Laban Movement Analysis:
A Bayesian Computational Approach to Hierarchical Motion Motion Analysis and Learning

PhD Dissertation
Luís Carlos Santos
Coimbra, September 2013
Laban Movement Analysis:
A Bayesian Computational Approach to
Hierarchical Motion Analysis and Learning

Luís Carlos Santos
Coimbra, September 2013
This dissertation is realized under Supervision of

**Professor Doctor Jorge Manuel Miranda Dias**
Professor at University of Coimbra
This thesis is dedicated to my family and friends.
Abstract

The development of intelligent and autonomous robots is envisioned as a breakthrough in science and is expected to have significant impact in the future of our society. Scientists estimate that in the future, robots will be able to coexist in society with humans. A wave of intelligent robots will be capable to autonomously perceive the environment, follow a set of contextual social rules, and make their own decisions into fulfilling several tasks, relieving them from humans. Moreover, a set of cognitive capabilities will allow them to reconfigure themselves to new tasks, learn new actions, reason about unknown reactions and learn new social rules. The challenge is that robots have to cope with a limited prior knowledge about themselves and the environment, while operating without supervision in an uncertain world. To address these challenges, currently robotic system as developed towards specific scenarios, within well defined environments and properties. This fact, makes it difficult for them to deal with unknown situations, where its perception is subject to uncertainty and noise. In our thesis, we present a set of novel methodologies and concepts that enable robots to comprehensively interpret human motion and extend such knowledge via hierarchical analysis of different types of information. The proposed methods in this thesis address the following main three topics: (1) Defining a model which can robustly infer different types of information from human motion, using a generalizable grounding language; (2) Encoding the unique expressive properties of each person’s motion, so as to develop action invariant motion signatures towards a person recognition framework; (3) Develop a system’s action memory, which can store and retrieve action generalized information towards incrementally learn new actions and executing them to solve a task in has respectively.

This thesis starts by presenting an innovative approach to hierarchical analysis of human motion, based on a descriptive motion language, Laban Movement Analysis. This allows a system to infer multiple levels of information, from dynamic characteristics to intentions or behaviour patterns, by observing the 3D trajectories, generated from a given motion instance. Then, we exploit the outcome of Laban qualities classification into encoding this information to develop individual motion profiles. Such
characteristics are then applied to develop a Bayesian-based action invariant person recognition framework. The two aforementioned techniques are then integrated and adapted to develop an intelligent video-surveillance framework, showing to be capable of robustly recognize actions and person identities. The last part of our work focuses on developing a set of cognitive skills, allowing the system to build its own memory, by either learning new actions or incrementally fuse newly performed actions to existing knowledge.

All methods have been developed using probabilistic learning and inference, more specifically, Bayesian methodologies. They have been implemented and thoroughly evaluated using cross-validation procedures and different kinds of experimental scenarios so as to allow withdrawing conclusions based on produced evidences. Results demonstrate a highly robust and precise framework, whose main characteristics are flexibility, scalability and adaptability, showing to be useful to increase perception capabilities of artificial systems and have the potential to make significant impact in our future economy and society.
Resumo

O desenvolvimento de robôs inteligentes e autónomos é previsto como o grande salto científico e espera-se que tenha um impacto significativo no futuro da nossa sociedade. Cientistas estimam que no futuro, os robôs sejam capazes de coexistir em sociedade com seres humanos. Uma onda de sistemas inteligentes será capaz de percepcionar o ambiente e, de uma forma autônoma, seguir um conjunto de regras sociais, tomar a suas próprias decisões para executar diferentes tarefas, libertando-as do encargo dos humanos. Um conjunto de funções cognitivas irá permitir que eles se reconfiguram para novas tarefas, aprendam novas ações, e racionalizar sobre as suas reações e aprendizagem de novas regras sociais. O grande desafio é que os robôs têm de lidar com conhecimento a priori limitado sobre eles próprios, bem como do ambiente onde se inserem, enquanto operam sem supervisão num mundo onde reina a incerteza. Para lidar com estes desafios, actualmente os sistemas robóticos são desenvolvidos em função de cenários específicos, dentro de ambientes cujas propriedades são bem conhecidas. Este facto, torna difícil para estes sistemas lidarem com situações desconhecidas, incerteza e ruído. Nesta tese, apresentamos um conjunto de novas metodologias e conceitos que permitem aos robôs interpretar movimento humano de forma compreensiva, e estender esse conhecimento via análise combinatória a outros tipos de informação. Os métodos propostos nesta tese focam-se sobre os seguintes tópicos: (1) definir um modelo que consiga robustamente inferir diferentes tipos de informação, usando para isso uma linguagem descritiva de movimento, com capacidade de ser generalizável; (2) Codificar as características expressivas únicas, com o objectivo de gerar assinaturas de movimento invariantes à actividade para aplicação num sistema de reconhecimento de pessoas; (3) Desenvolver um sistema de memória, com capacidade de guardar e devolver informação generalizada de movimento para aprendizagem incremental e execução de ações respectivamente.

Esta tese começa por apresentar uma aproximação inovadora a análise hierárquica de movimento, baseada num linguagem descritiva de movimento, Análise de Movimento de Laban. Este método permite ao sistema inferir diversos níveis de informação,
desde características dinâmicas a intenções ou padrões de comportamento, através da observação das trajectórias no espaço Cartesiano Tridimensional geradas a partir do movimento de partes do corpo humano. De seguida, exploramos o resultado da classificação do símbolos de Laban, que é codificada para o desenvolvimento de perfis de movimento individuais. Estas características são aplicadas para o desenvolvimento de um sistema Bayesian de reconhecimento de pessoas, invariante à actividade executada. As duas metodologias anteriormente descritas são adaptadas e integradas num sistema de vídeo-vigilância inteligente, mostrando ser capaz de, robustamente, reconhecer acções bem como a identidade do executante. O último tema abordado neste trabalho foca o desenvolvimento de um conjunto de funções cognitivas, permitindo a um sistema artificial a construção da sua própria memória através da aprendizagem de novas acções, bem como integrando incrementalmente novas performances de acções já conhecidas.

Todos os métodos desenvolvidos, usam aprendizagem e inferência probabilísticas, mais especificamente, metodologias Bayesianas. Os métodos foram implementados e cuidadosamente avaliados usando um processo de validação cruzada e testados em diferentes cenários experimentais para que as conclusões sejam suportadas por evidências concretas. Os resultados atingidos demonstram a robustez e alta precisão dos métodos propostos, cujas características principais são a escalabilidade, flexibilidade e adaptaibilidade, mostrando serem úteis para aumentar a capacidade cognitiva de sistemas robóticos, com potencial impacto económico e social significativo.
Acknowledgment

I would firstly to express my deepest gratitude to my supervisor Professor Doctor Jorge Manuel Miranda Dias, for his commitment in making me a better researcher, for the fruitful discussions and for his personal and academical support during this period.

