows:

- Guard : a defensive position where the boxer stands with the legs shoulder-width apart and the hands in front in order to protect the head from incoming punches.
- Jab : a quick, straight punch thrown with the lead hand from the guard position. The jab is accompanied by a small, clockwise rotation of the torso and hips, while the fist rotates 90 degrees, becoming horizontal upon impact.

Figure 4.2: Optical markers placement - front view

Figure 4.3: Optical markers placement - back view

- Lower Jab : similar to a jab in a crouching stance.
- Cross : a powerful, straight punch thrown with the rear hand. For additional power, the torso and hips are rotated counter-clockwise as the cross is thrown.
- Lower Cross : similar to a cross in a crouching stance.
- Right Hook : a semi-circular punch thrown with the lead hand to the side of the opponent's head. The torso and hips are rotated clockwise. The lead foot pivots clockwise too, turning the left heel outwards.
- Left Hook : similar to a Right Hook, but done with the rear hand.
- Lower Left Hook : similar to a Left Hook in a crouching stance.
- Right Uppercut : a vertical, rising punch thrown with the rear hand.

The table 4.1 gives an overview of the different combinations of boxing moves recorded in the same order by all participants. These sequences involve complex movements fusing into each other. After cleaning the data and eliminating faulty execution of Lower Jab, Lower Right Hook and Right Uppercut in some instances, combinations 5, 8, 9, 18, 19 and 20 were discarded, and the following six moves were retained: Guard, Jab, Cross, Left Hook, Right Hook, Lower Left Hook. A leaflet and an informed consent form for participants approved by Portsmouth University Ethics Review Board are attached in appendix D. As explained in these documents, the participant goes through the following steps:

- on arrival, after a 10 minutes long introduction, the participant is briefed on the experiment and given consent and information forms to read and sign.
- putting the suit and captors on : 10 min
- set up and calibration : 1 hour
- data capture of different motions spread over 20 minutes.
- changing and leaving.

Table 4.1: Combinations of boxing moves recorded (repeated 4 times each)

Comb	Detail of moves
Comb 1	(Guard - Jab)$\times 4$ Guard
Comb 2	(Guard - Cross)$\times 4$ Guard
Comb 3	(Guard - Lower Cross)$\times 4$ Guard
Comb 4	(Guard - Right Hook)$\times 4$ Guard
Comb 5	(Guard - Lower Right Hook)$\times 4$ Guard
Comb 6	(Guard - Left Hook)$\times 4$ Guard
Comb 7	(Guard - Lower Left Hook)$\times 4$ Guard
Comb 8	(Guard - Right Uppercut)$\times 4$ Guard
Comb 9	(Guard - Left Uppercut)$\times 4$ Guard
Comb 10	(Guard - Jab - Cross)$\times 4$ Guard
Comb 11	(Guard - Left Hook - Cross)$\times 4$ Guard
Comb 12	(Guard - Lower Jab - Cross)$\times 4$ Guard
Comb 13	(Guard - Jab - Right Hook)$\times 4$ Guard
Comb 14	(Guard - Jab - Cross - Left Hook)$\times 4$ Guard
Comb 15	(Guard - Jab - Cross - Lower Left Hook)$\times 4$ Guard
Comb 16	(Guard - Cross - Jab - Lower Cross)$\times 4$ Guard
Comb 17	(Guard - Cross - Left Hook - Right Uppercut)$\times 4$ Guard
Comb 18	(Guard - Jab - Cross - Left Uppercut - Right Uppercut)$\times 4$ G.
Comb 19	(Guard - Jab - Right Uppercut - Left Hook - Right Hook)$\times 4$ G.
Comb 20	(Guard - Cross - Left Hook - Right Uppercut - Left Hook)$\times 4$ G.

The main validity threats are maturation (during the experimental period, physical or psychological change takes place), extraneous variables (i.e. subjects with abnormal gaits) and instrumentation inconsistency (possible occlusion problems or sensing errors during motion capture). Regarding the problem of maturation, every motion is repeated four times, and every participant can rest in between each combination in order to minimize fatigue. Participants are selected at the national championship level. They all have a high stamina minimizing disturbances due to fatigue. To reduce extraneous variables, subjects with abnormal gaits are discarded, and the sample is relatively homogeneous (male subjects of similar age and average build). Optical Markers are placed in a similar way on each subject to ensure a consistent motion capture. Instrumentation and representation inconsistency are reduced by using motion-capture post-processing software. The Vicon software package is used to "clean" the data.

A set of fuzzy membership functions corresponding to a specific stance is extracted from various samples. First, all three participants are employed to learn and to test how well the system recognizes the stances. Then, an evaluation is conducted to see how the system copes to learn from only two participants, and test how well it recognizes stances from a third different participant. Analysis is initially focused on classifying only one move and then to six moves at the same time. The inputs for each given time frame are the six membership scores of each known move. These membership scores are re-scaled as indicated in section 3.

4.2.3 Fuzzy quantile generation classifier evaluation

An initial review of FQG performance as a standalone classifier is conducted using Receiver Operating Characteristic (ROC) analysis. It is followed by a more complex comparative exercise between FQG and sixteen other machine learning techniques involving the recognition of multiple moves from the same 3d motion capture boxing data set.

4.2.4 Initial analysis of accuracy using receiver operating characteristic

The classifier is evaluated by comparing its performance to a human observer. One expert identifies "Guard" frames of membership 1 and of membership 0 (non-guard frames). The number of false positives (frames identified by the expert as non-guards, but identified by the classifiers as guards) and false negatives (frames identified by an expert as guards, but classified as non-guards by the system) are taken into account. ROC analysis is used to plot the true positive rates versus the false positive rates as a function of different membership thresholds. The data are partitioned into sub-samples and tests are run using K-fold cross-validation. The k results from the folds are averaged to produce a single estimation. We present results for a 3-fold cross-validation where one third of the data is used for learning while the rest is used for testing as shown in figure 4.4. The testing samples represent about 107000 unidentified frames. In one evaluation case, all participants are used for learning and testing. This means there is a greater similarity between the learning and the test samples, as the gait differences are reduced. In a second situation, two participants are used for learning and a third different one for testing. This poses a greater challenge as there are stronger gait differences.

As seen in the ROC figure 4.4, the optimum accuracy of the classifier is 0.95 if the same participants are used for learning and testing, or 0.88 when different participants are used for learning and testing. Crisp evaluation (the accuracy obtained for detecting frames of "Guard" membership only equal to 1.0) gives inferior results: 0.906 in the first case and 0.506 in the second case. This gain is especially noticeable when the learning and the testing data present less similarity. Accuracy is defined as:

$$accuracy = \frac{tp + tn}{tp + fp + fn + tn} \tag{4.1}$$

where tp represents true positive rate, tn true negative rate, fp false positive rate, and fn false negative rate.

The system can recognize a Guard stance with an average accuracy of 88.68% when using half of the data for learning and the other half for testing, on all three participants. Some of these movements have very few learning examples available. In

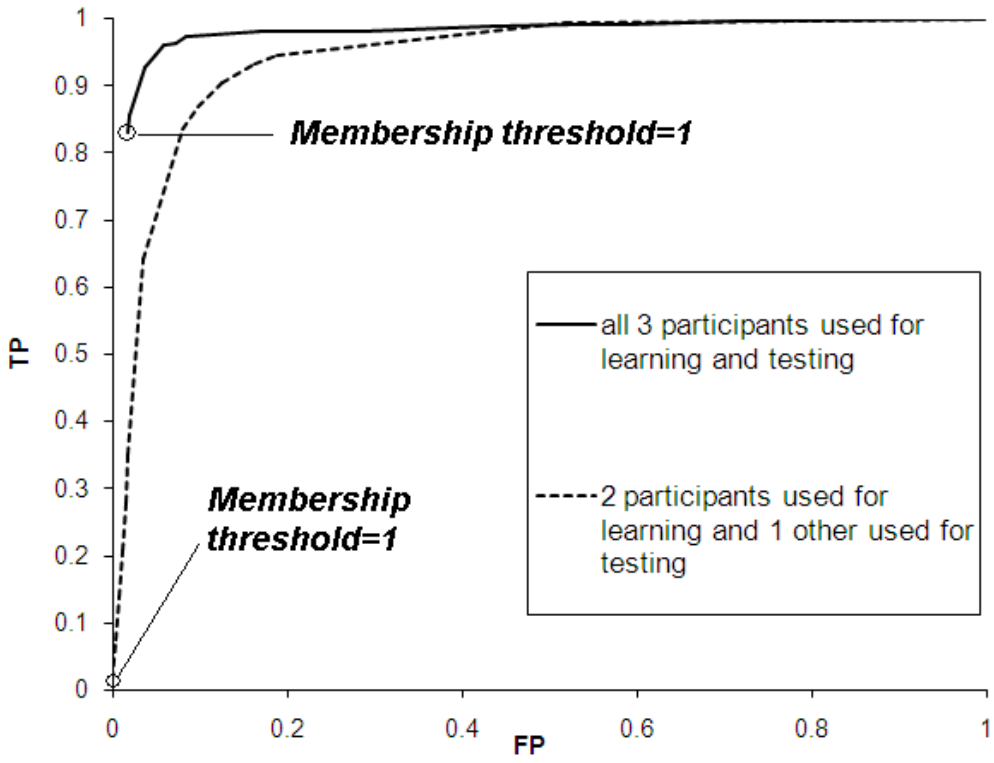


Figure 4.4: 3-Fold Cross-Validation ROC Analysis of the Guard Classifier

this case, the threshold is fine-tuned by decreasing it to compensate for the data sparsity. It is worth noticing that this system is not a binary but a fuzzy classifier. The threshold value will therefore stay between 0 and 1, which might give the illusion of an "unfinished" ROC curve if the learning and test samples are similar enough (see Figure 4.4 where the membership-one point start with a high True Positive rate because we learn from and test with the same boxers). The ROC curves show that, the fuzzy classifier performs better than its crisp counterpart (the one that only identifies Guards of membership threshold equal to 1). It has been observed that a high threshold value is needed to obtain good results. If the threshold is inferior to a membership degree of 0.8, we obtain a maximum True Positive Rate (most of known guards are correctly identified) and a minimum false Negative rate (nearly all known non-guard are identified as guards).

The time complexity for recognizing n stances is of the order $O(n)$. It takes in average 16.05 ms on laptop running non-optimized Matlab code to create a template

and evaluate a frame membership score for one stance. This system could potentially be optimized and implemented for real-time motion recognition applications. These initial results (appendix F[6,8]) lead to a comparative work between FQG and other techniques.

4.2.5 Comparing fuzzy quantile generation with other classifiers

Initial comparison was started with a time-dependent Hidden Markov Model (HMM) classifier. However, due to the particularities of the boxing data set, artificially created classes representing boxing stances could not be obtained via standard clustering methods and would otherwise give singular matrices with HMM. Another fact was that HMM seemed to need considerably larger learning samples to be able to perform. It was then decided to focus the study on a comparison between FQG and sixteen time-invariant classification algorithms aiming at recognising boxing stances. These classifiers take exactly the same input data as FQG in order to provide a fair comparison. All these algorithms are implemented using the WEKA Machine Learning package presented in Frank *et al.* (2005) and Hall *et al.* (2009). They represent diverse paradigms that can be roughly classified into seven types: Bayes, Function based, Nearest Neighbours, Tree based, Rule based, Neural Networks, and Miscellaneous (see detailed classification in appendix B). Several of these techniques correspond to some of the methods used for motion recognition presented in section 2.3. Bayes Net and Naive Bayes fit into the Probabilistic graphical models category. Similarly, Radial Basis Function networks and Multilayer Perceptrons are part of the connectionist approach. SMO can be classified as a kernel based method, while Hyperpipes and Voting Feature Intervals are part of the voting strategies paradigm. IB1 is an instance based classifier, and Fuzzy Nearest Neighbours and Fuzzy Rough Nearest Neighbours can fit into hybrid methods. The WEKA classifiers are used with parameters based on default settings. In order to keep the comparison relatively fair, FQG parameter s described in 3.2.1.2 was set by default to 0.7 through the whole comparative experiment. While some of these classifiers might perform substantially better with deeply optimised parameters, the purpose of this study, is not to focus on their optimization but is rather to give a comparison ground with FQG and commonly-used techniques representing

Table 4.2: Common terms used in binary classification

		Condition		
		True	False	
Test Outcome	Positive	TP	FP	Precision
	Negative	TN	FN	Negative Predictive Value
		Sensitivity or Recall	Specificity	Accuracy

different paradigms on the same boxing 3d Motion Capture data sample. These algorithms display high computational efficiency and are used in state of the art research. It is difficult to generalise comparisons between machine learning methods due to the particulars of the topology of the search space of each application domain. However, it is possible to offer a comprehensive and indicative study (appendix F[3]) that does not necessarily have to deal in absolutes to say if a method performs roughly at the same level of efficiency as the best methods available.

4.2.6 Results of comparative study

Classifiers performances can be evaluated in different ways. Accuracy, precision, negative predictive value, recall, specificity, F-Score, and Matthews correlation coefficient will be used to allow a detailed analysis on the performance of FQG. Each measure is formally defined and its results discussed. Table 4.2 gives a quick overview of mainstream measures detailing certain aspect of a classifier performance.

These measures, as well as F-score and Matthews correlation coefficient are defined and applied to compare FQG performance with sixteen other classifiers.

4.2.6.1 Accuracy

Accuracy is mathematically defined in equation 4.1 and can be defined as the proportion of true results (both true positives and true negatives) in the population. It is the most used empirical measure in the area of machine learning classifiers. at a first

glance (see figure 4.5), results seem quite encouraging as FQG shows the best average accuracy of 90% over sixteen high performance machine learning classifying methods (figure 4.5). This difference in accuracy with FQG is significant at the level 0.05 on a two-tailed t-test for fifteen of these methods at the exception of Bayes Net (86%) which performs similarly and does not show a significant difference, even at the level 0.20. So far, if exclusively focusing on the accuracy, FQG and Bayes Net seem to

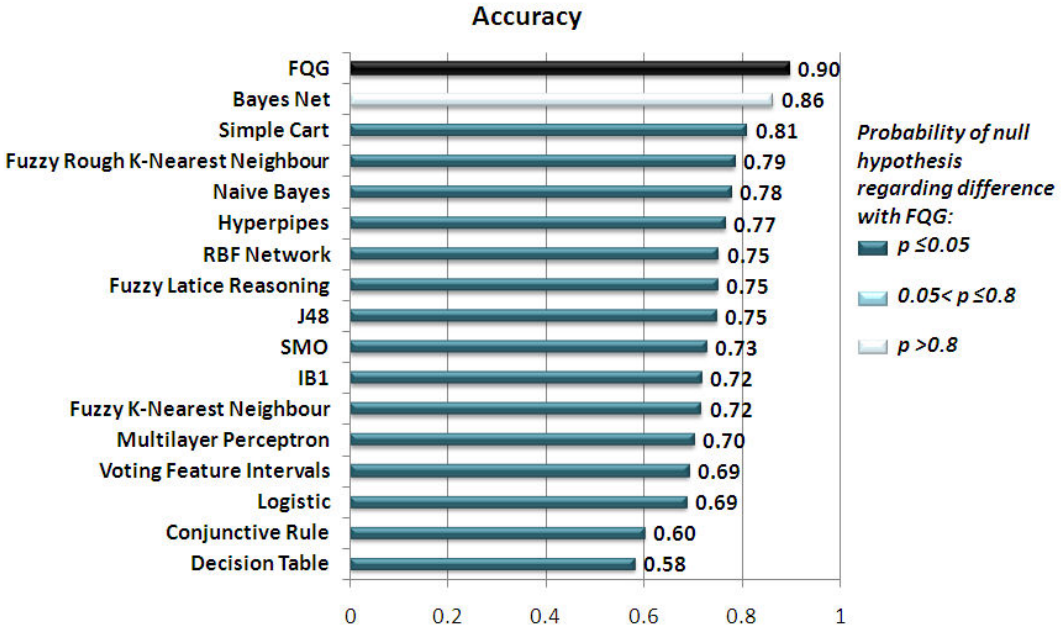


Figure 4.5: Comparison of accuracy between FQG and 14 other classifiers

be in first position, while the next three best methods are: SimpleCart(81%), Fuzzy Rough K-Nearest neighbours (79%) and Naive Bayes (78%). Beside FQG, the most accurate methods belong broadly to Bayes, decision trees and Fuzzy Rough Nearest Neighbours types. The worst performances belong to rule based classifiers - Conjunctive Rule (60%) and Decision Table (58%). They present a loss of accuracy of nearly 10% compared to the next best classifier. This gap could be explained by the difficulty for the bottom-up design to generate the sheer quantity of rules that could deal with real numbers in a noisy and biologically imprecise data set. An interesting fact is that a decision tree based method like minimal cost-complexity pruning performs so well (third position) and so much better than its direct "relative" J48 (9th position

with 75% accuracy). If both algorithms are using a relatively successful top-down approach, the CART algorithm is using Gini impurity, while, on the other side J48 is using Information Gain. In this experimental context the former seems to perform better than the latter. Similarly, there seem to be a significant difference of performance between Fuzzy K-Nearest Neighbour and Fuzzy Rough K-Nearest Neighbour (at the level 0.05). The former seems to have a lower accuracy than the latter, as it does not benefit from the added power of fuzzy rough representation. The difference between Bayes and Naive Bayes is significant at the level 0.05. There also seem to be a difference of accuracy between RBF Network (75%) and Multilayer Perceptron(70%), but a t-test between the two does not give it much significance. When using accuracy alone, results seem to be in favour of FQG and Bayes Net. However, using only accuracy to measure the performance of the FQG classifier would only show a part of the whole picture. The accuracy paradox for predictive analytics states that predictive models with a given level of accuracy may have greater predictive power than models with higher accuracy. It may be better to avoid the accuracy metric in favour of other metrics such as precision and recall. Sokolova *et al.* (2006) also underline the fact that, for example, accuracy, does not distinguish between the numbers of correct labels of different classes while other measures such as sensitivity and specificity do. Further analysis could tell which one of the two performs the best, and might explain why these types of classifiers perform better on this type of data set.

4.2.6.2 Precision

Precision or Positive Predictive Value (see equation 4.2)is the proportion of the true positives against all the positive results (both true positives and false positives). i.e. In the case of determining which patients are afflicted by a disease, it could be equated to the proportion of patients with positive test results who are correctly diagnosed.

$$Precision = \frac{TruePositives}{TruePositives + FalsePositives} \tag{4.2}$$

Regarding the average Precision or Positive Predictive Value (see figure 4.6), FQG seems to be performing poorly (11th position for 90%). On the other hand, Bayes Net (98%), SimpleCart (97%), Fuzzy Rough K-Nearest neighbours (95%) and Naive Bayes (94%) stay in top position. The difference of average precision with FQG is

significant at the level 0.05 on a two-tailed t-test for all these methods. One noticeable fact is that there seem to be a plateau at (90%) where no less than four different methods (FQG, IB1, SMO, and Fuzzy K-Nearest Neighbour share) share the same average precision. The difference of performance between Fuzzy K-Nearest Neighbour and Fuzzy Rough K-Nearest Neighbour is significant (at the level 0.05). The differences between respectively CART and J48, Bayes and Naive Bayes, and RBF Network and Multilayer Perceptron are less significant (only at the level 0.2). The worst performances still belong to rule based classifiers - Conjunctive Rule (86%) and Decision Table (72%). This measure of exactness or fidelity indicates that every result retrieved

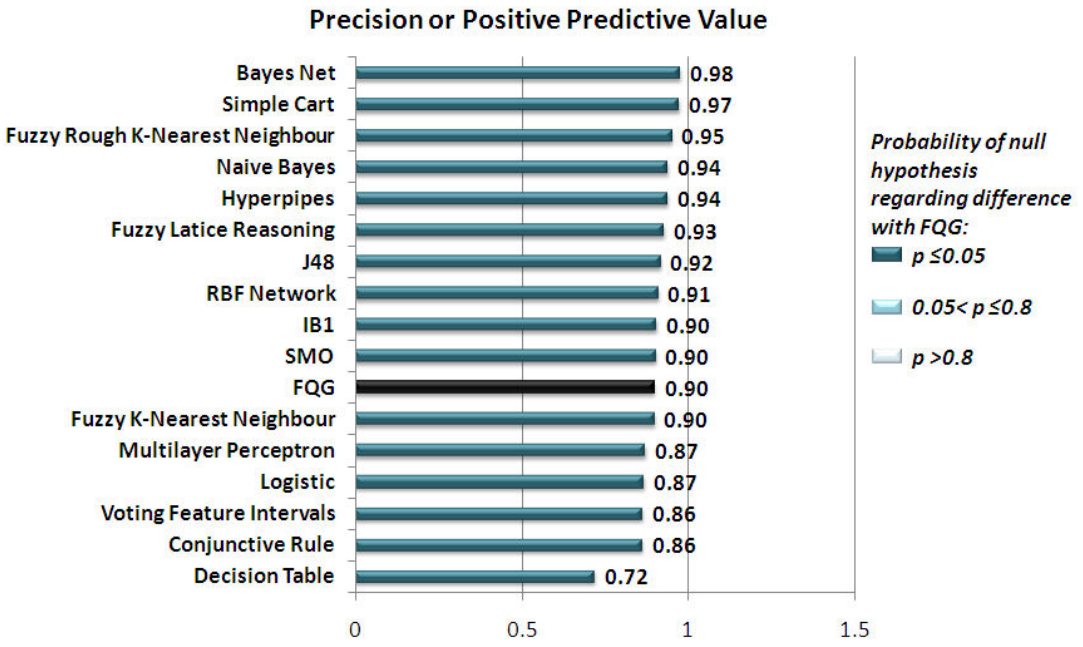


Figure 4.6: Comparison of precision between FQG and 14 other classifiers

by a FQG search was relevant with a comparatively low probability of 90%, but says nothing about completeness (whether all relevant instances were retrieved). This is why an analysis of Recall is required.

4.2.6.3 Recall

Recall or sensitivity or True Positive Rate (equation 4.3) measures the proportion of actual positives which are correctly identified as such e.g. The percentage of sick

people who are identified as having the condition. It is closely related to the concept of type I error concerning false positives.

$$Recall = \frac{TruePositives}{TruePositives + FalseNegatives} \tag{4.3}$$

FQG shows the best average recall of 89% over the sixteen other classifiers (figure 4.5). This difference in recall with FQG is significant at the level 0.05 on a two-tailed t-test for all other methods. Bayes Net (75%), SimpleCart (64%), Fuzzy Rough K-Nearest neighbours (60%) and Naive Bayes (59%) are still the next best classifiers. There is a noticeable gap of 14% between FQG and the second best classifier. This would indicate that the strong point of FQG is completeness. The difference of performance between Fuzzy K-Nearest Neighbour and Fuzzy Rough K-Nearest Neighbour is significant (at the level 0.05). The differences between respectively CART and J48, and RBF Network and Multilayer Perceptron are not tested as significant. The difference between Bayes and Naive Bayes is significant at the level 0.06. As usual, on this dataset, the worst performances belong to rule based classifiers - Conjunctive Rule (27%) and Decision Table (24%).

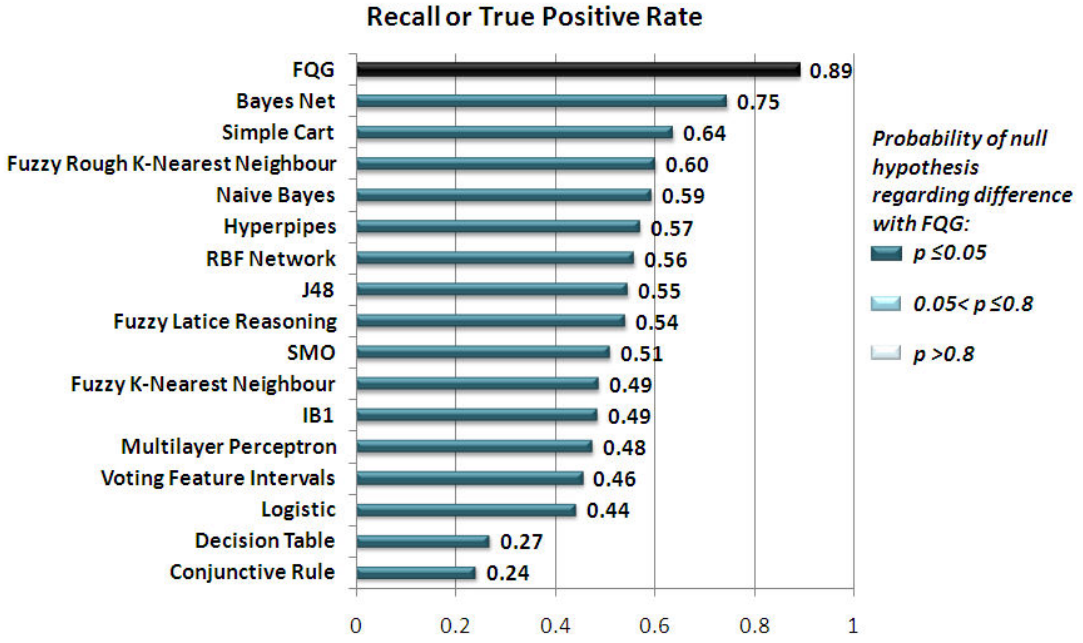


Figure 4.7: Comparison of recall between FQG and 14 other classifiers

As for accuracy and precision, CART(3rd position with 64% accuracy) has a much better recall than its direct "relative" J48 (8th position with 55% accuracy). Having examined precision and recall, it might be interesting to look at a weighted average of these measure, that is to say the F-score.

4.2.6.4 F-measure

F-Measure or balanced F-Score(equation 4.4) is a measure that combines Precision and Recall. It is the harmonic mean of precision and recall.

$$F - score = \frac{Precision \cdot Recall}{Precision + Recall} \tag{4.4}$$

FQG shows the best average F-score of 90% over the sixteen other classifiers (figure 4.5). This difference in F-score with FQG is significant at the level 0.05 on a two-tailed

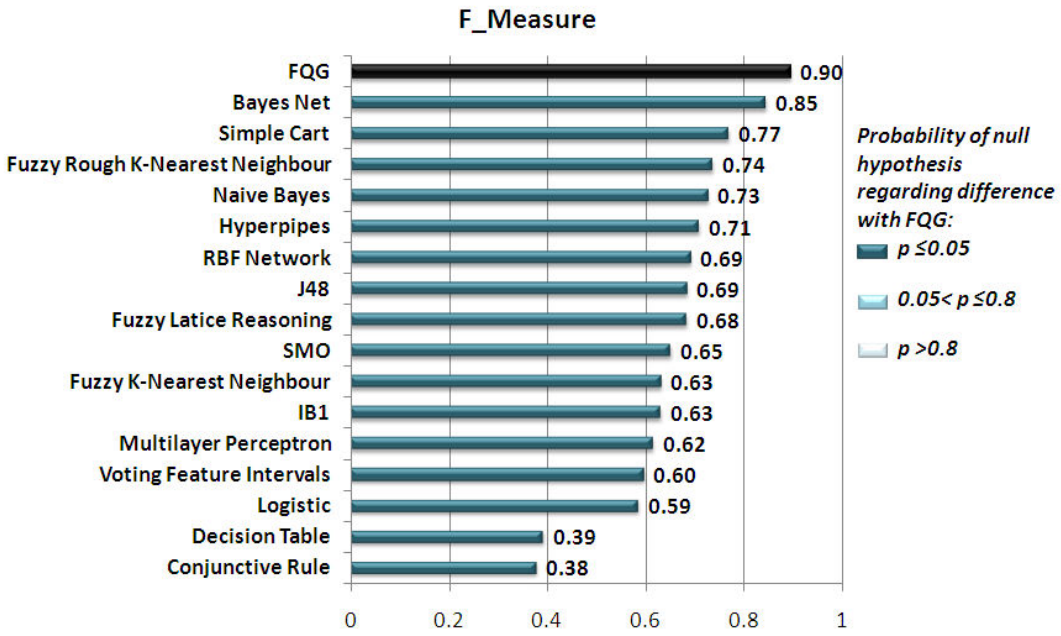


Figure 4.8: Comparison of F-measure between FQG and 14 other classifiers

t-test for all of these methods. The next four best classifiers are Bayes Net (85%), SimpleCart (77%), Fuzzy Rough K-Nearest neighbours (74%) and Naive Bayes (73%). These results confirm the overall dominance of FQG so far. The differences of performance between Fuzzy K-Nearest Neighbour and Fuzzy Rough K-Nearest Neighbour

are significant at the level 0.05. The differences between Bayes and Naive Bayes and between CART and J48 are only significant at the level 0.2. The difference between RBF Network and Multilayer Perceptron is not significant. Previous results suggested that the lowest performance of FQG was in estimation of the Positive Predictive Value. Therefore, it seems like a sensible course of action to expect a higher success on the Negative Predictive value.

4.2.6.5 Negative predictive value

The negative predictive value (equation 4.5) is the proportion of instances correctly described as not part of a class.

$$NegativePredictiveValue = \frac{TrueNegatives}{TrueNegatives + FalseNegatives} \tag{4.5}$$

FQG shows the best average F-score of 89% over the sixteen other classifiers (figure 4.5). This difference in NPV with FQG is significant at the level 0.05 on a two-tailed

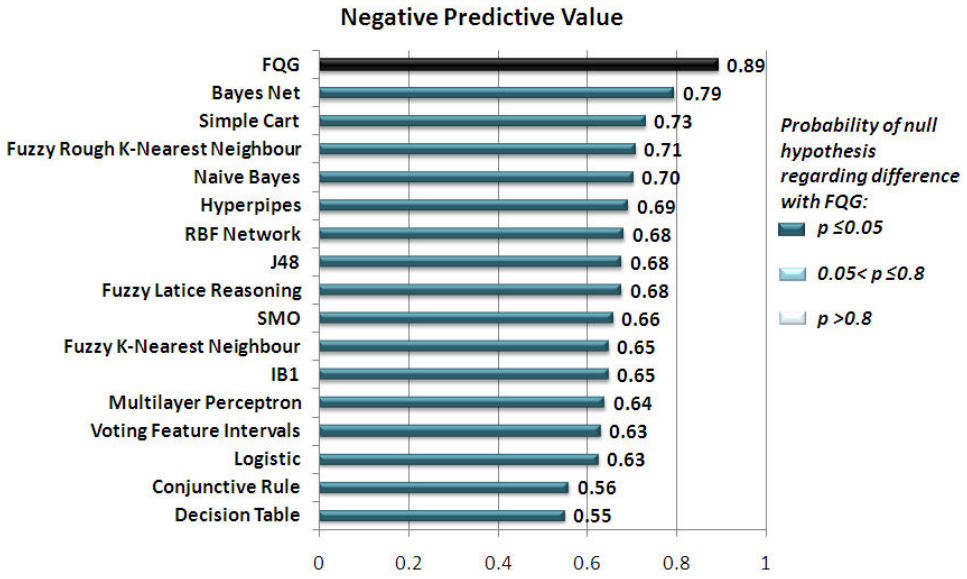


Figure 4.9: Comparison of NPV between FQG and 14 other classifiers

t-test for all of these methods. The next four best classifiers are Bayes Net (79%), SimpleCart (73%), Fuzzy Rough K-Nearest neighbours (71%) and Naive Bayes (70%). The difference of performance between Fuzzy K-Nearest Neighbour and Fuzzy Rough

K-Nearest Neighbour is significant at the level 0.05. The differences between CART and J48, and between Bayes and Naive Bayes are less significant($p = 0.2$). The difference between RBF Network and Multilayer Perceptron is not significant.