I also thank the support of the Institute of Systems and Robotics (ISR) at the University of Coimbra (UC) and for granting me the conditions to develop the work in this thesis. My acknowledgements also go to all my laboratory colleagues and friends for their support, friendship and discussions for the most part of this period. Most of them also assisted in the development of the UC-3D Motion Database. They are the Mobile Robotics Laboratory team during these past years, which are (without any special order of appearance): David Portugal, Pedro Trindade, Ricardo Martins, Micael Couceiro, Diego Faria, Hadi Aliakbarpour, Kamrad Khoshhal, José Prado, Amílcar Ferreira, Bruno Patrão, Christiana Tsiourti, Jafar Hosseini, João Quintas, João Filipe Ferreira, Luís Almeida, Paulo Freitas, Nuno Dias, Paulo Drews Jr., Pedro Machado, Rita Catarino, José Sousa, João Santos, João Martins, Gonçalo Augusto, André Araújo, Amadeu Fernandes, José Pereira, Nuno Ferreira and Luís Costa. I would also like to apologise in case I had forgotten anyone, but surely you had a part in this process too. I extend my gratitude to Doctor Jorge Sales, from University Jaume-I, Spain, with whom I had a fruitful collaboration during a 3 month stay.

Most of all, I would like to thank my parents, family and girlfriend, for their support and patience to endure my constant unavailability in the course of this work. Without your comprehension and motivation, it just would not be possible. This thesis is dedicated to you.

I thank also the members of the jury, for the time and commitment on the evaluation of this thesis, as well as the anonymous reviewers of my scientific papers for the constructive comments and corrections. I would like to thank the FCT-Fundaçao para a Ciência e a Tecnologia- for supporting my work with the Grant "SFRH/BD/65935/2009".
Contents

Abstract i

Resumo iii

Acknowledgment v

1 Introduction 1

1.1 Aims and Key Contributions .................................................. 3
1.2 Thesis Outline ................................................................. 4
1.3 List of Publications ........................................................... 5
1.4 Collaborations ................................................................. 7
1.5 Symbols, Notation and Acronyms ............................................. 8

2 Fundamentals 11

2.1 Action Recognition Modelling .............................................. 11
2.2 Probabilistic Reasoning ...................................................... 15
2.3 Bayesian Programming ....................................................... 19
2.4 Signal Processing and Feature Generation ................................. 20
2.5 Motion Data Processing ....................................................... 25
2.6 Movement Notations .......................................................... 26
2.7 Summary ................................................................. 35
3 Activity Recognition and Hierarchical Analysis 37

3.1 Introduction 37

3.2 Motion Information Hierarchy 41

3.3 Multilayer Activity Model 45

3.4 Model Evaluation and Performance 50

3.5 Extended Experimental Set-up 54

3.6 Applicability 65

3.7 Conclusions and Discussion 66

4 Motion-Based Person Identification 67

4.1 Introduction 67

4.2 Activity Invariant Symbolic Space 72

4.3 Laban Signatures 73

4.4 Methods for Person Identification 79

4.5 Experiments 83

4.6 Conclusions and Future Work 91

5 Case Study: Intelligent Video-Surveillance 93

5.1 Introduction 93

5.2 Visual Cues and Variables 95

5.3 Visual Laban Model 98

5.4 Application on Person Recognition 103

5.5 Conclusions and Discussion 107

6 Cognitive Skills for Action Learning and Synthesis 109

6.1 Introduction 109

6.2 Cognitive Skills Model 112

6.3 Categorical Skill Memory 115
List of Figures

1.1 Thesis Structure .................................................. 4
2.1 Bayesian Program Template ........................................ 19
2.2 Benesh Movement Notation ........................................ 27
2.3 Sutton Dance Writing notation .................................... 28
2.4 Laban Movement Analysis’ Components ............................ 29
2.5 LMA: Space Component ............................................ 30
2.6 LMA: Space Component’s Symbols ................................ 31
2.7 LMA: Effort Component ............................................ 32
2.8 LMA: Body component ............................................. 33
2.9 LMA: Shape Component ............................................ 33
2.10 Labanotation ....................................................... 34
3.1 Action Recognition and Hierarchical Motion Analysis Framework ........................................ 39
3.2 Bayesian Program for the Laban Movement Analysis Sub-Model ........................................ 46
3.3 Bayesian Program for the Activity Sub-Model .................... 47
3.4 Experimental Set-up Diagram ....................................... 48
3.5 Impact of Signal Noise in Model Convergence and Precision ........................................ 50
3.6 Impact of Feature Selection in Model Convergence and Precision ........................................ 52
3.7 Topologies Considered for Evaluating Error Propagation ........................................ 53
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.8</td>
<td>Impact of Node Connectivity in Model Convergence and Precision</td>
<td>53</td>
</tr>
<tr>
<td>3.9</td>
<td>MRL Motion Database (acquired within FP6 BACS European Project)</td>
<td>55</td>
</tr>
<tr>
<td>3.10</td>
<td>Examples from the KTH, WZ and UTI Databases</td>
<td>55</td>
</tr>
<tr>
<td>3.11</td>
<td>Laban Movement Analysis Qualities Classification Precision</td>
<td>56</td>
</tr>
<tr>
<td>3.12</td>
<td>Experimental Results on MRL Database</td>
<td>57</td>
</tr>
<tr>
<td>3.13</td>
<td>Experimental Results on KTH Database</td>
<td>58</td>
</tr>
<tr>
<td>3.14</td>
<td>Experimental Results on Weizmann Database</td>
<td>59</td>
</tr>
<tr>
<td>3.15</td>
<td>Experimental Results on UTI Database</td>
<td>60</td>
</tr>
<tr>
<td>3.16</td>
<td>Similarity of LMA Symbolic Description for 1 Action and 3 Different Actors</td>
<td>62</td>
</tr>
<tr>
<td>4.1</td>
<td>Person Recognition Framework Diagram</td>
<td>70</td>
</tr>
<tr>
<td>4.2</td>
<td>Topological Approach to Signature Encoding of Motion Expressive Properties</td>
<td>74</td>
</tr>
<tr>
<td>4.3</td>
<td>Directed Acyclic Graphs for 2 Person Recognition Methods</td>
<td>79</td>
</tr>
<tr>
<td>4.4</td>
<td>Bayesian Program for the Parametric Signature Model</td>
<td>80</td>
</tr>
<tr>
<td>4.5</td>
<td>Bayesian Program for the Compressed Signature Model</td>
<td>81</td>
</tr>
<tr>
<td>4.6</td>
<td>University of Coimbra 3-D Motion Database Overview</td>
<td>84</td>
</tr>
<tr>
<td>4.7</td>
<td>Confusion Tables for Person Recognition on UC-3D Database</td>
<td>85</td>
</tr>
<tr>
<td>4.8</td>
<td>Impact of Gait versus Non-gait Actions in Person Recognition</td>
<td>86</td>
</tr>
<tr>
<td>4.9</td>
<td>GMM-based Person Recognition Classifier Precision</td>
<td>87</td>
</tr>
<tr>
<td>4.10</td>
<td>Impact of Gait versus Non-gait Actions in GMM-based Person Recognition</td>
<td>87</td>
</tr>
<tr>
<td>4.11</td>
<td>Person Recognition Classification Precision in the 2-D KTH Database</td>
<td>88</td>
</tr>
<tr>
<td>4.12</td>
<td>Person Recognition Classification Precision in the 2-D WZ Database</td>
<td>89</td>
</tr>
<tr>
<td>5.1</td>
<td>Intelligent Video-Surveillance Framework Diagram</td>
<td>95</td>
</tr>
<tr>
<td>5.2</td>
<td>Static and Dynamic Visual Cues</td>
<td>97</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>----------------------------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>5.3</td>
<td>Visual Cues Experimental Indicators</td>
<td>98</td>
</tr>
<tr>
<td>5.4</td>
<td>Bayesian Program for the Vision-based Laban Movement Analysis Model</td>
<td>99</td>
</tr>
<tr>
<td>5.5</td>
<td>Example of Laban Symbolic Classification</td>
<td>103</td>
</tr>
<tr>
<td>5.6</td>
<td>Results for Person Recognition on KTH Database</td>
<td>105</td>
</tr>
<tr>
<td>5.7</td>
<td>Results for Person Recognition on KTH Database</td>
<td>106</td>
</tr>
<tr>
<td>5.8</td>
<td>Person Recognition Model Convergence Times</td>
<td>106</td>
</tr>
<tr>
<td>6.1</td>
<td>Cognitive Framework</td>
<td>111</td>
</tr>
<tr>
<td>6.2</td>
<td>Simplified Memorization Block Diagram</td>
<td>112</td>
</tr>
<tr>
<td>6.3</td>
<td>Action Encoding Process Using Gaussian Mixture Models</td>
<td>113</td>
</tr>
<tr>
<td>6.4</td>
<td>Synthesized function using Gaussian Mixture Regression technique</td>
<td>115</td>
</tr>
<tr>
<td>6.5</td>
<td>B-Tree Memory Structure</td>
<td>116</td>
</tr>
<tr>
<td>6.6</td>
<td>Generalized and Unknown action GMMs</td>
<td>117</td>
</tr>
<tr>
<td>6.7</td>
<td>Visualization of Fusion with Bag of Trajectories</td>
<td>119</td>
</tr>
<tr>
<td>6.8</td>
<td>Execution and Incremental Learning Block Diagram</td>
<td>120</td>
</tr>
<tr>
<td>7.1</td>
<td>Development Strategy with Increasing Scales of Complexity</td>
<td>125</td>
</tr>
<tr>
<td>7.2</td>
<td>Underwater Intervention Scenario</td>
<td>128</td>
</tr>
<tr>
<td>7.3</td>
<td>Proposed Log-spherical Space for Manipulation Phase Division</td>
<td>129</td>
</tr>
<tr>
<td>7.4</td>
<td>UWSim Underwater Manipulation Simulator</td>
<td>130</td>
</tr>
<tr>
<td>7.5</td>
<td>Bayesian Program for Inferring over Environment and Manipulator Characteristics</td>
<td>133</td>
</tr>
<tr>
<td>7.6</td>
<td>Directed Acyclic Graph for the Bayesian Classifier Responsible for the Interpretation</td>
<td>134</td>
</tr>
<tr>
<td>7.7</td>
<td>UWSim Operation and Feedback</td>
<td>135</td>
</tr>
<tr>
<td>7.8</td>
<td>Example of Learned Probabilistic Distribution</td>
<td>136</td>
</tr>
<tr>
<td>7.9</td>
<td>Knowledge Fusion Experimental Results using Cross Validation</td>
<td>138</td>
</tr>
</tbody>
</table>
7.10 Global system block diagram, encompassing acquisition, interpretation, memory and execution stages. ................. 139