4.2.6.6 Specificity

Specificity or True Negative Rate(equation 4.6) measures the proportion of negatives which are correctly identified (e.g. the percentage of healthy people who are identified as not having the condition). It is closely related to the concept of type II error concerning false negatives.

$$Specificity = \frac{TrueNegatives}{FalsePositives + TrueNegatives} \tag{4.6}$$

FQG seems to have the second lowest average specificity (90%) (figure 4.5). This

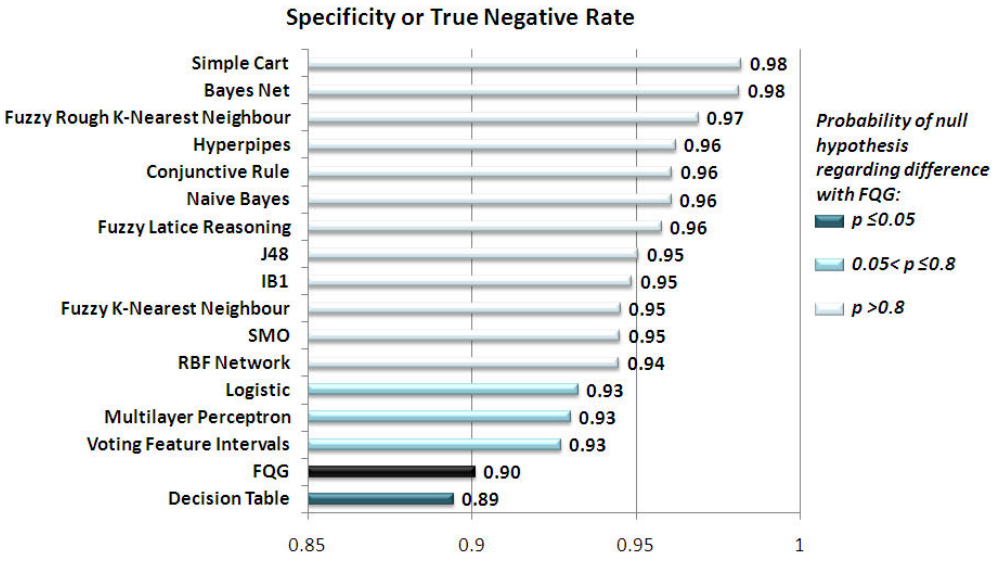


Figure 4.10: Comparison of specificity between FQG and 14 other classifiers

difference in specificity with FQG is significant at the level 0.05 on a two-tailed t-test with the only other worst method: Decision Table (89%). The difference with the next three better classifiers: Logistic (93%), Multilayer Perceptron (93%) and Voting Feature Interval (93%) is significant at the level 0.2. The difference between FQG and all remaining classifiers is not significant at the 0.2 level. The five best classifiers are

SimpleCart (98%), Bayes Net(98%), Fuzzy Rough K-Nearest neighbours (97%) Hyperpipes (96%), and Conjunctive Rule (96%). The differences of performance between Fuzzy K-Nearest Neighbour and Fuzzy Rough K-Nearest Neighbour , between CART and J48, between RBF Network and Multilayer Perceptron, and between Bayes and Naive Bayes are not significant. It is quite interesting to note that the results seem to conform less than to the other measures. Hyperpipes gets the 4th best result, showing that this method might possibly have a rather good specificity. Also, Conjunctive Rule, unlike Decision Table which is last as usual, finds itself in 5th best position. However, due to the low level of confidence of most measurements concerning specificity, these observations are more speculative than informative. To give a final overview of FQG performance, the Matthews Correlation Coefficient is used.

4.2.6.7 Matthews correlation coefficient

The Matthews correlation coefficient (equation 4.7) is used in machine learning as a measure of the quality of binary (two-class) classifications. It takes into account true and false positives and negatives and is generally regarded as a balanced measure which can be used even if the classes are of very different sizes. The MCC is in essence a correlation coefficient between the observed and predicted binary classifications; it returns a value between -1 and +1. A coefficient of +1 represents a perfect prediction, 0 an average random prediction and -1 an inverse prediction. The statistic is also known as the phi coefficient.

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{4.7}$$

FQG shows the best average MCC of 79% over the sixteen other classifiers (figure 4.5). This difference in MCC with FQG is significant at the level 0.05 on a two-tailed t-test for all of these methods at the exception of Bayes Net which does not show a significant difference, even at the level 0.20. The next four best classifiers are Bayes Net (75%), SimpleCart (66%), Fuzzy Rough K-Nearest neighbours (61%) and Naive Bayes (60%). The differences of performance between Fuzzy K-Nearest Neighbour and Fuzzy Rough K-Nearest Neighbour and between Bayes and Naive Bayes are significant at the level 0.05. The difference between CART and J48 is less significant ($p = 0.2$). The difference between RBF Network and Multilayer Perceptron is not

significant. As in the previous measures, the worst performances belong to rule based classifiers - Conjunctive Rule (29%) and Decision Table (21%).

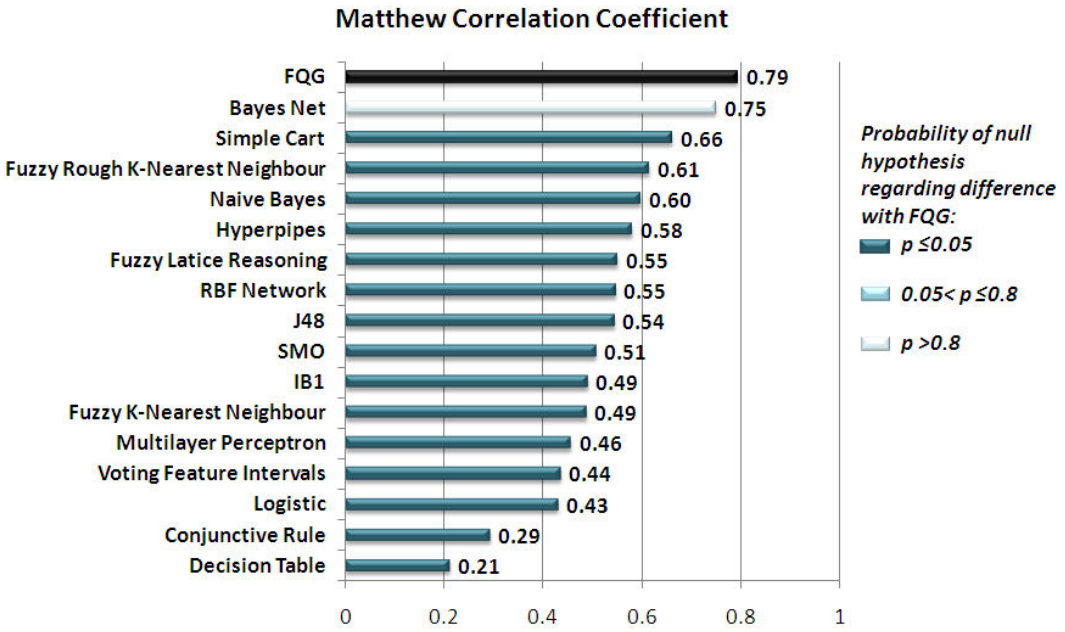


Figure 4.11: Comparison of MCC between FQG and 14 other classifiers

Regarding present results, a careful examination in light of these different measures will help to uncover and explain some of FQG strong points and weaknesses compared to other classifiers.

4.2.7 Discussion

The linear time and space complexity of FQG is comparable to those of VFI and Hyperpipes while most of the other methods presented here tend to exhibit a time and space complexity of polynomial order at best, which put FQG in an advantageous position. There are differences between the performances of some of the classifiers that belong to similar paradigms. If these differences do not give direct information about FQG, they help to spot which strategies are the most successful ones when dealing with this data set. Indirectly, this provides information about the difficulties that FQG overcomes when it outperforms such methods. The levels of significance of the dif-

ferences of performance between the Nearest Neighbours classifiers are expressed in table 4.3 and 4.4.

Table 4.3: T-test levels of significance for the measured differences between IB1 and fuzzy k-nearest neighbour

Measure	p value
Accuracy	0.09189
Precision	0.05237
Recall	0.17786
F-Measure	0.05244
Specificity	0.88637
NPV	0.06204
MCC	0.11711

Table 4.4: T-test levels of significance for the measured differences between fuzzy k-nearest neighbour and fuzzy k-rough nearest neighbour

Measure	p value
Accuracy	0.01285
Precision	0.04840
Recall	0.04314
F-Measure	0.00714
Specificity	0.94560
NPV	0.01988
MCC	0.02011

Although the difference is not significant for Specificity, results show that, among the Nearest Neighbours classifiers, the Fuzzy K-Rough Nearest Neighbour classifier performs best, followed by respectively Fuzzy K-Nearest Neighbour and IB1. The downside of a simple approach like IB1 is the lack of robustness . The high degree of local sensitivity makes this nearest neighbour classifier highly susceptible to noise in the training data. One of the drawbacks of IB1 might be that, while assigning class membership values (i.e., the weights that represent the likelihood of different secondary structure types), atypical vectors and true representatives of the classes are

given equal importance. Secondly, once the class has been assigned to a vector, there is no indication of the strength (significance) of membership to indicate how much the vector belongs to a particular class. Fuzzy K-Nearest Neighbours performs better due to the fact that when determining the class of the current object, the algorithm is capable of taking into consideration the ambiguous nature of the neighbours if any. The algorithm has been designed such that these ambiguous neighbours do not play a crucial role in the classification of the current object. Another advantage is that objects are assigned a membership value in each class rather than binary decision of the type "belongs to" or "does not belong to". The advantage of such assignment is that these membership values act as strength or confidence with which the current object belongs to a particular class. Fuzzy K-Nearest Neighbour, as argued in Sarkar (2000) might have problems to deal adequately with insufficient knowledge. In particular, when every training pattern is far removed from the test object, and hence there are no suitable neighbours, the algorithm is still forced to make clear-cut predictions. Fuzzy Rough K-Nearest Neighbour constructs a lower and upper approximation of each decision class, and then computes the membership of the test objects to these approximations. This ability to assume and deal with insufficient knowledge explains the gain of performance. The fact that FQG outperforms these methods indicates that it is able to successfully emulate some of their advantages, namely three. First, the ability to generalise a model from similar samples (by mapping a distribution to the known learning samples). Secondly, the ability to deal with "fuzzy uncertainty" caused by overlapping classes. And thirdly, the ability to deal with partial knowledge (FQG has a parameter that capture that concept when expressing the degree of incompleteness of the learning sample). This last trait is not the same thing as being able to deal with insufficient information, but it allows one to deal with sufficient but sparse data.

Overall the significance of the difference of performance between Bayes Net and Naive Bayes is expressed in table 4.5. Results show that Bayesian Nets generally perform significantly better than Naive Bayes. This can be explained by Bayesian Nets ability to give conditional probability distribution of the classification node given values of other attributes. Although FQG has an ability similar to Naive Bayes to decouple conditional feature distributions into n one-dimensional sets, it is also able to link these features together when estimating an average or weighted average that combines the memberships of all these features. This is not as fine grained as a network

Table 4.5: T-test levels of significance for the measured differences between Bayes net and naive Bayes

Measure	p value
Accuracy	0.03265
Precision	0.12371
Recall	0.05983
F-Measure	0.11863
Specificity	0.76990
NPV	0.10226
MCC	0.02356

of probability distributions, but this is enough to make features interdependent. So, in a way, FQG might be emulating these particular properties of Bayesian based classifiers.

Overall the significance of the difference of performance between CART and J48 is expressed in table 4.6. Results show that CART generally performs significantly better

Table 4.6: T-test levels of significance for the measured differences between CART and J48

Measure	p value
Accuracy	0.17309
Precision	0.18356
Recall	0.31903
F-Measure	0.11761
Specificity	0.79407
NPV	0.30539
MCC	0.15650

than J48. This could be explained by the fact that the CART algorithm is using Gini impurity, while, on the other side J48 is using Information Gain. Raileanu & Stoffel (2004) analyzed the difference of the frequency of agreement/disagreement between Gini Index function and Information Gain on decision points. This difference has been found to be only in 2%. This explains why most empirical studies conjectured that there is no significant difference between the two criteria. As this difference seems to

have a significant impact on the classifiers efficiency on this particular data set, one can only hypothesize that the data have quite specific and unusual properties that remain to be defined in further work.

Overall the difference of performance between RBF network and multilayer perceptron is not significant as shown in table 4.7. This suggests that, in the context of this experiment, Multilayer Perceptron RBF Network exhibit roughly similar performances.

Table 4.7: T-test levels of significance for the measured differences between RBF Network and multilayer perceptron

Measure	p value
Accuracy	0.86449
Precision	0.18998
Recall	0.91163
F-Measure	0.43333
Specificity	0.36763
NPV	0.96124
MCC	0.86238

There seem to be several distinct advantages to using fuzzy quantile generation at the core of this framework. If the experimental setup of this study implies that BVH human skeletal representation is obtained via pre-processing of the data from cameras, fuzzy quantile generation as a machine learning technique, can be applied directly to input data without pre-processing. A simple list of examples that associate discrete class values to real numbers is enough to learn a model and perform a classification. Also, there is no need to apply discretisation, dimensionality reduction, or time segmentation to the data. Learning samples can also use artificially created classes that could not be obtained via standard clustering methods and would otherwise give singular matrices when using methods such as HMM. Finally models can be obtained from very few examples, and parameters can be set to tailor the precision of every model to the quantity of available data.

4.3 Case study for context aware classification

The accuracy of FQG for individual frames has been measured and has shown satisfying results. The next step is to look at the accuracy of the classifier regarding frames depending on their relative positions. The "context-aware" accuracy (as opposed to the "short-sighted" accuracy of an individual frame) of FQG is compared to a mixture of FQG and Genetic Programming(see Figure 4.12) over five different moves (Jab, Cross, Right Hook, Left Hook, and Lower Left Hook) performed by each one of all three individuals (data are 3-fold validated). The data set is spread over 32 files representing a total of 22938 frames that are known instances of these five moves. Results show the accuracy when the same individuals are used for learning and testing and when the individuals used for learning are different from the individuals used for testing. A set

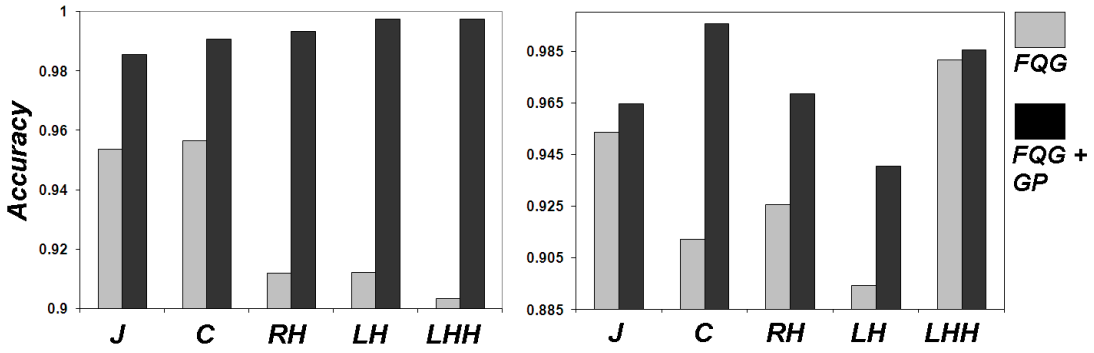


Figure 4.12: Comparing "context-aware" accuracy: FQG versus FQG+GP

of rules is generated using Genetic Programming from learning samples. These rules are then used to filter the output of FQG over test samples. A t-test confirms with 95% confidence that the mixture of FQG and Genetic Programming performs significantly better than FQG alone, even with as little as four rules in total. Although the models performed well when the individual concerned formed part of the training group, the classifier performance worsened significantly when they were removed. Despite this phenomenon in line with previous findings Yamato *et al.* (1992) Darby *et al.* (2007), it is worth noticing that the association of FQG and GP still shows consistently better results than FQG on its own (appendix F[3,5]), while keeping the ability of FQG to learn from small data sets without pre-processing.

4.4 Occlusion module evaluation

Feature selection and feature reconstruction as presented in this chapter are being tested in experiments involving the recognition of stances from occluded data. The purpose is not to show any sort of pre-eminence of the present system to other existing methods, but rather to demonstrate a "proof of concept" regarding the ability of FQG to be extended to deal with occlusion.

4.4.1 Feature selection evaluation

The performance of Laplace similarity is compared with Gaussian and triangle-2 fuzzy similarity relations. Each Fuzzy Similarity relation is measured by taking into account both the reduction of the number of attributes and the resulting accuracy of a Naive Bayesian Classifier. Compression is measured with the ratio of attributes selected the total number of attributes. Accuracy is obtained with the average over a 10 fold evaluation of the accuracy of the Bayesian classifier. Results are obtained with a boxing motion sample coming from 3d motion capture data (figure 4.8). Additional results using different datasets from the UCI machine learning repository are available in appendix C. A version of the WEKA data-mining package from Hall *et al.* (2009) modified by Jensen & Shen (2008) in order to include Fuzzy Rough Feature Selection was employed to compute these results (appendix F[2]). For all datasets, Fuzzy Laplace Similarity displays the highest compression rate for the first to second best accuracy. When combining both compression and accuracy by evaluating the percentage of accuracy gained per attribute, it becomes possible to compare the performance of each measure of similarity (figure 4.13). In all cases, when looking at the ratio of compression over accuracy, the Fuzzy Laplace Similarity relation systematically outperforms

Table 4.8: Accuracy and compression of Laplace, gaussian and triangular-2 fuzzy similarity relations

Dataset	Similarity	Classifier accuracy	Features selected
Boxing sample	Laplace	88.54%	23.08%
	Gaussian	97.04%	38.46%
	Triangular-2	93.70%	30.77%

the Gaussian and the Triangular-2 relations. These results are promising and the authors intend to use the Laplace similarity relation in the context of attribute selection when developing the occluded human motion classification framework. In the case of

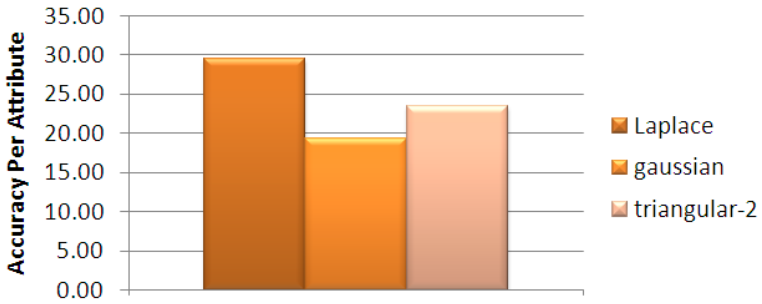


Figure 4.13: Comparing accuracy per attribute for Laplace, gaussian and triangular-2 similarity relations

the motion capture boxing sample, Triangular 2 similarity gave a smaller reduct set (with 3 numbered features 45, 48, 56) than Gaussian similarity (with 4 features 7, 44, 45, 48 - see table 4.9). They both gave reduct sets with similar dataset consistency around 0.91. Laplace similarity gave the smallest subset of all with only 2 attributes 3, 31 and a data consistency around 0.95. All these ended up with an acceptable accuracy when classifying the motions. Triangular-2 selects the head right shoulder and right elbow rotations to differentiate boxing motions. Gaussian selects the left knee, right shoulder, and right elbow to differentiate motions. Interestingly, Laplace similarity uses the left shoulder and the hip to differentiate boxing motions. This is actually closer to what boxing experts watch as they take into account not only the upper body movements but also the hips rotations. Results show that the Fuzzy Laplace Similarity relation outperforms other known fuzzy similarity relations such as Gaussian and Triangular-2 measures in the context of Fuzzy Rough Feature Selection. This new fuzzy similarity relation is integrated into the framework in order to identify joints essential to the classification of motions. In this data set, these are found to be shoulders and hip. The FRFS algorithm would estimate uncertainty around respectively 0.95 if all these essential joint are occluded, 0.316 if only one of them is occluded, and 0.05 if none is occluded - we only consider left or right shoulders as the hips are always visible in this data set. The case when only one of them is occluded is of interest as it

Table 4.9: Numbered key features of joint Euler angles identified by fuzzy rough feature Selection (gaussian: $*$, triangular-2: $\bullet$, Laplace: $\circ$)

Joint Name	Euler Z	Euler X	Euler Y
ROOT Hips	1	2	3 $\circ$
JOINT LeftHip	4	5	6
JOINT LeftKnee	7 $*$	8	9
JOINT LeftAnkle	10	11	12
JOINT RightHip	13	14	15
JOINT RightKnee	16	17	18
JOINT RightAnkle	19	20	21
JOINT Chest	22	23	24
JOINT Chest2	25	26	27
JOINT LeftCollar	28	29	30
JOINT LeftShoulder	31 $\circ$	32	33
JOINT LeftElbow	34	35	36
JOINT LeftWrist	37	38	39
JOINT RightCollar	40	41	42
JOINT RightShoulder	43	44$*$	45$*\bullet$
JOINT RightElbow	46	47	48$*\bullet$
JOINT RightWrist	49	50	51
JOINT Neck	52	53	54
JOINT Head	55	56 $\bullet$	57

Table 4.10: Classification of motions with one occluded shoulder joint

Joint	Nb Frames	Correct Guesses	Rates %
Right Shoulder	3389	2604	76.83%
Left Shoulder	3096	2343	75.67%
Total	6490	4947	76.22%

appears about 60% of the time in the present boxing motions data samples. If the Euler Angles rotations of these joints can be accurately reconstructed, it becomes possible to reduce uncertainty and suggest plausible end results for the classification.

4.4.2 Case study for feature reconstruction

The ability to suggest correct plausible moves from occluded data is evaluated on samples with simulated occlusion from 8 different points of views. Occlusion is reconstructed using Matlab by computing which joints are masked by a 3 dimensional mesh composed of cylinders representing the limbs and portions of the human body. This multiplies by 8 the size of the data set. The process of generation of these occluded data is computationally expensive and slow. Therefore, considering the time constraints, one sample of reasonable complexity is used for testing: a combination of guards and jabs. A binary mask expressing the state of occlusion for all 19 joints is produced at every frame. It seems that in more than half of the cases (58.35% of all the frames when combining the two samples), only one of the essential joints that were previously determined through feature selection is occluded. In more than 99% of the cases, this is a shoulder joint. This represents a data set of around 6490 frames with one shoulder joint occluded. As shown in table 4.10, the most plausible rotation chosen for occluded shoulder joints are leading to the correct classification of the moves for 4947 of these 6490 frames. That is to say that, when looking at one participant performing moves from different points of views, on cases when one shoulder joint is occluded (in 58% of all frames), the classifier suggests the correct moves 76.2% of the time (appendix F[4]). The occluded data classification is likely to improve when applying the GP filter. This experiment shows that FQG can be extended to deal with occlusion reasonably well. Initially all occluded data would be intractable. After using the system described above on the given sample, around 44% of all the occlusions scenarios

are correctly guessed. One drawback of this approach is that it is computationally expensive. The number of Fuzzy Qualitative Angles generated to express one plausible rotation often exceeds 160. Having 3 plausible rotations for one occluded joint is not rare. The system has to compute geometrical distances to seven Fuzzy Membership Functions of known moves, that is to say $3 \times 160 \times 7 = 3360$ operations. Then, it has to sort all these in order to find the three closest moves suggested by all these plausible rotations. These numerous operations have to be computed for one frame of average occlusion complexity. This suggests that future improvements are needed in order to apply this occluded motion recognition process in near real-time conditions.

4.5 Summary

Initial results show that when put to the test with a boxing data set presenting challenges such as real biologically "noisy" data, cross-gait differentials from one individual to another, relatively high dimensionality (with a skeletal representation that has 57 degrees of freedom) and learning samples of suboptimal size, the FQG based classifier outperforms sixteen other known machine learning techniques. Further results also show that the context-aware filter improves FQG performance consistently, and that the framework can deal successfuly with occluded data. Experimental results therefore demonstrate the effectiveness of the proposed inference engine for 3d occluded human motion recognition.

Chapter 5

Conclusions

5.1 Overview

The recognition of 3D human motions is a challenging task that requires view-invariant actions to be recognised a) independently from differences in execution in the spatial and temporal domains resulting in overlapping classes, b) by using learning samples of sub-optimal size, c) in a context-aware and time-sensitive fashion, and d) despite occlusion. A review of related techniques confirms a new growing trend of hybrid Machine Learning methods that seek to fit into a niche satisfying the following functional requirements: the ability to classify from learning samples of sub-optimal size, a low sensitivity to noise, and simplicity regarding the parameter tuning process. This thesis has addressed the above issues with three contributions that can be described as follows. First, a standalone classifier using Fuzzy Quantile Generation, a novel method that generates Fuzzy Membership Functions using metrics derived from the probabilistic quantile function. This method has demonstrated its effectiveness on the classification of noisy, imprecise and complex motions while using learning samples of sub-optimal size with motion capture data from real boxers. Fuzzy Quantile Generation outperforms other time-invariant classifiers in a comparative study made on the boxing data set. Secondly, a genetic programming based filter is developed to produce time-dependent and context-aware rules to smooth the qualitative outputs of fuzzy quantile generation. Various factors such as speed, previous and next movements, and best ranked membership scores are taken into account to generate a complex and subtle network of conditional statements that would otherwise be difficult to identify in

an empirical fashion for a human observer. Experimental results on the boxing motion capture data show that the filter consistently improves the accuracy of the Fuzzy Quantile Generation classifier. Thirdly, the occlusion module based on feature selection and reconstruction. The new Laplace fuzzy similarity relation developed in the context of Fuzzy Rough Feature Selection to identify important joints in case of occlusion is shown to outperform other measures consistently over four different standard benchmarking datasets from the University of California, Irvine (UCI) Machine Learning Repository and on the boxing motion capture sample. The feature reconstruction mechanism reduces the uncertainty caused by occlusion by suggesting plausible rotational data from hidden joints. This work uses Fuzzy Qualitative Euler Angles, a modified version of the Fuzzy Qualitative Trigonometry representation system exposed in Liu & Coghill (2005) and Liu (2008). Results show that the system correctly guesses 44% of the initially intractable occluded data in the boxing experiment.

The motion recognition framework has demonstrated its effectiveness on motion capture data from real boxers in terms of fuzzy membership generation, context-aware rule generation, and motion occlusion. It is worth noting that the motion capture data presents challenges for classification problem in general: real biologically "noisy" data, cross-gait differentials from one individual to another, relatively high dimensionality of a skeletal representation that has 57 degrees of freedom, and large number of learning samples of suboptimal size.

5.2 Conclusion

The theoretical significance of the motion recognition framework can be explained in light of the following considerations.

First, the successful combined application of the contributions detailed above to the demanding problem of 3d motion capture data classification validates the presented work and confirms the potential of this framework as an effective contender in the field of motion recognition. The suitability of this methodology is reinforced by the fact that at present, there is a growing need for techniques that can deal with view-invariant based representations.

Secondly, the methodological impact of Fuzzy Quantile Generation lies in bridging fuzzy set theory to probability theory through a modelling paradigm that leads to practical results. The direct mapping from membership function to probability distribution allows the framework to deal efficiently with problems such as the spatio-temporal variations resulting in overlapping classes, the scarcity of learning samples and the simplicity of parameter tuning.

Thirdly, the extension of Genetic Programming with strongly-typed mechanisms and its successful application to the problem of motion classification demonstrate the suitability of this evolutionary paradigm for the novel application problem of 3D motion capture classification.

Finally, the effective handling of occlusion from motion capture data independently from image based representations contributes to the novelty and the validity of the feature selection/reconstruction approach in the context of motion classification.

The motion classification methodology exposed in this thesis can potentially impact the area of behaviour understanding, but there is still room for further improvement, and possible extensions to different application domains can be considered as explained in the next section.

5.3 Future work

There are a number of improvements as well as substantial additions that could be made to the work that has been discussed. Future work might involve modifying Fuzzy Quantile Generation in order to automate and extend the choice of a probability distribution, enhancing temporal pattern recognition with probabilistic paradigms, optimising the occlusion module, and adapting the present framework to different application domains.

5.3.1 Automating and extending the choice of a probability distributions

The proposed method does not automate the choices of the type of Fuzzy Membership Function and associated type of probability distributions which best represent the domain the sample is extracted from. This analysis is valid for distributions with only one mode. However, there are times when multi-modal distributions might offer a better alternative for different application domains. In order to map a membership to such a distribution, either the aggregation of trapezoidal fuzzy membership functions or the creation of a different type of composite membership function would be required. Although the automated generation of a Fuzzy Membership Function would still use metrics based on the quantile function, some of the mapping mechanisms would need to be accordingly adjusted. This would require not only a delicate refinement of the method, but also new experiments involving data samples that reflect such multinomial distributions.

5.3.2 Enhancing temporal pattern recognition by adding probabilistic paradigms

While the association of Fuzzy Quantile Generation and Genetic Programming shows consistently better results than Fuzzy Quantile Generation on its own, it can be argued that the choice of genetic programming generates very specialised solutions bound to a local optimum in some cases. Considering the omnipresence of probabilistic methods in the context of the context-aware temporal classification, filtering the qualitative outputs of Fuzzy Quantile Generation by using methods such as Bayes Net, or Hidden Markov Models might be a good way to enhance the temporal pattern recognition of the framework. The succession of time-ordered discrete values resulting from Fuzzy Quantile Generation can be used as inputs for the nodes of such representations. Putting aside the time constraints intensified by the complications induced by the nature of the boxing data set, building another additional layer of discrete data based on the outputs of the Fuzzy Quantile Generation classifier, and integrating the resulting model seamlessly in the training and testing process would be the next logical step.

5.3.3 Optimise the occlusion module

One drawback of the presented occluded motion reconstruction approach is that it generates plausible rotations in a computationally expensive way. The number of Fuzzy Qualitative Angles generated to express one plausible rotation is high and the system has to compute geometrical distances to all of these and then rank them. These numerous operations have to be computed frame by frame and future improvements are needed in order to apply this occluded motion recognition process in near real-time conditions. One possible approach might be to use a pruning strategy based on probabilistic knowledge by limiting the generation not just to plausible, but also to most expected rotations inferred from a knowledge base that would take into account previous states of rotation. Considering the inherent complexity of the data, building such a sizable knowledge base and using it for the real-time inference of occluded data might make a very useful and challenging prospect.

5.3.4 Adaptation of the present framework to different application domains

Finally, considering the flexibility of the present framework, extensions to similar application domains such as surveillance, elderly health care, and Human-Computer Interaction are under consideration. Independently from the initial focus on human motion recognition, Fuzzy Qualitative Generation is reusable in the more general context of automated Fuzzy Membership Function generation. This is especially true as fuzzy quantile generation as a machine learning technique, can be applied directly to real valued data associated with discrete class values in order to learn a model and perform a classification.