A.1 Envelope function for the growth percentage. When $x \rightarrow \infty$ then $y \rightarrow 100\%$ 152

B.1 Different data types in the UC-3D Database. .................. 156

B.2 Different actions in the UC-3D Database. .................... 157
List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Summary of Different Approaches to Action Recognition</td>
<td>14</td>
</tr>
<tr>
<td>2.2</td>
<td>Separability Ratio Between States of Variables A and B</td>
<td>24</td>
</tr>
<tr>
<td>2.3</td>
<td>Effort qualities and their subjects</td>
<td>31</td>
</tr>
<tr>
<td>3.1</td>
<td>Effort and Shape Qualities Properties</td>
<td>44</td>
</tr>
<tr>
<td>3.2</td>
<td>Topological Indicators of Similarity</td>
<td>63</td>
</tr>
<tr>
<td>3.3</td>
<td>Dominant Laban Qualities for Actions from all Databases</td>
<td>64</td>
</tr>
<tr>
<td>4.1</td>
<td>Activity-based Person Recognition Benchmark</td>
<td>68</td>
</tr>
<tr>
<td>4.2</td>
<td>Summary of Implemented Effort and Shape Qualities</td>
<td>71</td>
</tr>
<tr>
<td>4.3</td>
<td>Separability Criterion for the Proposed Encoding Approaches</td>
<td>78</td>
</tr>
<tr>
<td>4.4</td>
<td>Experimental results summary</td>
<td>90</td>
</tr>
<tr>
<td>5.1</td>
<td>Dominant Laban Qualities for Actions from all Databases</td>
<td>102</td>
</tr>
<tr>
<td>7.1</td>
<td>Confusion Table for State Classifications</td>
<td>136</td>
</tr>
<tr>
<td>7.2</td>
<td>Global precision per symbolic variable</td>
<td>136</td>
</tr>
<tr>
<td>A.1</td>
<td>Summary of implicit signal rules. N/R = Not relevant</td>
<td>150</td>
</tr>
<tr>
<td>B.1</td>
<td>Actions in the UC-3D Database, example and description</td>
<td>158</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

Interpreting each other’s actions is a fundamental aspect of human life and also a key research area in psychology and cognitive science. Social psychologists have been researching the dimensions of social interaction for decades and found out that social signals strongly determine human behaviour. Most of these signals are consciously produced, in the form of spoken language. However, besides the spoken words, behaviour also involves non-verbal elements which are extensively and, mainly, unconsciously used in human communications. The non-verbal communication is conveyed as wordless messages, in parallel to the spoken words, through aural cues (voice quality, speaking style, rhythm, intonation) and also through visual cues (gestures; body language or posture; facial expression and gaze). These non-verbal signals are used to predict human behaviour, mood, personality, and social relations, in a very wide range of situations. It has been shown that, in many social situations, humans can correctly interpret non-verbal signals, reasoning and learning about theirs and each other’s behaviours with high accuracy [PA]. In speech analysis and recognition, a grounding language exists, via the use of word dictionaries or basic units, such as syllables. Contrary to the field of human motion analysis, which is still lacking a generic underlying modelling language, a statement by Moeslund and Granum [MG01], which still holds its meaning in the present days. To such intent, this thesis exploits a notation language for movement description, Laban Movement Analysis (LMA), which defines a comprehensive symbolic space, to describe different aspects of human motion. The human brain can perceive signals which are interpreted with the same meaning, using different available senses. It is the brain’s function to associate such signals to meaningful descriptors and reason over them, formulating what it estimates to be the correct interpretation. In the proposed model, the perceived information is based on body part trajectories in Cartesian space. The incompleteness and subjectiveness of human motion lead a viable solution
to be approached from a non-deterministic perspective.

Human motion can be generalizable, but ultimately every person has its own way of moving. Similarities may exist and are a desirable property when developing a model to recognize the same actions from different actors. However, we aim at exploiting the comprehensiveness of LMA, into finding those slight expressive differences and characterize each person’s unique motion characteristics. In fact, we demonstrate the capability of LMA’s generalization into describing human movements, while simultaneously using it to develop a set of motion signatures. These capture and enhance characteristics allowing to differentiate between different persons.