References

AHA, D.W., KIBLER, D. & ALBERT, M.K. (1991). Instance-based learning algorithms. *Machine Learning*, **6**, 37–66. 122

AHMAD, M. & LEE, S.W. (2006). Human action recognition using multi-view image sequences features. In *FGR '06: Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition*, 523–528, IEEE Computer Society, Washington, DC, USA. 14

ALLMEN, M. & DYER, C. (1990). Cyclic motion detection using spatiotemporal surfaces and curves. In *Pattern Recognition, 1990. Proceedings., 10th International Conference on*, vol. 1, 365–370. 12

APPLE (2007). iphone mobile phone. Website: http://www.apple.com/uk/iphone/. x, 16

ARULAMPALAM, M.S., MASKELL, S., GORDON, N. & CLAPP, T. (2002). A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, **50**, 174–188. 24

ATHANASIADIS, I.N., KABURLASOS, V.G., MITKAS, P.A. & PETRIDIS, V. (2003). Applying machine learning techniques on air quality data for real-time decision support. In *In: First International NAISO Symposium on Information Technologies in Environmental Engineering (ITEE'2003*, 24–27, ICSC-NAISO Publishers. 126

BASTIAN, A. (2000). Identifying fuzzy models utilizing genetic programming. *Fuzzy Sets Systems*, **113**, 333–350. 5

BELARBI, K., TITEL, F., BOUREBIA, W. & BENMAHAMMED, K. (2005). Design of mamdani fuzzy logic controllers with rule base minimisation using genetic algorithm. *Engineering Applications of Artificial Intelligence*, **18**, 875–880. 5, 45

BENOIT, E., ALLEVARD, T., UKEGAWA, T. & SAWADA, H. (2003). Fuzzy sensor for gesture recognition based on motion and shape recognition of hand. 63–67. 30

BIRCHFIELD, S. & TOMASI, C. (1996). Depth discontinuities by pixel-to-pixel stereo. *International Journal of Computer Vision*, **35**, 1073–1080. 17

BLANK, M., GORELICK, L., SHECHTMAN, E., IRANI, M. & BASRI, R. (2005). Actions as space-time shapes. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, vol. 2, 1395 –1402 Vol. 2. x, 12, 13

BOBICK, A.F. & WILSON, A.D. (1997). A state-based approach to the representation and recognition of gesture. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **19**, 1325–1337. 23

BOBICK, A.F., DAVIS, J.W., SOCIETY, I.C. & SOCIETY, I.C. (2001). The recognition of human movement using temporal templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **23**, 257–267. x, 11, 12

BOSER, B.E., GUYON, I.M. & VAPNIK, V.N. (1992). A training algorithm for optimal margin classifiers. In D. Haussler, ed., *Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory*, 144–152, ACM Press, Pittsburgh, PA. 121

BOURLARD, H. & WELLEKENS, C.J. (1990). Links between markov models and multilayer perceptrons. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **12**, 1167–1178. 29

BRAND, M. (1996). Understanding manipulation in video. In *Proc. Int. Conf. Automatic Face and Gesture Recognition*, 94–99. 27

BREGLER, C. (1997). Learning and recognizing human dynamics in video sequences. In *CVPR*, 568, IEEE Computer Society. 14

BREIMAN, L. *et al.* (1984). *Classification and Regression Trees*. Wadsworth. 125

BUGMANN, G. (1998). Normalized gaussian radial basis function networks. *Neurocomputing*, **20**, 97–110. 26, 125

CALLEJAS BEDREGAL, B., DIMURO, G. & ROCHA COSTA, A. (2006). Interval fuzzy rule-based hand gesture recognition. In *Scientific Computing, Computer Arithmetic and Validated Numerics, 2006. SCAN 2006. 12th GAMM - IMACS International Symposium on*, 12 –12. 30

CASE, K. & FAIR, R. (2003). *Principles of economics*. Prentice Hall. 43

CEDRAS, C. & SHAH, M. (1995). Motion-based recognition: A survey. *Image and Vision Computing*, **13**, 129–155. 8

COKES, C. (1980). *The Complete Book of Boxing for Fighters and Fight Fans*. Etc Publications. 67

COOGAN, T., AWAD, G., HAN, J. & SUTHERLAND, A. (2006). Real time hand gesture recognition including hand segmentation and tracking. In G. Bebis, R. Boyle, B. Parvin, D. Koracin, P. Remagnino, A.V. Nefian, M. Gopi, V. Pascucci, J. Zara, J. Molineros, H. Theisel & T. Malzbender, eds., *Advances in Visual Computing, Second International Symposium, ISVC*, vol. 4291 of *Lecture Notes in Computer Science*, 495–504, Springer. 29

COOPER, G.F. & HERSKOVITS, E. (1991). A bayesian method for constructing bayesian belief networks from databases. In B. D'Ambrosio & P. Smets, eds., *UAI '91: Proceedings of the Seventh Annual Conference on Uncertainty in Artificial Intelligence*, 86–94, Morgan Kaufmann. 120

COOPER, G.F. & HERSKOVITS, E. (1992). A bayesian method for the induction of probabilistic networks from data. *Machine Learning*, **9**, 309–347. 120

CORRADINI, A. (2001). Dynamic time warping for off-line recognition of a small gesture vocabulary. In *Proceedings of the IEEE ICCV Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems (RATFG-RTS'01)*, 82, IEEE Computer Society, Washington, DC, USA. 29

DANAFAR, S. & GHEISSARI, N. (2007). Action recognition for surveillance applications using optic flow and SVM. In Y. Yagi, S.B. Kang, I.S. Kweon & H. Zha, eds., *ACCV 2007, 8th Asian Conference on Computer Vision, Proceedings, Part II*, vol. 4844 of *Lecture Notes in Computer Science*, 457–466, Springer. 25, 121

DARBY, J., LI, B. & COSTEN, N. (2007). Human activity recognition: Enhancement for gesture based game interfaces. In *3rd Int. Conf. on Games Research and Development, CyberGames*. 89

DAVIS, J. & SHAH, M. (1994). Visual gesture recognition. *IEEE Proceedings-Vision, Image and Signal Processing*, **141**, 101–106. 23

DEMIRÖZ, G. & GÜVENIR, H.A. (1997). Classification by voting feature intervals. In M. van Someren & G. Widmer, eds., *Machine Learning: ECML-97, 9th European Conference on Machine Learning, Proceedings*, vol. 1224 of *Lecture Notes in Computer Science*, 85–92, Springer. 29, 126

DUBOIS, D. & PRADE, H. (2001). Possibility theory, probability theory and multiple-valued logics: A clarification. *Annals of Mathematics and Artificial Intelligence*, **32**, 35–66. 4

EFROS, A.A., BERG, A.C., BERG, E.C., MORI, G. & MALIK, J. (2003). Recognizing action at a distance. In *IEEE International Conference on Computer Vision (ICCV'03)*, 726–733. x, 11, 12

EISENSTEIN, J. & DAVIS, R. (2004). Visual and linguistic information in gesture classification. In R. Sharma, T. Darrell, M.P. Harper, G. Lazzari & M. Turk, eds., *Proceedings of the 6th International Conference on Multimodal Interfaces, ICMI 2004*, 113–120, ACM. 29, 126

FALCO, I.D., CIOPPA, A.D., MAISTO, D., SCAFURI, U. & TARANTINO, E. (2008). Automatic recognition of hand gestures with differential evolution. In *European Workshop on Evolutionary Computation in Image Analysis and Signal Processing*, 265–274. 29, 122

FANG, G., GAO, W. & ZHAO, D. (2004). Large vocabulary sign language recognition based on fuzzy decision trees. vol. 34, 305 – 314. 30

FAVRE, J., AISSAOUI, R., JOLLES, B.M., SIEGRIST, O., DE GUISE, J.A. & AMINIAN, K. (2006). 3d joint rotation measurement using mems inertial sensors: Application to the knee joint. In *ISB-3D: 3-D Analysis of Human Movement, 28-30 June 2006, Valenciennes, France.*. 18

FRANK, E., HALL, M.A., HOLMES, G., KIRKBY, R. & PFAHRINGER, B. (2005). Weka - a machine learning workbench for data mining. In O. Maimon & L. Rokach, eds., *The Data Mining and Knowledge Discovery Handbook*, 1305–1314, Springer. 29, 74, 119, 126

FRANTTI, T. (2001). Timing of fuzzy membership functions from data. Academic Dissertation - University of Oulu - Finland. 4, 34

FREEMAN, W.T., TANAKA, K., OHTA, J. & KYUMA, K. (1996). In *Computer vision for computer games*, 2nd International Conference on Automatic Face and Gesture Recognition. 11

GABRIEL, P.F., VERLY, J.G., PIATER, J.H. & GENON, A. (2003). The state of the art in multiple object tracking under occlusion in video sequences. In *In Advanced Concepts for Intelligent Vision Systems (ACIVS), 2003*, 166–173. 5, 59

GAVRILA, D.M. (1999). The visual analysis of human movement: A survey. *Computer Vision and Image Understanding*, **73**, 82–98. 8

GOBI, A.F. & PEDRYCZ, W. (2007). Fuzzy modelling through logic optimization. *International Journal of Approximate Reasoning*, **45**, 488–510. 5, 45

GONZALEZ, J. (2004). *Human Sequence Evaluation: The Key-Frame Approach*. Ph.D. thesis, Department of Mathematics and Computer Science, University of Barcelona,, Bellaterra, Spain. 11

GUO, F. & QIAN, G. (2006). Dance posture recognition using wide-baseline orthogonal stereo cameras. In *Automatic Face and Gesture Recognition, 2006. FGR 2006. 7th International Conference on*, 481 –486. 25

HALL, M., FRANK, E., HOLMES, G., PFAHRINGER, B., REUTEMANN, P. & WITTEN, I. (2009). The weka data mining software: An update. *ACM SIGKDD Explorations Newsletter*, **11**, 10–18. 74, 90, 119

HARITAOGLU, I., HARWOOD, D. & DAVIS, L.S. (2000). W4: Real-time surveillance of people and their activities. *IEEE Trans. Pattern Analysis and Machine Intelligence*, **22**, 809–830. 11

HONG, P., TURK, M. & HUANG, T.S. (2000). Gesture modeling and recognition using finite state machines. In *In Proceedings of the Fourth IEEE International Conference and Gesture Recognition*, 410–415. 23

HU, W., TAN, T., WANG, L. & MAYBANK, S. (2004a). A survey on visual surveillance of object motion and behaviors. *IEEE Transactions on Systems, Man and Cybernetics*, **34**, 334–352. 8

HU, W., XIE, D. & TAN, T. (2004b). A hierarchical self-organizing approach for learning the patterns of motion trajectories. *IEEE Transactions on Neural Networks*, **15**, 135–144. 27

ISARD, M. & BLAKE, A. (1996). Contour tracking by stochastic propagation of conditional density. In B.F. Buxton & R. Cipolla, eds., *4th European Conference on Computer Vision*, vol. 1064 of *Lecture Notes in Computer Science*, 343–356, Springer. 24

IVANOV, Y.A. & BOBICK, A.F. (2000). Recognition of visual activities and interactions by stochastic parsing. *IEEE Trans. Pattern Analysis and Machine Intelligence*, **22**, 852–872. 27

JAN, T., PICCARDI, M. & HINTZ, T. (2003). Neural network classifiers for automated video surveillance. *13th IEEE Workshop on Neural Networks for Signal Processing*, 729–738. 26, 125

JENSEN, R. & CORNELIS, C. (2008). A new approach to fuzzy-rough nearest neighbour classification. In C.C. Chan, J.W. Grzymala-Busse & W. Ziarko, eds., *Rough Sets and Current Trends in Computing, 6th International Conference, Proceedings*, vol. 5306 of *Lecture Notes in Computer Science*, 310–319, Springer. 123

JENSEN, R. & SHEN, Q. (2007). Fuzzy-rough sets assisted attribute selection. *Fuzzy Systems, IEEE Transactions on*, **15**, 73–89. 5, 56, 57

JENSEN, R. & SHEN, Q. (2008). Computational intelligence and feature selection: rough and fuzzy approaches. *Ieee Press Series On Computational Intelligence*, 340. 90

JENSEN, R. & SHEN, Q. (2009). New approaches to fuzzy-rough feature selection. *IEEE Transactions on Fuzzy Systems*, **17**, 824–838. 5, 56

JOHN, G.H. & LANGLEY, P. (1995). Estimating continuous distributions in bayesian classifiers. In P. Besnard & S. Hanks, eds., *UAI 95: Proceedings of the Eleventh Annual Conference on Uncertainty in Artificial Intelligence*, 338–345, Morgan Kaufmann. 119

JOHNSON, N. & HOGG, D.C. (1996). Learning the distribution of object trajectories for event recognition. *Image and Vision Computing*, **14**, 609–615. 27

JOJIC, N., HUANG, T., BRUMITT, B., MEYERS, B. & HARRIS, S. (2000). Detection and estimation of pointing gestures in dense disparity maps. In *FG '00: Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition 2000*, IEEE Computer Society, Washington, DC, USA. 11

JU, S. (1996). Human motion estimation and recognition. Tech. rep., University of Toronto. 8

JUANG, C.F. & KU, K.C. (2005). A recurrent fuzzy network for fuzzy temporal sequence processing and gesture recognition. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, **35**, 646 –658. 30

KAÂNICHE, M.B. & BRÉMOND, F. (2009). Tracking hoG descriptors for gesture recognition. In S. Tubaro & J.L. Dugelay, eds., *Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*, 140–145, IEEE Computer Society. 28

KAÂNICHE, M.B. & BRÉMOND, F. (2010). Gesture recognition by learning local motion signatures. In *IEEE Conference on Computer Vision and Pattern Recognition*. 29

KAKADIARIS, I.A. & METAXAS, D. (2000). Model-based estimation of 3d human motion. vol. 22, 1453–1459. 5, 59

KAPUR, A., KAPUR, A., VIRJI-BABUL, N., TZANETAKIS, G. & DRIESSEN, P.F. (2005). Gesture-based affective computing on motion capture data. In J. Tao, T. Tan & R.W. Picard, eds., *Affective Computing and Intelligent Interaction, First International Conference, ACII*, vol. 3784 of *Lecture Notes in Computer Science*, 1–7, Springer, Beijing, China. 25, 119, 121, 124, 125

KEERTHI, S.S., SHEVADE, S.K., BHATTACHARYYA, C. & MURTHY, K.R.K. (2001). Improvements to platt's smo algorithm for svm classifier design. *Neural Computation*, **13**, 637–649. 25, 121

KELLER, J., MICHAEL, G.R. & A., G.J. (1985). A fuzzy k-nearest neighbor algorithm. *IEEE Transactions on Systems, Man, and Cybernetics*, **15**, 581. 122

KHOURY, M. (2009). pystep or python strongly typed genetic programming. Available online at : http://pystep.sourceforge.net/. 5, 45, 136

KIM, J.S., JANG, W. & BIEN, Z. (1996). A dynamic gesture recognition system for the korean sign language (ksl). *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, **26**, 354 –359. 30

KOHAVI, R. (1995). The power of decision tables. In N. Lavrac & S. Wrobel, eds., *Machine Learning: ECML-95, 8th European Conference on Machine Learning, Proceedings*, vol. 912 of *Lecture Notes in Computer Science*, 174–189, Springer. 124

KOLLORZ, E., PENNE, J., HORNEGGER, J. & BARKE, A. (2008). Gesture recognition with a time-of-flight camera. *International Journal of Intelligent Systems Technologies and Applications*, **5**, 334–343. 28

KORDE, S. & JONDHALE, K. (2008). Hand gesture recognition system using standard fuzzy c-means algorithm for recognizing hand gesture with angle variations for unsupervised users. In *Emerging Trends in Engineering and Technology, 2008. ICETET '08. First International Conference on*, 681 –685. 30

KOZA, J.R. (1992). *Genetic Programming: On the Programming of Computers by Natural Selection*. MIT Press, Cambridge, MA. 45, 49

KOZA, J.R. (2007). Welcome to www.genetic-programming.org (a source of information about the field of genetic programming and the field of genetic and evolutionary computation). xi, 46

KUZMANIC, A. & ZANCHI, V. (2007). Hand shape classification using dtw and lcss as similarity measures for vision-based gesture recognition system. 264 –269. 29

KWOK, C.C.T., FOX, D. & MEILA, M. (2003). Real-time particle filter. In S. Becker, S. Thrun & K. Obermayer, eds., *Advances in Neural Information Processing Systems*, 1057–1064, MIT Press. 24

LAFFERTY, J., MCCALLUM, A. & PEREIRA, F. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. 18th International Conf. on Machine Learning*, 282–289, Morgan Kaufmann, San Francisco, CA. 22

LAWRENCE, N.D. (2007). Mocap toolbox for matlab. Available on-line at http://www.cs.man.ac.uk/∼neill/mocap/. 66

LE CESSIE, S. & VAN HOUWELINGEN, J.C. (1992). Ridge estimators in logistic regression. *Applied Statistics*, **41**, 191–201. 121

LEVENTON, M.E. & FREEMAN, W.T. (1998). Bayesian estimation of 3-d human motion. Tech. rep., Mitsubishi Electric Research Laboratories. 22, 120

LIU, H. (2008). A fuzzy qualitative framework for connecting robot qualitative and quantitative representations. In *IEEE Transactions on Fuzzy Systems*, vol. 16, 1522–1530. 4, 6, 59, 96

LIU, H. & COGHILL, G.M. (2005). Fuzzy qualitative trigonometry. *IEEE International Conference on Systems, Man and Cybernetics (IEEE-SMC)*, **2**, 1291– 1296. 4, 96

LIU, H., BROWN, D.J. & COGHILL, G.M. (2008). Fuzzy qualitative robot kinematics. *IEEE T. Fuzzy Systems*, **16**, 808–822. 6, 59

LU, P., HUANG, X., ZHU, X. & WANG, Y. (2005). Head gesture recognition based on bayesian network. In J.S. Marques, N.P. de la Blanca & P. Pina, eds., *Pattern Recognition and Image Analysis, Second Iberian Conference, IbPRIA, Proceedings, Part I*, vol. 3522 of *Lecture Notes in Computer Science*, 492–499, Springer, Estoril, Portugal. 22, 120

MASOUD, O. & PAPANIKOLOPOULOS, N.P. (2003). A method for human action recognition. *Image and Vision Computing*, **21**, 729–743. 11

MAY, S., WERNER, B., SURMANN, H. & PERVOLZ, K. (2006). 3d time-of-flight cameras for mobile robotics. In *IEEE International Conference on Intelligent Robots and Systems*, 790 –795. 17

MEIER, U., STIEFELHAGEN, R., YANG, J. & WAIBEL, A. (2000). Towards unrestricted lip reading. *International Journal of Pattern Recognition and Artificial Intelligence*, **14**, 571–585. 27

MEREDITH, M. & MADDOCK, S. (2000). Motion capture file formats explained. Tech. rep., Computer Science Department, University of Sheffield. 18

MICROSOFT (2010). Microsoft kinect for xbox 360. Website: http://www.xbox.com/en-US/kinect. x, 17

MITRA, S. & ACHARYA, T. (2007). Gesture recognition: A survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, **37**, 311–324. 8

MODLER, P. & MYATT, T. (2008). Recognition of separate hand gestures by time-delay neural networks based on multi-state spectral image patterns from cyclic hand movements. In *Systems, Man and Cybernetics, 2008. SMC 2008. IEEE International Conference on*, 1539 –1544. 27

MOESLUND, T., HILTON, A. & KRUGER, V. (2006). A survey of advances in vision-based human motion capture and analysis. *Computer vision and image understanding*, **104**, 90–126. x, 8, 9, 10, 31

MOESLUND, T.B. & GRANUM, E. (2001). A survey of computer vision-based human motion capture. *Computer Vision and Image Understanding: CVIU*, **81**, 231–268. x, 8, 9, 10

MORI, T., SHIMOSAKA, M. & SATO, T. (2004). Svm-based human action recognition and its remarkable motion features discovery algorithm. vol. 21 of *Springer Tracts in Advanced Robotics*, 15–25, Springer. 25, 121

NG, C.W. & RANGANATH, S. (2000). Gesture recognition via pose classification. In *15th International Conference on Pattern Recognition (ICPR'00)*, Vol III: 699–704. 26, 125

NG, C.W. & RANGANATH, S. (2002). Real-time gesture recognition system and application. *Image and Vision Computing*, **20**, 993–1007. 26, 125

NINTENDO (2006). Wii console. Website: http://www.nintendo.com/wii/console/controllers. x, 16

OIKONOMOPOULOS, A. & PANTIC, M. (2007). Human body gesture recognition using adapted auxiliary particle filtering. In *AVSS '07: Proceedings of the 2007 IEEE Conference on Advanced Video and Signal Based Surveillance*, 441–446, IEEE Computer Society, Washington, DC, USA. 25

OLIVER, N.M., ROSARIO, B. & PENTLAND, A.P. (2000). A bayesian computer vision system for modeling human interactions. *IEEE Trans. Pattern Analysis and Machine Intelligence*, **22**, 831–843. 22

OWENS, J. & HUNTER, A. (2000). Application of the self-organizing map to trajectory classification. In *Proc. IEEE Int. Workshop Visual Surveillance*, 77–83. 27

PENTLAND, A.P., OLIVER, N. & BRAND, M. (1996). Coupled hidden markov models for complex action recognition. In *Massachusetts Institute of Technology, Media Lab*. 22

POLANA, R., NELSON, R. & NELSON, A. (1994). Low level recognition of human motion (or how to get your man without finding his body parts). In *In Proc. of IEEE Computer Society Workshop on Motion of Non-Rigid and Articulated Objects*, 77–82, Press. 11

QUINLAN, J.R. (1993). *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA. 124

RADZIKOWSKA, A.M. & KERRE, E.E. (2002). A comparative study of fuzzy rough sets. *Fuzzy Sets and Systems*, **126**, 137 – 155. 56

RAHMAN, M.M. & ROBLES-KELLY, A. (2006). A tuned eigenspace technique for articulated motion recognition. In *In European Conference on Computer Vision, ECCV' 2006*, 174–185, Springer. 12

RAILEANU, L. & STOFFEL, K. (2004). Theoretical comparison between the gini index and information gain criteria. *Annals of Mathematics and Artificial Intelligence*, **41**, 77–93. 87

RANGARAJAN, K., ALLEN, W. & SHAH, M. (1993). Matching motion trajectories using scale-space. *Pattern Recognition*, **26**, 595–610. 12

REN, H.B. & XU, G.Y. (2002). Human action recognition with primitive-based coupled-hmm. In *16th International Conference on Pattern Recognition*, vol. 2, 494–498. 14

REN, H.B., XU, G.Y. & KEE, S.C. (2004). Subject-independent natural action recognition. In *Sixth IEEE International Conference on Automatic Face and Gesture Recognition (FG'04)*, 523–528. 14

RICQUEBOURG, Y. & BOUTHEMY, P. (2000). Real-time tracking of moving persons by exploiting spatio-temporal image slices. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **22**, 797 –808. 14

RITTSCHER, J., BLAKE, A. & ROBERTS, S.J. (2002). Towards the automatic analysis of complex human body motions. *Image and Vision Computing*, **20**, 905–916. 14

ROH, M.C., CHRISTMAS, W.J., KITTLER, J. & LEE, S.W. (2006). Robust player gesture spotting and recognition in low-resolution sports video. In A. Leonardis, H. Bischof & A. Pinz, eds., *9th European Conference on Computer Vision*, vol. 3954 of *Lecture Notes in Computer Science*, 347–358, Springer. 12

RUMELHART, D., HINTON, G. & WILLIAMS, R. (2002). Learning representations by back-propagating errors. *Cognitive modeling*, 213. 26, 125

SARKAR, M. (2000). Fuzzy-rough nearest neighbors algorithm. In *2000 IEEE International Conference on Systems, Man, and Cybernetics*, vol. 5. 86, 123

SCHLDT, C., LAPTEV, I. & CAPUTO, B. (2004). Recognizing human actions: A local svm approach. In *Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04)*, vol. 3, 32–36. 25, 121

SCHÖLKOPF, B., BURGES, C.J.C. & SMOLA, A.J., eds. (1999). *Fast Training of Support Vector Machines using Sequential Minimal Optimization*, MIT Press, Cambridge, MA. 25, 121

SHAWE-TAYLOR, J. & CRISTIANINI, N. (2004). *Kernel Methods for Pattern Analysis*. CUP. 25, 121

SIDENBLADH, H., BLACK, M.J. & SIGAL, L. (2002). Implicit probabilistic models of human motion for synthesis and tracking. *Lecture Notes in Computer Science*, **2350**, 784–?? 22, 120

SMINCHISESCU, C., KANAUJIA, A. & METAXAS, D. (2006). Conditional models for contextual human motion recognition. *Computer Vision Image Understanding*, **104**, 210–220. 23

SOKOLOVA, M., JAPKOWICZ, N. & SZPAKOWICZ, S. (2006). Beyond accuracy, F-score and ROC: A family of discriminant measures for performance evaluation. In A. Sattar & B.H. Kang, eds., *Australian Conference on Artificial Intelligence*, vol. 4304 of *Lecture Notes in Computer Science*, 1015–1021, Springer. 77

STARNER, T. & PENTLAND, A. (1997). Real-time american sign language recognition from video using hidden markov models. *Motion-Based Recognition*. 22

SU, M.C. (2000). A fuzzy rule-based approach to spatio-temporal hand gesture recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, **30**, 276–281. 30

SUMPTER, N. & BULPITT, A.J. (2000). Learning spatio-temporal patterns for predicting object behaviour. *Image and Vision Computing*, **18**, 697–704. 27

SYMEONIDIS, K. (1996). Hand gesture recognition using neural networks. *Neural Networks*, **13**, 5–1. 26, 125

TAKAHASHI, K., SEKI, S., KOJIMA, E. & OKA, R. (1994). Recognition of dexterous manipulations from time-varying images. In *IEEE Workshop on Motion of Non-Rigid and Articulated Objects*, 23–28. 21, 29

TAYLOR, G.W., HINTON, G.E. & ROWEIS, S.T. (2006). Modeling human motion using binary latent variables. In B. Schölkopf, J.C. Platt & T. Hoffman, eds., *Advances in Neural Information Processing Systems*, 1345–1352, MIT Press. 27

THINGVOLD, J. (1999). Biovision bvh format. 18

UEDA, E., MATSUMOTO, Y., IMAI, M. & OGASAWARA, T. (2003). A hand-pose estimation for vision-based human interfaces. *Industrial Electronics, IEEE Transactions on*, **50**, 676 – 684. 5, 59

UTSUMI, A. & OHYA, J. (1999). Multiple-hand-gesture tracking using multiple cameras. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, 473–478. 5, 59

VAIL, D.L. (2008). *Conditional Random Fields for Activity Recognition*. Ph.D. thesis, CMU School of Computer Science. 23

VERMA, R. & DEV, A. (2009). Vision based hand gesture recognition using finite state machines and fuzzy logic. In *Ultra Modern Telecommunications Workshops, 2009. ICUMT '09. International Conference on*, 1 –6. 30

WADA, T. & MATSUYAMA, T. (2000). Multiobject behavior recognition by event driven selective attention method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**, 873–887. 23

WANG, L., HU, W. & TAN, T. (2003). Recent developments in human motion analysis. *Pattern Recognition*, **36**, 585–601. 8

WANG, S.B., QUATTONI, A., MORENCY, L.P., DEMIRDJIAN, D. & DARRELL, T. (2006). Hidden conditional random fields for gesture recognition. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 2, 1521 – 1527. 29

WANG, X. & DAI, G. (2007). A novel method to recognize complex dynamic gesture by combining hmm and fnn models. In *Computational Intelligence in Image and Signal Processing, 2007. CIISP 2007. IEEE Symposium on*, 13 –18. 30

WARREN, G. (2000). *Statistical Modelling With Quantile Functions*. Chapman and Hall/CRC. 36

WEINLAND, D. (2008). *Action Representation and Recognition*. Ph.D. thesis, Institut National Polytechnique de Grenoble, France. 2

WEINLAND, D., RONFARD, R. & BOYER, E. (2005). Motion history volumes for free viewpoint action recognition. In *IEEE International Workshop on Modeling People and Human Interaction*, 104, 249–257, IEEE, Beijing, China. x, 12, 13

WELCH, G. & BISHOP, G. (1995). An introduction to the kalman filter. Tech. Rep. TR 95-041, University of North Carolina at Chapel Hill, Department of Computer Science, Chapel Hill, NC 27599-3175. 24

WU, J., PAN, G., ZHANG, D., QI, G. & LI, S. (2009). Gesture recognition with a 3-d accelerometer. In *Proceedings of the 6th International Conference on Ubiquitous Intelligence and Computing*, 38, Springer-Verlag. 119

YAMATO, J., OHYA, J. & ISHII, K. (1992). Recognizing human action in time-sequential images using hidden markov model. In *IEEE Computer Vision and Pattern Recognition*, 379–385. 14, 22, 89

YANG, M.H. & AHUJA, N. (1998). Extraction and classification of visual motion patterns for hand gesture recognition. In *in Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 892–897. 26

YANG, M.H. & AHUJA, N. (1999). Recognizing hand gesture using motion trajectories. In *Proceedings of the IEEE Computer Science Conference on Computer Vision and Pattern Recognition*, 466–472, IEEE, Los Alamitos. 26

YASIN, H. & KHAN, S.A. (2008). Moment invariants based human mistrustful and suspicious motion detection, recognition and classification. In *UKSIM '08: Proceedings of the Tenth International Conference on Computer Modeling and Simulation*, 734–739, Washington, DC, USA. 123

YEASIN, M. & CHAUDHURI, S. (2000). Visual understanding of dynamic hand gestures. *Pattern Recognition*, **33**, 1805–1817. 23

YILMAZ, A. & SHAH, M. (2005). Actions sketch: a novel action representation. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005*, vol. 1, 984 – 989. x, 12, 13

YU, T., WILKINSON, D. & XIE, D. (2003). A hybrid GP-fuzzy approach for resevoir characterization. In R.L. Riolo & B. Worzel, eds., *Genetic Programming Theory and Practice*, chap. 17, 271–290, Kluwer. 5, 45

ZADEH, L. (1995). Discussion: Probability theory and fuzzy logic are complementary rather than competitive. *Technometrics*, **37**, 271–276. 4

ZADEH, L. (2008). Toward human level machine intelligence - is it achievable? the need for a paradigm shift. *Computational Intelligence Magazine, IEEE*, **Volume 3**, 11 – 22. 4

ZADEH, L.A. (1978). Fuzzy sets as a basis for a theory of possibility. *Fuzzy sets and systems, 1978*, **1**, 3–28. 4

ZADEH, L.A. (1986). Fuzzy sets. *Information and Control*, **8**, 338–353. 19, 35

ZHANG, J., LIN, H. & ZHAO, M. (2009). A fast algorithm for hand gesture recognition using relief. In *Fuzzy Systems and Knowledge Discovery, 2009. FSKD '09. Sixth International Conference on*, vol. 1, 8 –12. 29

ZHANG, X. & NAGHDY, F. (2005). Human motion recognition through fuzzy hidden markov model. *Proceedings of the International Conference on Computational Intelligence for Modelling*, **Volume 02**, 450–456. 30

ZHU, J., WANG, L., YANG, R. & DAVIS, J. (2008). Fusion of time-of-flight depth and stereo for high accuracy depth maps. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008*, 1 –8. x, 17

ZIAIE, P., MÜLLER, T., FOSTER, M. & KNOLL, A. (2009). A naïve bayes classifier with distance weighting for hand-gesture recognition. *Advances in Computer Science and Engineering*, 308–315. 29

Appendix A

Acronyms

API	Application Programming Interface
BVH	Biovision Hierarchy file format
CRF	Conditional Random Fields
DTW	Dynamic Time Warping
CART	Classification and Regression Trees
FLR	Fuzzy Lattice Reasoning classifier
FMF	Fuzzy Membership Function
FQG	Fuzzy Quantile Generation
FRFS	Fuzzy Rough Feature Selection
FSM	Finite State Machine
GP	Genetic Programming
HMM	Hidden Markov Models
IB1	Instance-Based learning for $k = 1$
MCC	Matthews Correlation Coefficient
MHI	Motion-History Images
MHV	Motion-History Volumes
RBM	Restricted Boltzmann Machines
ROC	Receiver Operating Characteristic
RVM	Relevance Vector Machines
SMC	Sequential Monte Carlo

SMO	Sequential Minimal Optimization
STV	Spatio Temporal Volumes
SVM	Support Vector Machines
PCA	Principal Component Analysis
PySTEP	Python Strongly Typed gEnetic Programming
TDNN	Time Delay Neural Network
TSK	Takagi-Sugeno-Kang
UCI	University of California - Irvine
VFI	Voting Feature Interval
WEKA	Waikato Environment for Knowledge Analysis

Appendix B

The Waikato Environment for Knowledge Analysis classifiers

The classifiers implemented using the WEKA Machine Learning package presented in Frank *et al.* (2005) and Hall *et al.* (2009) represent diverse paradigms that can be roughly classified into seven types: Bayes, Function based, Nearest Neighbours, Tree based, Rule based, and Miscellaneous. They are used "as is" from the Weka library using mostly default settings.

B.1 Bayesian methods

The Bayesian methods used in this comparative work are Naive Bayes and Bayes Net classifiers.

Naive bayes presented by John & Langley (1995) makes use of Bayes' theorem to predict which class an example most likely belongs to. Wu *et al.* (2009) and Kapur *et al.* (2005) have been using it for comparative purpose in the domain of gesture recognition. It is said to be "naïve" because it assumes that all the attributes characterizing a class are independent from each other. It chooses the class that maximizes the likelihood of the feature assignments for one example. Naive Bayes classifiers

exhibit useful properties: firstly, parameters such as means and variances can be estimated with a small amount of training data. Secondly, the assumption of independence between variables limits the amount of computation required. The reason being that there is no need to find the entire covariance matrix when only the variances of the variables for each class are required. Thirdly the absence of connective logic between features distributions for each class simplifies the representation to independent one-dimensional distributions. The size of the data set does not need to be exponentially scaled with the number of features, which gives Naive Bayes a certain resilience to the curse of dimensionality. Lastly, Naive Bayes displays a good robustness, especially when dealing with models presenting a reduced coupling between variables.