Cases exist where the actions known by the artificial system may not be sufficient to fulfill some, or any, of its tasks. Most action models have limited space states and are built into a system memory, which is learned a priori under some sort of supervised training. We propose a set of cognitive skills, which allow a robot to autonomously interpret observed actions and environment characteristics. This symbolic description is used into developing a categorical action memory, which can expanded by either memorizing new actions, or incrementally refine ones already existing. This so called memory, much like what happens with humans, will be probed for retrieving adequate knowledge, which can then be synthesized into producing the adequate actions.

In summary, this thesis addresses three main challenges:

- **Action Recognition and Hierarchical Analysis**: The first addressed problem in this thesis, concerns the development of a probabilistic model, encompassing the ability to provide different types of information, inferred from the observation of human motion, more specifically, from body part trajectory signals.

- **Person Recognition using Activity Invariant Motion Signatures**: The second main problem addresses the development of motion signatures, which encode a person’s unique expressive motion characteristics, using them in a classifier towards the discrimination between different observable persons.

- **Cognitive Skills for Learning and Synthesizing Actions**: The final main problem, focuses on generating a set of cognitive skills, which upon an initial stage of supervised training by demonstration, allow the system to autonomously interpret, memorize and synthesize new actions.

The proposed work carries an underlying wide range of applications, which can have significant impact in the future of our society: improve Human-Robot Interaction
and Human-Machine interfaces, intelligent surveillance systems and monitoring (e.g. elderly care centres) improving algorithms to provide better predictions of potentially dangerous situations, reflecting either in social aspects as well as quality of life, automatic analysis of psychotherapeutic sessions, or find application in the current wave of new generation gaming using mobile phone cameras to provide control input, amongst others. There are two case studies selected to evaluate the models which are proposed in this thesis, in real life challenging conditions.

- **Intelligent Video-Surveillance System**: The first is a surveillance scenario, where we demonstrate the movement analysis, action recognition and person identification capabilities of our model.

- **Autonomous Underwater Vehicle for Intervention Missions**: The second will demonstrate we can give a robot the ability to autonomously develop its own action memory. Moreover, such memory can be used in future tasks, in which actions are retrieved and synthesized into solving specific tasks, within the context of underwater manipulation robotics.

### 1.1 Aims and Key Contributions

The presented work follows a structured and correlated set of works, where each chapter functions as research motivation for the next. From the developed research and solutions to the key problems, the following main contributions emerge:

**Main Contributions**

1. In the area of activity recognition, we present a flexible, scalable Multilayer Hierarchical Bayesian-based classifier, which shows state of the art precision. We have showed our computational implementation of Laban Movement Analysis to be generalizable, such that the our framework does not need to be re-trained for classifying different datasets.

2. The generated Laban symbolic analysis, showed to be discriminant with respect to whom was performing actions, which allowed the development of motion signatures, which capture these discriminant properties, exploited in the development of an activity invariant person identification framework.
3. We have developed a cognitive skills model, which allows a system, to autonomously perform inference over its own actions and use such information towards building and extending its own action memory.

1.2 Thesis Outline

This document is divided in as many chapters as the number of relevant contributions made within this research scope. Each chapter has an introduction, problem statement, related work overview and concludes with a discussion of achieved results and future work. In most cases, subsequent chapters are a natural consequence of identified future work and applicability of research from preceding chapters. These dependencies are depicted as a Directed Acyclic Graph, (Figure 1.1) where each node represents a chapter and directed arrows represents the dependencies, inspired in the theoretical representation of the dominant methodology throughout this thesis. In Chapter 2 we introduce the fundamental concepts, which are felt needed for a full comprehension of this work and which are not sufficiently detailed in posterior chapters.

The multilayer Laban-based model for activity analysis and recognition is presented in Chapter 3, where LMA’s generalization capabilities are exploited and demonstrated. Laban symbolic space showed discriminant combinations of Laban components, which are used to identify different persons based how the way they move in Chapter 4. Some

![Figure 1.1: Thesis structure represented as a Bayesian Network.](image-url)
challenges are identified in order to make the previous research applicable on Intelligent Video Surveillance Systems, which are addressed in Chapter 5. This research allowed extending LMA model to visual features, something that had not been yet explored in the current state of art. Our scientific research is concluded with a study on how actions can be autonomously learned and/or synthesized into generating reproducible movements in Chapter 6 for which we present a case study in the scope of underwater robotics in Chapter 7. This thesis concludes with the final remarks and highlights of the present research in Chapter 8.

1.3 List of Publications

In this section, the articles published in the scope of this thesis are enumerated. We divide this list in Peer-review Journals and Conferences, identifying also some works where the candidate has made scientific and technical contributions, as a collaborator whose contribution is explicitly acknowledged in the cited manuscripts.

Peer-reviewed International Journals

- Luís Santos and Jorge Dias. “Cognitive Functions for Autonomous Learning and Synthesis of Motion Activity”, Journal to be selected, ongoing work

As research assistant:


**Peer-reviewed Conference Proceedings**


• Jorge Sales, **Luís Santos**, Pedro J. Sanz, Jorge Dias and Javier C. GarcÃ­a. "Increasing the Autonomy Levels for Underwater Intervention Missions by using Learning and Probabilistic Techniques”, in Robot 2013: First Iberian Robotics Conference, in Madrid, Spain, November, 2013

• **Luís Santos**, José Sousa and Jorge Dias. "Vision-based Motion Signatures for Person Identification”, 2nd workshop on Recognition and Action for Scene Understanding (REACTS) 2013, satellite event of the 15th International Conference of Computer Analysis of Images and Patterns (CAIP), 2013.


• **Luís Santos** and Jorge Dias. “Hierarchy and Reversibility in human motion modelling”, *Workshop on Recognition and Action for Scene Understanding*, 2011

• **Luís Santos** and Jorge Dias. Oral presentation on “Introduction to Hierarchy and Reversibility in human motion modelling: a Bayesian approach”, *Workshop Operational Research in Robotics*, 2011

• **Luís Santos** and Jorge Dias. “Motion Patterns: Signal Interpretation Towards the Laban Movement Analysis Semantics”, *Doctoral Conference on Computing, Electrical and Industrial Systems (DOCEISS’11)*, 2011
1.4 Collaborations

- **Luís Santos** and Jorge Dias. “Laban Movement Analysis : Towards Behaviour Patterns”, *Doctoral Conference on Computing, Electrical and Industrial Systems (DOCEIS’10)*, 2010

- **Luís Santos** and Jorge Dias. “Human motion classifier based on Laban Movement Analysis”, *4th International Conference on Cognitive Systems, (COGSYS)* 2010


- **Luís Santos** and Jorge Dias. “Human-Robot Interaction: Invariant 3-D Features for Laban Movement Analysis Shape Component”, *14th IASTED International Conference on Robotics and Applications (RA)*, 2009

- José Prado, **Luís Santos** and Jorge Dias. “A Technique for Dynamic Background Segmentation using a Robotic Stereo Vision Head”, *RO-MAN 2009, 18th IEEE International Symposium on Robot and Human Interactive Communication*, 2009

- José Prado, **Luís Santos** and Jorge Dias. “Horopter based Dynamic Background Segmentation applied to an Interactive Mobile Robot”, *14th International Conference on Advanced Robotics (ICAR)*, 2009


1.4 Collaborations

During this thesis, there have been some collaborations with other people. I started my work integrated in an FP6 European Project named Bayesian Approaches to Cognitive Systems (BACS), whose collaborative research provided valuable and fundamental concepts about Bayesian probability. As a complement to my research as a PhD student, I supervised a master dissertation and took part in a collaboration between the
Mobile Robotics Laboratory at the Institute of Systems and Robotics, University of Coimbra and the Computer Science and Engineering Department, from University of Jaume-I, Castellón, Spain. Chapter [5] is an extension of the master dissertation of José Sousa [Sou13] on Motion-based Person Recognition System. The development of a cognitive function models towards the development of an Autonomous Underwater Vehicle for Intervention Missions (Chapter [7]) is integrated in the scope of a Spanish project, for which my host institution has received a guest professor, Jorge Sales, with the purpose of sharing our expertise in Probabilistic Learning and Classification methodologies. During the 3 month stay, we have jointly published two peer-reviewed conference papers and on still under peer-review process.