Bayes net presented by Cooper & Herskovits (1991, 1992) uses a Bayesian network, that is to say a directed acyclic graph where nodes represent variables, and edges represent conditional dependencies. Unconnected nodes represent conditionally independent variables. Probability functions associate a set of input value from a node's parent variables to the probability of the variable represented by the node. Bayes Net can be used to compute the conditional probability of one node, given values assigned to the other nodes. This gives the posterior probability distribution of the classification node given the values of other attributes. Bayes Net has been used for recognition of head gestures by Lu *et al.* (2005), general motion understanding from video sequences by Leventon & Freeman (1998), and similarly, has been used by Sidenbladh *et al.* (2002) in conjunction with optical flow. Bayes Nets are theoretically well equipped to deal with classification problems in the sens that they maximize the expected utility of choices. They inherit as well Naive Bayes' robustness and their performances do not drop dramatically when models are slightly modified by small alterations. Since they take into account dependencies between variables, they can also handle incomplete data. On the other hand, they can also present distinct disadvantages. The validity of the network is generally highly dependent on accuracy of the prior model of beliefs. Furthermore, the whole network has to be computed to get the probability of any node, which is known to be a NP-hard (nondeterministic polynomial-time hard) problem. This can sometimes result in Bayes Net to be computationally expensive.

B.2 Function based classifiers

FQG is compared to two function based classifiers: Support Vector Machine(SVM) and Logistic Regression.

Support vector machine presented by Boser *et al.* (1992) is trained using Sequential Minimal Optimization or SMO. Support Vector Machine is a kernel based method that maps examples of different categories to "points" or hyperplanes in a high dimensional space. This hyperplane-based representation amplifies the differences between examples that belong to different categories. This gap called functional margin is maximized to be the largest distance to the nearest training data points of any class. The wider and the clearer the gap, the easier the separation that leads to the classification of new examples, and the lower the generalisation error of the classifier. Finding the parameters that define the hyperplane with the maximum margin is a non trivial optimisation problem. The SMO algorithm used here to train the SVM classifier attempts to solve this problem by scaling it down into 2-dimensional sub-units. Further details about SMO can be found in Schölkopf *et al.* (1999) and Keerthi *et al.* (2001). SVM has been used to recognize human actions from video samples by Schldt *et al.* (2004), and in conjunction with optical flow by Danafar & Gheissari (2007). Mori *et al.* (2004) also used SVM to discover remarkable motion features. Kapur *et al.* (2005) used it as a comparison method in motion classification. The use of kernels makes SVM computationally efficient as it does not have to represent explicitly the feature vectors. Shawe-Taylor & Cristianini (2004) underlined some advantages such as: " the absence of local minima, the sparseness of the solution and the capacity control obtained by optimising the margin".

Logistic regression builds a multinomial logistic regression model, that is to say a function that describes the relationships between independent variables and multiple classes in term of probabilities. Data are fitted to a logistic curve using a variant of Multiple Linear Regression called a Ridge Estimator presented by le Cessie & van Houwelingen (1992). Logistic regression was used by Kapur *et al.* (2005) for classifying emotions from 3d gestures. Logistic regression is quite robust as it does not need independent variables to be normally distributed, or to have an equal variance in

each group. Dependant and independent variables do not have to be linked by a linear relationship. One known drawback of this technique is that it generally requires more data than standard regression to give stable and meaningful results.

B.3 Nearest neighbour classifiers

.

The IB1 Nearest Neighbour, Fuzzy K-Nearest Neighbour, and Fuzzy Rough K-Nearest Neighbour classifiers are used for comparison.

IB1 Nearest-neighbour is the simplest instance-based learning algorithm. The WEKA implementation is based on a paper by Aha *et al.* (1991). In the case of IB1, learning to generalise and classify is a matter of finding the most similar instance (hence the "nearest neighbour") and labeling the next unknown instance with the same label as the known neighbour. One measures the Normalised Euclidean distance to find the training sample closest to an existing test sample and then integrate the latter into the same class. If several closest training instances are equally distant to the same test sample, the first one is selected. Learning is an encapsulation process during which training data are not generalised before the end, which is to say at classification time. That is why such instance-based classifiers are referred to as "lazy learners". IB1 is a simpler case of the K-Nearest Neighbour scenario for $k = 1$(in K-Nearest Neighbour, the object being assigned to the class most common amongst its k nearest neighbours is classified by a majority vote). IB1 has the advantage of simplicity, and is able to cope with little information (missing values are tolerated). One downside is its sensitivity to noise, which can sometimes impair its robustness. IB1 was used for comparison in human gesture recognition by Falco *et al.* (2008).

Fuzzy k-nearest neighbours introduced in Keller *et al.* (1985) extends the K-Nearest Neighbour algorithm by adding features from Fuzzy Set Theory. One direct consequence is that objects can present partial memberships to different classes. Another is that one can take into account the relative importance (closeness) of each neighbour

with respect to the test instance. This is a substantial improvement regarding the capacity to deal with noise compared to the standard Nearest Neighbour approach. However, a study by Sarkar (2000) showed that this fuzzy approach encounters difficulties when confronted to insufficient knowledge. In the training samples are really too different from the test samples, there might not be any suitable "neighbours". In this case, the system will still have to make artificial predictions as the sum of all membership degrees to different classes must always be equal to 1. Fuzzy K-Nearest Neighbour was used as a comparison classifier for benchmark evaluation in human motion recognition by Yasin & Khan (2008).

Fuzzy-rough k-nearest neighbour introduced by Jensen & Cornelis (2008), it extends Fuzzy K-Nearest Neighbour by adding features from Rough Set Theory. The first step of the algorithm consists in building lower and upper approximations for each decision class from the information provided by the nearest neighbours of a test object. The second step of the algorithm computes membership scores of the test object to these approximations using a Gaussian fuzzy similarity relation. The obtained Fuzzy-Rough ownership function allows to handle both fuzzy uncertainty (caused by overlapping classes) and rough uncertainty (caused by insufficient knowledge). This method has not been used yet in motion recognition; however, considering the particularity of the application domain, it might make for an interesting comparison ground.

B.4 Rule learners classifiers

This can be roughly described as a "bottom-up" approach. It first separates the solution space by generating localised rules that cover subsets the training examples. These rules are then iteratively added to each other until the whole training set has been captured. Two methods belonging to rule learner classifiers are used in this study: Conjunctive Rule and Decision table. While these methods are not associated with motion recognition, they might provide clues regarding the data set

Conjunctive rule. The WEKA implementation produces rules that allow the classification of the entire data set. The rules capture sets of conditions that define the most

represented classes in the data set. Each rule is composed of an antecedent and a consequent. The antecedent is constructed using Information Gain in order to select the most appropriate values for the features defining a distribution of resulting class labels (or consequent). The Information Gain of one antecedent can be computed by looking at the weighted average of the entropies of both the data covered and the data ignored by the rule. The generated rule is pruned using Reduced Error Pruning (REP) in which the weighted average of the accuracy rates is used for classification.

Decision table presented by Kohavi (1995) builds a matrix that maps all possible conditional values for independent attributes to all possible outputs (class labels). In order to reduce the model to a smaller and condensed table, a voting system is introduced in order to prune the attributes that introduce little or no change and that are considered irrelevant to the classification.

B.5 Decision tree based classifiers

This can be described as a "top-down" approach which recursively divides general rules into conjunctions of more precise rules fitting the subsets of examples. The division occurs as long as some of the rules live negative examples. The final result is a set of specialised rules which map to positive examples only. The J48 and CART classifiers are used as comparison methods.

J48 is a WEKA implementation of the known C4.5 algorithm presented in Quinlan (1993). It is used in Kapur *et al.* (2005) as comparison in the context of motion recognition. J48 implements a greedy technique using Information Gain (based on the concept of Entropy) in order to determine the most predictive attributes, and create decision points in the tree over the value of each one of these attributes. Numeric attributes can be dealt with using thresholds for the decision splits. This process can sometimes lead to create decision-trees of great length and complexity. This is why the algorithm implements pruning mechanisms based on error minimisation. Overall, this technique generates readable solutions, is robust to noise and generally presents a

good accuracy. Drawbacks can occur such as duplications of sub-trees, or the inability to deal with first-order logic(it cannot refer to several different objects in one test).

CART or minimal cost-complexity pruning is presented by Breiman *et al.* (1984). The CART algorithm build trees in a similar fashion as C4.5, except that it is using Gini Impurity instead of Information Gain to find the most predictive attributes. Like the entropy function, this measure is essentially a way to measure vector sparseness. It is maximised if all classes are evenly distributed and minimum when all instances belong to one class.

B.6 Neural network based classifiers

The Multi-Layer Perceptron, and Radial Basis Function network are used for comparison.

Radial basis function network presented by Bugmann (1998) is used for gesture recognition by Ng & Ranganath (2002) and Ng & Ranganath (2000). It can be defined as a statistical feed-forward two-layer artificial neural network that uses Gaussian radial basis functions as activation functions in their hidden units. Output units are weighted sums of the hidden units' results. A non-linear input is approximated into a linear output. This gives Radial Basis Networks the ability to model and approximate efficiently complex functions.

Multi-layer perceptron presented in Rumelhart *et al.* (2002) is used in video surveillance activity recognition Jan *et al.* (2003), hand gesture recognition Symeonidis (1996), and as comparison method in motion recognition by Kapur *et al.* (2005). This is one of the most popular types of neural networks. A Multilayer Perceptron is a feed-forward artificial neural network model uses at least three layers of nodes with nonlinear activation functions based on Sigmoids.

B.7 Miscellenous classifiers

These algorithms are not necessary used in motion recognition. However, they provide different paradigms known for their simplicity. They can be used as an additional and exploratory comparison ground for the data set.

HyperPipes presented by Frank *et al.* (2005) is an algorithm that, for each class, builds bounds for the attribute-values found in the examples belonging to this class. Each hyperpipe contains each attribute-value found in the examples from the class it was built to cover. A test example is classified by finding the hyperpipe that most contains the instance. Hyperpipes has the advantages of speed, simplicity, and can cope well with large numbers of attributes. Eisenstein & Davis (2004) attempted to develop a human gesture classifier based on HyperPipes.

Voting Feature Intervals (VFI) presented by Demiröz & Güvenir (1997), builds intervals for each attribute inside each class. Class counts are recorded for each interval on each attribute. The predicted class is the one with the highest count. The WEKA implementation uses an attribute weighting scheme where higher weights have a higher confidence expressed as a measure of entropy.

Fuzzy Lattice Reasoning Classifier (FLR) introduced by Athanasiadis *et al.* (2003), induces rules of increasing diagonal sizes from the training data. The size of the maximum number of rules induced is set by a threshold that expresses the degree of granularity of learning. The size of an individual rule is inversely proportional to the number of rules needed to cover the training data. Each rule maps a fuzzy lattice to a class label. There is a mechanism that joints the lattice rules pointing to the same class that are sufficiently similar and formulates corresponding generalised rules of higher size triggered by a fuzzy degree of activation.

Appendix C

Comparing Laplace and other similarity measures over several datasets

Comparison of Laplace, gaussian and triangular-2 fuzzy similarity relations in the context of Fuzzy Rough Feature Selection over four standard datasets from the UCI Machine Learning Repository.

Table C.1: Accuracy and compression of Laplace, gaussian and triangular-2 fuzzy similarity relations

Dataset	Similarity	Classifier accuracy	Features selected
SPECTF	Laplace	88.71%	6.82%
	Gaussian	75.27%	43.18%
	Triangular-2	82.26%	15.91%
German-credit	Laplace	60.16%	29.17%
	Gaussian	84.88%	83.33%
	Triangular-2	64.46%	54.17%
Heart Statlog	Laplace	77.13%	10.53%
	Gaussian	83.62%	21.05%
	Triangular-2	79.18%	14.04%
CMC	Laplace	50.34%	77.78%
	Gaussian	51.22%	88.89%
	Triangular-2	51.22%	88.89%
Boxing sample	Laplace	88.54%	23.08%
	Gaussian	97.04%	38.46%
	Triangular-2	93.70%	30.77%

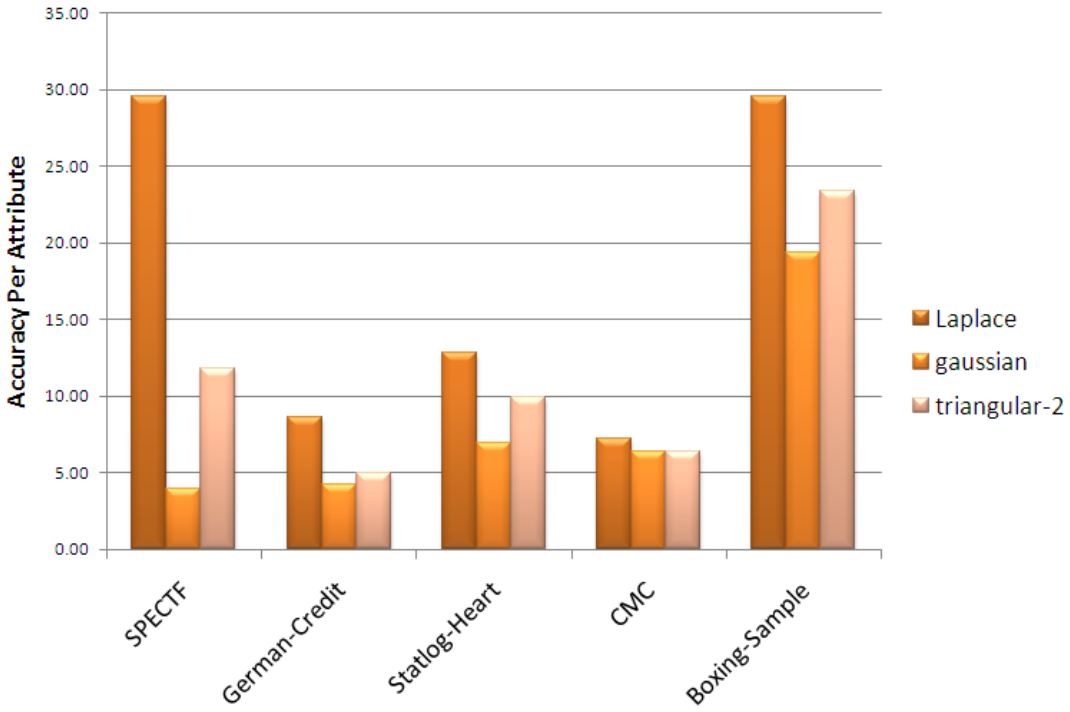


Figure C.1: Comparing accuracy per attribute for Laplace, gaussian and triangular-2 similarity relations on different datasets

Appendix D

Experimental design: consent form and leaflet

INFORMED CONSENT FORM FOR PARTICIPANTS

Probabilistic Template-Based Learning Methods for Continuous Human Motion Matching and Its Application

Investigator: Mehdi Khoury

1. The Purpose of this Research

   The purpose of this research is, by the use of existing 3D human motion tracking and recognition tools, to gather kinetic data on sport practitioners, and then produce qualitative templates that allow computers to generalise these motions to recognizable behaviours. These templates could later be reused anonymously as target models in an application domain such as computer assisted coaching.

2. Procedures

   The room contains sensors and data processing devices that can capture your motion. Infrared sensitive cameras will be directed towards you while you are moving. You are being asked to wear a suit and a set of reflectors, and to perform specific motions such as a walk, a slow run, a jab, a cross, a hook, and an uppercut for a duration of few seconds. This is not strictly speaking a video recording as only the movements of the reflectors are recorded, and no details identifying individual features such as the face are captured. The information is kept as a relatively anonymous kinetic pattern that takes the shape of a moving "stick-figure" skeleton model of the subject. We would like you to use a changing room in order to fit a suit. The technical staff will then help to place reflectors at keys points such as articulations, and guide you through simple standing positions and motions in order to calibrate the system. As a participant in this study, you are requested to perform the following duties:

   (a) Carefully read this informed consent form and then sign it if you agree to participate.