1.5 Symbols, Notation and Acronyms

Symbols and Notations used in this manuscript are presented in following table. The following table describes various abbreviations and acronyms used along this thesis and their respective significance.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ω</td>
<td>Database</td>
</tr>
<tr>
<td>ω</td>
<td>trajectory segment</td>
</tr>
<tr>
<td>x</td>
<td>scalar</td>
</tr>
<tr>
<td>x</td>
<td>column vector</td>
</tr>
<tr>
<td>³x</td>
<td>estimation of x</td>
</tr>
<tr>
<td>p(x)</td>
<td>probability distribution of a random variable x</td>
</tr>
<tr>
<td>p(x</td>
<td>y)</td>
</tr>
<tr>
<td>N(µ, Σ)</td>
<td>normal distribution with mean µ and covariance Σ</td>
</tr>
<tr>
<td>U(x)</td>
<td>uniform distribution of a vector x</td>
</tr>
<tr>
<td>f_b ∈ F</td>
<td>Trajectory feature</td>
</tr>
<tr>
<td>c_n ∈ L</td>
<td>Laban Component variable</td>
</tr>
<tr>
<td>τ_n ∈ χ</td>
<td>Laban Space dimension</td>
</tr>
<tr>
<td>R = [τ_1, ..., τ_n]</td>
<td>Coordinate in Laban Space</td>
</tr>
<tr>
<td>Π = [R_1, ..., R_t]^T</td>
<td>Set of Laban Space Coordinates</td>
</tr>
<tr>
<td>Λ</td>
<td>Action variable</td>
</tr>
<tr>
<td>γ_r ∈ Γ</td>
<td>Signature variable</td>
</tr>
<tr>
<td>Θ_p</td>
<td>= Mixture Signature Variables</td>
</tr>
<tr>
<td>{θ_p,k,µ_p,k,Σ_p,k}</td>
<td>No. of Components in the Mixture</td>
</tr>
<tr>
<td>k</td>
<td></td>
</tr>
<tr>
<td>p ∈ ζ</td>
<td>Identity variable</td>
</tr>
<tr>
<td>A,M</td>
<td>Generic Matrices</td>
</tr>
</tbody>
</table>
### 1.5. Symbols, Notation and Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>BDN</td>
<td>Dynamic Bayesian Network</td>
</tr>
<tr>
<td>DAG</td>
<td>Directed Acyclic Graph</td>
</tr>
<tr>
<td>DFT</td>
<td>Discrete Fourier Transform</td>
</tr>
<tr>
<td>GMM</td>
<td>Gaussian Mixture Model</td>
</tr>
<tr>
<td>GMR</td>
<td>Gaussian Mixture Regression</td>
</tr>
<tr>
<td>HMM</td>
<td>Hidden Markov Model</td>
</tr>
<tr>
<td>KLD</td>
<td>Kullback-Leibler Divergence</td>
</tr>
<tr>
<td>KLT</td>
<td>Karhunen-Loève Transform</td>
</tr>
<tr>
<td>LLE</td>
<td>Local Linear Embedding</td>
</tr>
<tr>
<td>LMA</td>
<td>Laban Movement Analysis</td>
</tr>
<tr>
<td>LOOCV</td>
<td>Leave-One-Out Cross-Validation</td>
</tr>
<tr>
<td>MAP</td>
<td>Maximum A Posteriori</td>
</tr>
<tr>
<td>MLE</td>
<td>Maximum Likelihood Estimation</td>
</tr>
<tr>
<td>PCA</td>
<td>Principal Component Analysis</td>
</tr>
<tr>
<td>SVD</td>
<td>Singular Value Decomposition</td>
</tr>
<tr>
<td>SVM</td>
<td>Support Vector Machine</td>
</tr>
</tbody>
</table>
Chapter 2

Fundamentals

This chapter presents an overview of fundamental concepts and techniques that have been applied in our research. Its main objective is to introduce the key concepts, and a brief overview of some of the mathematical principles applied throughout our research. In depth details to some of the present methods will require further reading, for which adequate references are provided. We will cover the theoretical foundations of our work such as Laban Movement Analysis, Bayesian Programming and Inference, as well as methods that have been applied in signal processing and feature generation.

2.1 Action Recognition Modelling

Activity recognition can be seen as a classification process of a sequence of motion data, of variable size, whose attributes change along time, that is, the comparison between each observed sequence and some reference of representative sequence classes. This comparison is reliable, only if the evaluation tolerates the existence of noise in the data. However, this similarity evaluation is highly dependent of the chosen model to represent and then classify this data.

Known approaches in action recognition, can be broadly divided into 4 different categories: (1) deterministic models, (2) discriminative models, (3) descriptive and (4) generative stochastic models. The next subsections summarize the main properties of each category.
2.1.1 Deterministic Models

Deterministic models, encompass very simple theoretical concepts. Because of that, they do not allow any type of learning, forcing them to be user-defined, which requires previous and exact knowledge of the motion we intent to observe. Park et al. [PPA04] modelled human behaviour as sequences of state changes that represent the configuration and the movement individual body parts in spatio-temporal space. Their method finds events or state changes while reading the model body motion data, while checking for known sequences. However, given the finite number of states, every observation must fall into one of them. Moreover, deterministic modelling fails to provide generalization, as different persons may have different ways of performing the same action. Increasing the space state to new action or observation variables would exponentially augment their cardinality, turning it into an intractable approach. However, their work provided some encouraging results in a structure-wise analysis, and demonstrated that hierarchical structures are a suitable method to organize and model human motion. Another relevant conclusion also states that the defined states are not enough to cover all aspects of human motion.

2.1.2 Discriminative Models

Discriminative Models usually map in a direct way, a input vector into a symbol belonging to a set of possible symbols. This requires the definition of boundaries for each region associated to distinct symbols. In real work applications, however, the regions cannot be perfectly separated by decision boundaries. The overlap of different classes is frequently verified. Class separation is given by a function which maximizes some criteria. Neural Networks are a popular solution, however they require large amounts of training data, in order to build models adequate for real world applications. The use of discriminative methods in literature, [Bra99, MJ02, SVD03, RS02, AT04b, AT04a, EL04] aims to estimate the state conditional directly, in order to simplify inference. They are typically supervised methods, which use a set of data $\tau = \{(r_i, x_i|i = 1...N}\}$, from the 3D human configurations $x$ paired with their correspondent 2D image appearance $r$, and focus on modelling this association.

Inference, on the other hand, involves missing data and does not always require supervision. But this process can be complex, because modelling perceptual data often produces highly multi-modal distributions [SBS02, SJ04a, SJ04b]. While this strictly implies that inverse mapping from observations to states is multi-valued and can-
not be functionally approximated, several methods aimed to do so \cite{SVD03, MJ02, AT04a, AT04b, AS03, TSA03}. Some authors constructed data structures for fast nearest-neighbour retrieval \cite{SVD03, MJ02, AS03, TSA03} or learned regression parameters \cite{AT04a, AT04b, EL04}. The inference process can: use nearest neighbour indexing for locally weighted predictions; directly predict from the learned regressor parameters \cite{AT04a, AT04b, EL04}; or perform affine reconstruction from joint centers \cite{MJ02, LC85, Tay00}. Among discriminative methods, a notable exception is \cite{RS02}, who clustered a dataset into soft partitions and associated them to learned functional approximations (e.g. perceptrons or regressors). However, clusterwise functional approximation \cite{QR72, DW88, RS02} is only going halfway towards a multivalued inversion, because inference is not straightforward. The problem is that the model represents the joint distribution and not the conditional. Therefore, for new inputs, cluster / perceptron membership probabilities cannot be supervisory computed, because the state is missing. Nevertheless, clusterwise regression \cite{QR72, DW88, RS02} is useful as a proposal mechanism, e.g. during generative inference based on quadrature-style Monte-Carlo approximations and indeed this is how it has been primarily used \cite{RS02}. A related method has been proposed by \cite{KGT03}, where a mixture of probabilistic PCA is fitted to the joint distribution represented as silhouette features in multiple views paired with their 3D pose. Reconstruction is based on MAP estimates, in which the state conditional could be unimodal, but missing data makes inference non-trivial.

It has been argued that discriminative models can provide fast inference and can interpolate flexibly in the trained region. However, they can fail on novel inputs, especially if trained upon small datasets. Increasing complexity inevitably leads to multimodal state conditionals and learning such distributions is difficult, as most exciting methods \cite{SVD03, AT04a, AT04b, EL04, TSA03, KGT03} are unimodal. Finally, discriminative methods lack a clear probabilistic temporal estimation framework that has been so fruitful with generative models \cite{IB98, DBR00, ST03, SB01}.

\section*{2.1.3 Descriptive Stochastic Models}

Descriptive stochastic models, allow the learning sequences without the need of supervision, focusing on the data’s intrinsic structure and their relations, creating a model to represent its properties. There is virtually no literature nor applications for Descriptive Stochastic Models in human motion analysis or synthesis. They fail at capturing motion’s temporal characteristics, making them hard to apply in systems which intend to perform recursive estimations.
2.1.4 Generative Stochastic Models

Generative Stochastic Models, allow to create a statistical model of predictive behaviour. In this approach, inference is influenced by a set of variable factors. For that, the existence of a hidden set of variables is assumed, organized in an unknown way, responsible for the generation of observed data. The challenge of generative learning consists in the identification of the variables and in the way these relate. The success of generative models depends on the capability of acquiring the structure of the inherent phenomenon to the observations. Some of these techniques include Hidden Markov Models and Bayesian Networks. Bayesian theory gives us the possibility to deal with incompleteness and uncertainty, make predictions on future events and, most important, provides an embedded scheme for learning. Included in the Bayesian framework are specialized models which have a long tradition in many areas. Some examples of these models are Hidden Markov Models (HMMs), Kalman Filters and Particle Filters. Bayesian models have already been used in a broad range of technical applications (e.g. navigation, speech recognition, etc.). In areas closely related to the field of gesture recognition, these models have proven their usability [Sta95, Pav99, Ret09]. Recent findings indicate, that Bayesian models can also be useful in the modelling of cognitive processes. Research on the human brain and in its computations for perception and action report that Bayesian methods have proven successful in building computational theories for perception and sensorimotor control [KP04].

2.1.5 Summary

Our research allowed concluding that Bayesian methodologies can fulfil the two main areas addressed in this thesis: action recognition/analysis and a computational function for perception and sensorimotor control. In Table 2.1, we summarize some of the techniques and properties of each category.

Table 2.1: Table summarizing the different approaches, references, pros and cons of each approach.

<table>
<thead>
<tr>
<th>Category</th>
<th>Technique</th>
<th>Hierarchy</th>
<th>Multimodal</th>
<th>Predictive</th>
<th>Learning</th>
<th>Uncertainty</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deterministic</td>
<td>e.g. Finite State Machines</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>Discriminative</td>
<td>e.g. Neural Networks</td>
<td>NO</td>
<td>YES</td>
<td>NO</td>
<td>YES</td>
<td>YES</td>
</tr>
<tr>
<td>Descriptive</td>
<td>e.g. Maximum Likelihood</td>
<td>NO</td>
<td>YES</td>
<td>NO</td>
<td>YES</td>
<td>NO</td>
</tr>
<tr>
<td>Generative</td>
<td>e.g. Bayesian</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
</tr>
</tbody>
</table>
The analysed parameters encompass the possibility to implement hierarchy, in the sense of whether or not they are easily scalable. How easy is it to retrain or add new variables is verified in the multi-modal column. Only the Generative approaches provide an intuitive scheme of recursive computation for predictive behaviours. All algorithms support learning, however generative provide an embedded scheme, which allows for computationally efficient online learning. The last parameter addresses each category’s ability to deal with uncertainty in the observable data.

2.2 Probabilistic Reasoning

We can divide probabilistic reasoning into frequentist or evidential approaches, also known as objectivist and subjectivist respectively. The first loosely defines probability as the limit of an event’s relative frequency in a large number of trials, in a context where experiments are random and well defined. In the case of the latter, is often associated with Bayesian inference, in which probabilities can be assigned to any statement, even in the presence of non-random processes. In this thesis, we will focus on a subjectivist approach, where Bayesian Inference is a way to represent a degree of belief in a statement, or given evidence. This discussion will continue with a presentation of relevant properties and parameters of Bayesian methods.

2.2.1 Bayes Rule

Bayes theorem is the most popular representation of conditional probability. It has its foundations on two basilar probability rules, the general conjunction rule and marginalization. It establishes a probability of an event $A$, given the knowledge of evidence $B$. In a mathematical notation, it is formulated as follows,

$$P(A|B) = P(A) \frac{P(B|A)}{P(B)} \quad (2.1)$$

From equation (2.1), we can identify different arguments:

- $P(A)$ is the set of prior distributions for parameter set $A$ and represents uncertainty about $A$ before data is taken into account.

- $P(B|A)$ is the likelihood distribution, which related variables via a probability model.
• $P(A|B)$ is the joint posterior distribution, expressing belief about parameter $A$, taking into consideration the prior and the data.

• The term $P(B)$ is the (constant) normalization factor. It is a result of the marginalization rule and its sole purpose is to ensure the posterior integrates to "1". Generally, it is often omitted in the inference formulation, for simplicity purposes.

The following subsections present a brief introduction over the first three arguments, as the fourth simply represents a constant factor.