   (b) Wear a suit with a set of sensors.

   (c) Receive instructions and move accordingly while wearing the suit.

3. Risks and Discomforts

   There are no particular risks associated with this experiment, assuming standard motions and demonstration of basic boxing positions are within the range of expertise of the subjects. If executing motions while wearing the suit and the reflectors becomes a source of discomfort, feel free to point this out and immediately remove them. All of the data gathering equipment is inspected and configured

by the Department of Creative Technologies so that it does not present a hazard.

4. Benefits to You

   You will be paid a gratuity for your participation, and if interested, you can be given a video file of the three-dimensional reconstruction of your performance, and get more information on computer assisted coaching systems. There are no other known direct benefits to you. No promises or guarantee of benefits other than those listed in this informed consent form have been made to encourage you to participate. You may however enjoy discovering more about motion capture technologies, and it is likely that your participation in this experiment will help provide a better understanding of how computers can capture and identify human motion and how computer based sport coaching systems might be developed.

5. Extent of Anonymity and Confidentiality

   The information gathered in this experiment will be treated with confidentiality. It will be used for research purposes only, and only by qualified researchers. Your name and other identifiers will be removed from the overall data set and in any resulting publications. As indicated, a ”skeletal” representation will be recorded while you are moving. The record does not include personal identification features. The record will be treated with confidentiality and kept secure. It will be shared only with other qualified researchers, and not published except as noted in the following paragraph. If at a later time we wish to use the recorded motion for other than research purposes, say, for public education, or if we wish to publish (for research or for other purposes) we will only do so after making sure that the study does not identify you either directly or indirectly . Your data will be pooled with that of at least five other participants. (The expected number

of participants is likely to be between five and ten.)

6. Compensation

   You will receive a gratuity for participating in this experiment. You will be paid 20. It is possible that a data gathering equipment malfunction may occur during some portion of your participation. If this should occur, we may have to temporarily suspend the experiment for few minutes to service the data gathering equipment. You may then be asked to extend your participation for a reasonably brief amount of time. If you choose to do so, you will not be paid an additional amount for this delay.

7. Freedom to Withdraw

   As a participant in this research, you are free to withdraw at any time without penalty. If you choose to withdraw, you will be compensated in accordance with the terms in Section VI. of this document.

8. Approval of This Research

   Before this experiment begins, the research must be approved by the Faculty Research Ethics Committee for research involving human subjects at University of Portsmouth. You should know these approvals have been obtained.

9. Participant's Permission

   I have read and understood this informed consent form and conditions of my participation. I have had all my questions answered. I hereby acknowledge the above and give my voluntary consent to participate. If I participate, I understand that I may withdraw at any time without penalty. I agree to abide by the rules of this project.

Participant's Signature..... Date .....

Should I have any questions about this research or its conduct, I may contact:

Mehdi Khoury, Research Investigator

Institute of Industrial Research, Burnaby Building, University of Portsmouth, Burnaby Road, Portsmouth, PO1 3QL

Join an experiment on boxing motion capture !

You may enjoy discovering more about motion capture technologies, and it is likely that your participation in this experiment will help provide a better understanding of how computers can capture and identify human motion and how computer based sport coaching assistance tools might be developed.

You will be asked to perform boxing motion in a suit equipped with captors.

You will get a 3-dimensional animation of your boxing performance in a movie format.

You will be paid £20 per person!

Program for experiment (total duration, maximum 1h40min):

- on arrival, 10 min introduction: participant is briefed on the experiment and given consent and information forms to read and sign.
- putting the suit and captors on : 10 min
- set up and calibration : 1 hour
- data capture of 12 different motions spread over 20 minutes (including: 1 walk, 1 run (low speed), jab, cross, uppercut left, uppercut right, hook left, hook right, 4 steps boxing motion forward, 4 steps boxing motion backward , 4 steps boxing circular motion on the right, 4 steps boxing circular motion on the left) - during this time , the next participant does the arrival and put on another suit...
- changing and leaving...

If interested, contact Mehdi at:

mehdi.khoury@port.ac.uk

or on my mobile: 07964682581

Figure D.1: Leaflet distributed to potential participants

Appendix E

The Python Strongly Typed gEnetic Programming open source package

The PySTEP or Python Strongly Typed gEnetic Programming Open Source package has been specifically developed for the purpose of building a context-aware Genetic Programming filter in the context of this study. It is available for download on Sourceforge (see Khoury (2009)) at http://sourceforge.net/projects/pystep/. The pySTEP website is shown in picture E.1. Figures E.2 and E.3 present in a chronological fashion the project web traffic and download history for all files related to pySTEP from March 2009 to July 2010. All the PySTEP grammar rules that control the syntax of trees in the context of the boxing motion capture experiments are presented in the Python code below.

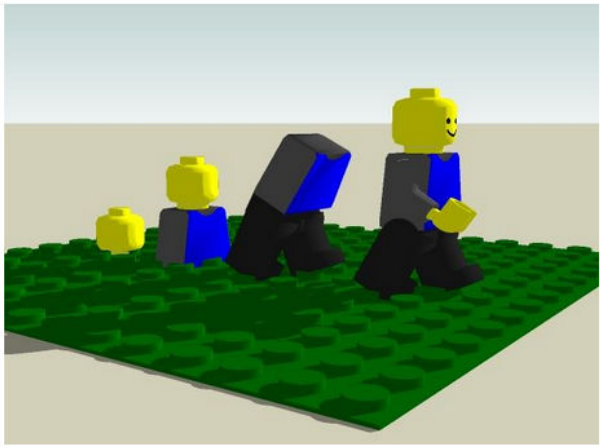
```
1 ifTerminalSet= [(4,0,'is_short'),(4,0,'is_medium'),(4,0,'is_long')]
2 treeRules={
3   'root':[([(2,3,'if_then_replace')],[]),([(2,3,'if_then_replace')
      ],[]),([(2,3,'if_then_replace')],[]),([(2,3,'if_then_replace')
      ],[])],
```

```
4   'if_then_replace':[([(1,1,'membership_1'),(1,1,'membership_2')
        ,(1,1,'membership_3'),\
5     (1,1,'left_1'),(1,2,'left_2'),(1,3,'left_3'),\
6     (1,1,'right_1'),(1,2,'right_2'),(1,3,'right_3'),\
7     (1,2,'and_'),(1,2,'or_'),(1,1,'not_')],ifTerminalSet)\
8     ,([],set_mvts),([],set_mvts)],
9   'membership_1':[([],set_mvts)],
10  'membership_2':[([],set_mvts)],
11  'membership_3':[([],set_mvts)],
12  'left_1':[([],set_mvts)],
13  'left_2':[([],set_mvts),([],set_mvts)],
14  'left_3':[([],set_mvts),([],set_mvts),([],set_mvts)],
15  'right_1':[([],set_mvts)],
16  'right_2':[([],set_mvts),([],set_mvts)],
17  'right_3':[([],set_mvts),([],set_mvts),([],set_mvts)],
18  'and_':[([(1,1,'membership_1'),(1,1,'membership_2'),(1,1,'
        membership_3'),\
19      (1,1,'left_1'),(1,2,'left_2'),(1,3,'left_3'),\
20      (1,1,'right_1'),(1,2,'right_2'),(1,3,'right_3'),\
21      (1,2,'and_'),(1,2,'or_'),(1,1,'not_')],ifTerminalSet),\
22      ([(1,1,'membership_1'),(1,1,'membership_2'),(1,1,'membership_3'
            ),\
23      (1,1,'left_1'),(1,2,'left_2'),(1,3,'left_3'),\
24      (1,1,'right_1'),(1,2,'right_2'),(1,3,'right_3'),\
25      (1,2,'and_'),(1,2,'or_'),(1,1,'not_')],ifTerminalSet)],
26  'or_':[([(1,1,'membership_1'),(1,1,'membership_2'),(1,1,'
        membership_3'),\
27      (1,1,'left_1'),(1,2,'left_2'),(1,3,'left_3'),\
28      (1,1,'right_1'),(1,2,'right_2'),(1,3,'right_3'),\
29      (1,2,'and_'),(1,2,'or_'),(1,1,'not_')],ifTerminalSet),\
```

```
30      ([(1,1,'membership_1'),(1,1,'membership_2'),(1,1,'membership_3'
            ),\
31      (1,1,'left_1'),(1,2,'left_2'),(1,3,'left_3'),\
32      (1,1,'right_1'),(1,2,'right_2'),(1,3,'right_3'),\
33      (1,2,'and_'),(1,2,'or_'),(1,1,'not_')],ifTerminalSet)],
34   'not_':[([(1,1,'membership_1'),(1,1,'membership_2'),(1,1,'
        membership_3'),\
35      (1,1,'left_1'),(1,2,'left_2'),(1,3,'left_3'),\
36      (1,1,'right_1'),(1,2,'right_2'),(1,3,'right_3'),\
37      (1,2,'and_'),(1,2,'or_')],ifTerminalSet)],
38 }
```

pySTEP

Python Strongly Typed gEnetic Programming



A *light* **Genetic Programming API** that allows the user to easily evolve populations of trees with *precise grammatical and structural constraints*. In other worlds you can set up building blocks and rules that define individuals during the evolution.

Why using rules and constraints ?

Let's take the example of the DNA of a human being which is 95% similar to the DNA of a chimp. This means that there is some strong similarity in the building blocks and the structure of the genetic code. A completely random genetic code would make no sense on a functional practical point of view because it would give some kind of unfit and disfunctional goo... I just think that, to be useful, artificial evolution has to be able to integrate basic rules that determine the shape of the individuals generated, so that we can get practical results. This does not mean we eliminate random search and we still admit poorly fit individuals. But at least we focus the search in the direction of the problem by only generating potentially useful solutions.

pySTEP is presently functional, running and stable. Still in the process of adding extra tutorials, extra features for reading the populations from the database, and modify some of the crossover code (so far, only 1-point crossover is used).

Figure E.1: pySTEP website

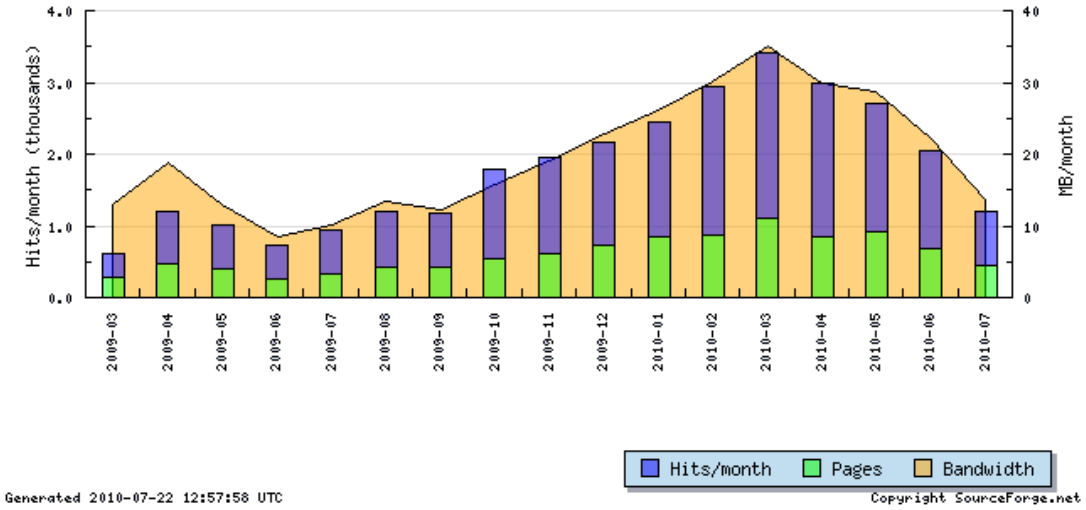


Figure E.2: pySTEP web traffic

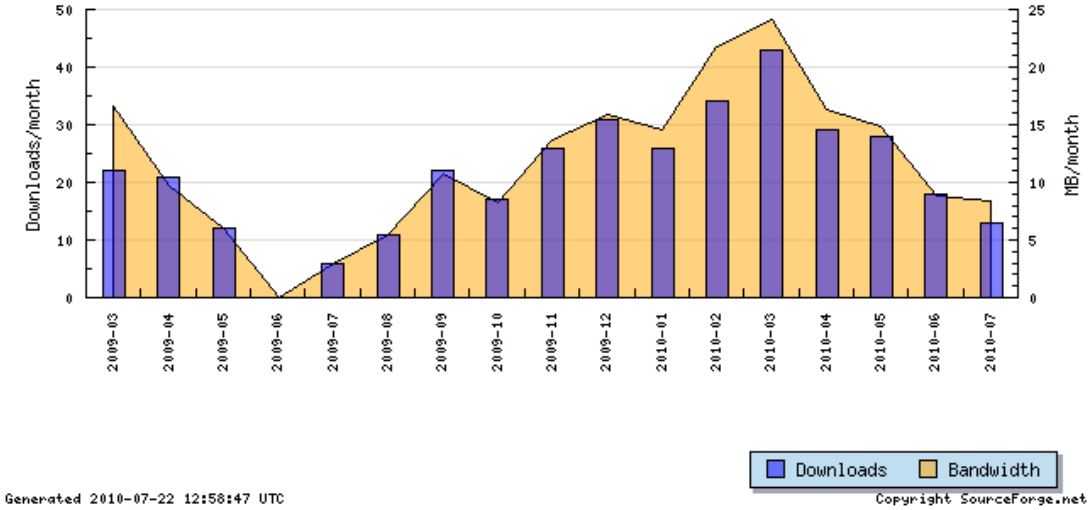


Figure E.3: pySTEP download history

Appendix F

Publications

- **Book Chapter**

  1 Khoury, M. and Liu, H. Using Fuzzy Gaussian Inference and Genetic Programming to Classify 3D Human Motions, *Robot Intelligence: An Advanced Knowledge Processing Approach, Springer, pp 95-116, 2009*, In Liu, Gu, Howlett and Liu (Eds.),

- **Conferences/workshops papers**

  2 Khoury, M. and Kubota N. and Liu, H. (in print) Optimizing Feature Selection Using Laplace Similarity in Occluded Human Motion Recognition, *Proceedings of IEEE World Automation Congress, International Forum on Multimedia and Image Processing, 2010*

  3 Khoury, M. and Liu, H. (in print) Recognizing 3D Human Motions using Fuzzy Quantile Inference, *Proceedings of Second International Conference on Intelligent Robotics and Applications - ICIRA, 2010*

4 Khoury, M. and Liu, H. (in print) Extending Evolutionary Fuzzy Qualitative Inference to Classify Partially Occluded Human Motions, *Proceedings IEEE International Conference on Fuzzy Systems, World Congress on Computational Intelligence, Spain, 2010*

5 Khoury, M. and Liu, H. Boxing Motions Classification through Combining Fuzzy Gaussian Inference with a Context-Aware Rule-Based System, *Proceedings IEEE International Conference on Fuzzy Systems, pp 842-847, Jeju island, Korea, 2009*

6 Khoury, M. and Liu, H. Fuzzy Qualitative Gaussian Inference: Finding Hidden Probability Distributions using Fuzzy Membership Functions, *Proceedings IEEE Workshop on Robotic Intelligence in Informationally Structured Space, pp 12-18, Nashville, USA, 2009*

7 Khoury, M. and Liu, H. Classifying 3d human motions by mixing fuzzy gaussian inference with genetic programming, *Proceedings of Second International Conference on Intelligent Robotics and Applications - Lecture Notes in Artificial Intelligence, Springer, Volume 5928, pp 55-56, 2009*

8 Khoury, M. and Liu, H. Mapping Fuzzy Membership Functions to Normal Distributions to Understand Boxing Motions, *Proceedings UK National Workshop on Computational Intelligence, pp 1-6, Sept 10-12, Leicester, UK, 2008*

- **Submitted Work (Journal Article)**

9 Khoury, M. and Liu H. , Inferencing Constrained 3D Human Motion, *IEEE Transactions on Fuzzy Systems*.