2.2.1.1 Prior Distribution

The prior distribution $P(A)$, as the name implies, represents the degree of personal belief before any evidence is known. It Bayes theorem, it is multiplied by the likelihood, hence affecting the outcome of the posterior. Its presence is what distinguishes frequentist from subjectivist approaches. Moreover, it provides regularization, that is, it prevents over-fitting and introduces additional information in order to solve ill-posed problems, guaranteeing that a solution exists. Classically, there are two types of prior distributions, informative and uninformative prior. Recently there have been works presenting four categories of prior distributions.

• Informative Priors should be used when prior information is available. For example, in cases where the model form is similar to previous versions and the current model is intended to be an updated version, but in this case, based on more recent data, then the previous model should be used as prior distribution for the present.

• Weakly Informative Priors are used mainly for regularization, which given enough prior information, prevent results from contradicting the available knowledge or algorithmic inference solutions which are unable to explore the space state. This prior type provides the benefit of using prior information without taking the risk of using information that does not exist. The fact this prior is not flat, it prevents numerical approximations of becoming stuck in regions of flat density, as opposed to frequentist approaches, which consider flat priors and become stuck more frequently. When using this prior approach, one must examine the posterior, ensuring if does not contradict knowledge, or the prior should be revised into becoming consistent with knowledge.
2.2. Probabilistic Reasoning

- In the case of Least Informative Priors, their goal is to minimize the amount of subjective information and use a prior which is solely based on the model and observed data. A popular example is the flat prior, represented by a uniform distribution. One other example is the hierarchical prior, where the parameters of this distribution, are themselves modelled through the observed training data.

- Some of the distributions now categorized as least informative, have been traditionally associated to Uninformative Priors. However, these do not truly exist and all priors are informative in a way.

2.2.1.2 Likelihood Distribution

The term $P(B|A)$ is known as the likelihood distribution, and is a probability density function of $B$, given the knowledge of the outcomes of $A$. It represents the way evidence affects the state of $A$, usually emerging from experimental data or strong knowledge of the event to be modelled. The likelihood principle states that inference from data to hypothesis, depends on how likely is the actual data under competing hypothesis, and not based on what could have been seen. Two probability models with the same likelihood yield the same inference for the same variable. It is very common to associate likelihood and probability. However, in probability theory, these are two different concepts. Probability is the process which allows to predict unknown outcomes based on known parameters, contrary to Likelihood, which allows estimating unknown parameters based on known outcomes. The latter, measures how the evidence affects the posterior and can be seen as a reversed version of conditional probability. It represents a probabilistic model of variable relations and is usually based on training data associated via a learning process.

2.2.1.3 Posterior Distribution

The posterior probability $P(A|B)$ represents the degree of belief in the state of variable $A$, after new evidence $B$ is available. It quantifies the uncertainty of a given event and is computed based on Bayesian Inference, and is a result of the prior and likelihood distributions. The solution maximizing the posterior distribution is usually achieved through Bayesian Inference.
2.2.2 Bayesian Inference

Bayesian Inference used Bayes Rule to update the belief for a hypothesis given new evidence. The objective of inference is finding the value of a random variable, which maximizes a given posterior distribution, based on estimation theory.

- *Maximum-Likelihood Estimation (MLE)* is used to find an estimate for a random variable, maximizing the likelihood function of the data. This is a point estimate process, which is very exposed to over-fitting. Also a common occurrence are zero probability events, that are events for which no evidence has been observed. A popular solution is using Laplacian estimator, or Rule of Succession [Lap14], which in the presence of zero probability events, adds one count to each of the possible states. Moreover, using MLE does not guarantee solution uniqueness. The process of computing MLE has the goal of finding the value for $\theta$, such that,

$$\Theta_{MLE} = \arg\max_{\theta} P(data|\theta) \quad (2.2)$$

where $P(data|\theta)$ is a must have expression representing the likelihood model. This approach is associated to frequentist approaches.

- A method, which is likely the most probable Inference technique in Bayesian Inference is the *Maximum A Posteriori (MAP)*. This is, naturally, the Inference method used in this thesis. It is used to choose a value of $\theta$ providing a good solution for the available data, which maximizes the posterior function given a sample of data. It is a derivation of the MLE, with the different it uses the prior distribution. By applying the conjunction rule, MAP is seen as the MLE biased by the influence of the prior distribution.

$$\Theta_{MAP} = \arg\max_{\theta} P(\theta|data) = \arg\max_{\theta} P(data|\theta)P(\theta) \quad (2.3)$$

The term $P(data|\theta)$ represents the likelihood distribution, whereas $P(\theta)$ the prior. There are several techniques to compute MAP, such as, numerical estimation, iterative and Monte Carlos methods or analytically, if the problem has closed-form, that is, if we have a finite number of known functions which analytically express the joint distribution.
2.3 Bayesian Programming

The Bayesian models in our work are partially implemented using a probabilistic programming API, ProBT\footnote{http://www.probayes.com/index.php/en/products/sdk/probt}, supported by an intuitive formalism, allowing an efficient model development, oriented towards its implementation. Bayesian Programming can be applied to develop models in multiple areas, from email spam filters to robotics \cite{BAMM12}, or even cognitive bio-inspired models \cite{CDB10} for programming using GPU CUDA Processing \cite{JFFD11}. The main elements for specifying a probabilistic model are the relevant variables, their conditional dependencies which are represented by probability distributions, and associated parameters. A Bayesian program is divided in two main phases:

1. a \textit{description} which is the probabilistic model of the studied phenomenon;
2. a \textit{question}, which represents the inference problem to be solved by the model.

The \textit{description} itself is also divided two-fold:

1.a) The \textit{specification} formalizes the knowledge of the person who is developing the model;

\begin{itemize}
\item \textbf{Variables:} define the relevant variables for our model. They encompass observable information, state estimation.
\item \textbf{Decomposition:} Factoring a joint distribution into a product of simpler distributions, which exploit conditional dependency.
\item \textbf{Formulation:} Assign different parametric, or non-parametric probabilistic functions which best suit each of the elements of the decomposition.
\item \textbf{Identification:} Bernoulli, Binomial parameters based on the data.
\item \textbf{Question:} Inquire the model for information about a variable. The answer is given by the posterior probability.
\end{itemize}

Figure 2.1: Bayesian Program template, where each stage of the process is briefly described.
(1.b) The *identification* is where the parameters are learned from sets of training data or, alternatively heuristically defined upon expert knowledge.

The final step is the *specification* having three stages:

(1.a.i) Selecting *variables* to model the phenomenon;

(1.a.ii) The following stage is the *decomposition*, where the joint distribution on the relevant variables is decomposed as a product of simpler conditional probability distributions representing their dependencies;

(1.a.iii) Each distribution is, at the *formulation* step, represented by a parametric mathematical function. Densities are assigned taking into consideration the variable types.

Upon having the complete Bayesian Program, we may pose the *question*. The model can be queried for information about a given variable, which is answered using Bayesian Inference, generally applying the *Maximum A Posteriori* method [SB12, FLB+13]. Figure 2.1 shows the template structure of a Bayesian Model implementation under Bayesian Programming formalism. Given the complete description of the model, Bayes formalism is applied to make questions, which are answered by the posterior probability via Bayesian Inference.

### 2.4 Signal Processing and Feature Generation

In this thesis we approach the classification process using a sliding window paradigm, which is used to acquire a sample of the raw signal, during a period of time \( t \). We have formulated a method for an adaptive version of this process, in Appendix A. The data within the window is usually of high dimension, reason for which we need to have alternative representations of lower dimensionality. Different methods to perform such dimensionality reduction are presented in the immediate subsections. We conclude this section with a key definition of for this work, trajectory variables. The core of feature generation is find alternative representation for observable data. This transformation is usually expected to reveal meaningful information about the original signal. There is a wide range of methods and categories. Since implementing and testing all methods is an intractable task, we have selected four, all of them representing a different methodology class.
2.4.1 Karhunen-Loeve Transform

The computation of the Karhunen-Loève (KL) transformation matrix will exploit the statistical information describing the data. The first assumption is that the data values have zero mean. The goal is to obtain mutually uncorrelated features in an effort to avoid information redundancies. The method computes the data correlation matrix, which by its symmetric properties generates a set of mutually orthogonal eigenvectors $V$, known as the KL transform. As it turns out, KL has a number of other important properties, which provide different ways for its interpretation. One is the actually generated orthogonal eigenvectors, which encompass the principal directions of the spanned data, as well as the variance along each its directions. Thus we will use this information to represent trajectories in the resultant component space. We decided to use this information rather than the original purpose of the KL (re-project data in a dimensional space smaller the original), because data reduction methods are not optimized regarding class separability, and they do not assure that the principal components provide the best discriminatory properties. KL transform, is a widely recognized technique, and further information on this method and its properties can be found in [Jol02].

2.4.2 Local Linear Embedding

The starting point of this method is the assumption that the data points lie on a smooth manifold (hyper-surface). The main philosophy behind Local Linear Embedding [TK09] is to compute a low-dimensional representation of the data however preserving the local neighbourhood information. The outcome of this algorithm attempts to reflect the geometric structure of the data. This algorithm can be resumed in its basic form with the following three steps:

1. Select the nearest neighbors for each of the data points $x_i, i = 1, 2, \cdots, n$. Some common techniques are Euclidean distances or the K-nearest neighbors.

2. Compute the weights $W(i,j)$ that best reconstruct the point $x_i$ from its neighbors minimizing the cost function

$$\arg min_{W} E_W = \sum_{i=1}^{n} ||x_i - \sum_{j=1}^{n} W(i,j)x_{j,i}||^2$$

(2.4)
A typical weight functions is given by the following equation.

\[
W(i,j) = \begin{cases} 
\exp\left(-\frac{||x_i-x_j||^2}{\sigma^2}\right), & \text{if points correspond to neighbours} \\
0, & \text{otherwise}
\end{cases}
\]  

(2.5)

where \( \sigma^2 \) is a user-defined parameter. The weights are constrained such that the rows of the weight matrix, i.e., the sum of the weights over all neighbours equals to 1.

3. Use the weights obtained from the previous step to compute the corresponding points \( y_i \in \mathbb{R}^m, i = 1, 2, ..., n \), to minimize the cost with respect to the unknown points \( Y = \{y_i, i = 1, 2, ..., n\} \).

\[
\text{argmin} E_y = \sum_{i=1}^{n} ||y_i - \sum_j W(i,j)y_j||^2
\]  

(2.6)

This method explores the local linearity of the data and tries to predict each point through its neighbours using the least squares error criterion. Minimizing the cost regarding to the constraint given in step (2) results in a solution that satisfies the following interesting properties: Scale, rotation and translation invariance.

Solving (3) for the unknown points \( y_i, i = 1, 2, ..., n \), is equivalent to:

- Performing the eigen-decomposition of the matrix \( (I-W)^T(I-W) \).
- Discarding the eigenvector corresponding to the smallest eigenvalue.
- The remaining eigenvectors corresponding to the other eigenvalues yield the low-dimensional outputs \( y_i, i = 1, 2, ..., n - 1 \).

### 2.4.3 Discrete Fourier Transform

The Discrete Fourier Transform (DFT) transforms a time series into a sum of functions that represent it in the frequency domain. There is an assumption that the signal must be finite, which is accomplished in our case due to signal nature. The aim of this technique is to quantify how much of the signal lies in a determined frequency, i.e. to determine the dominant frequencies in a signal. We have tested a classification
2.4. Signal Processing and Feature Generation

model using the dominant frequencies and their coefficients to define the feature space state.

2.4.4 Seven Moments of Hu

Under the scope of geometric moments, which are used to characterize data such as areas or information about orientation, we have the known 7 moments of Hu [TK09]. Within this class of methods, we have opted for Hu’s moments because this technique intrinsically encompasses invariance to rotation, translation and scale. These are important properties because of the assumption that trajectory contours can be performed at different scales and orientations or space, depending on the physical structure of performer. The moments of Hu base themselves in the definition of central moments

\[ \mu_{pq} = \int \int (x - \bar{x})^p (y - \bar{y})^q f(x, y) dx \, dy \]  

(2.7)

which are then normalized. We will not describe the mathematics of Hu’s 7 moments, as they are somewhat cumbersome to this article and are readily available in [TK09] for the interested reader. An important remark is the statement that the first six moments are also invariant under the action of reflection, while the seventh moment changes signal. This property is interesting in the sense that it allows both left and right handed performers to be considered indifferent in terms of generated data. The values of these quantities can be quite different. In practice, to avoid precision problems, the logarithms of their absolute values are usually used as features.

2.4.5 Algorithm Benchmark

We have tested the separability of each of the presented methods with real motion data, in a classification space whose variables could have one of two possible states. To establish a comparison criterion of class separability, a method based on Scatter Matrix (SM) was applied. Other methods such as Divergence or Bhattacharyya Distance turn to be computationally demanding if a Gaussian assumption of data distribution is not employed. Scatter Matrices are built upon information related to the way feature vector samples are scattered in the l-dimensional space. The method defines the following matrices:

\[ S_W = \sum_{i=1}^{n} P_i \Sigma_i \]  

(2.8)
Which is known as within class scatter matrix, and $\Sigma_i$ is the covariance matrix for class $w_i$ and $P_i$ is the a priori probability of class $w_i$, i.e. $P_i \sim n_i/N$, where $n_i$ is the number of samples of class out of a total $N$ samples. The Between-class scatter matrix is defined as

$$S_b = \sum_{i=1}^{M} P_i(\mu_i - \mu_0)(\mu_i - \mu_0)^T \quad (2.9)$$

where $\mu_0$ is the global mean vector. The simplified computation for the Mixture scatter matrix turns out

$$S_m = S_W + S_b \quad (2.10)$$

with $S_m$ the covariance matrix of the feature vector with respect to the global mean. Its trace$^*$ is the sum of variances of the features around they global mean. From these definitions, the criterion formulates as

$$J_1 = \frac{\text{trace}\{S_m\}}{\text{trace}\{S_W\}} \quad (2.11)$$

where the ratio $J_1$ takes large values when samples are well clustered around their mean and the clusters are well separated.

From the observation of results in Table 2.2, the conclusion is that there is not a single method outperforming all others in both variables. Different variables exhibit better separability for different methods. However, this test shows Principal Component Analysis to exhibit the best global performance.

Table 2.2: Separability ratio between states of variables A and B, for each of the presented techniques.

<table>
<thead>
<tr>
<th></th>
<th>Variable A</th>
<th>Variable B</th>
</tr>
</thead>
<tbody>
<tr>
<td>KLT</td>
<td>246.7</td>
<td>210.2</td>
</tr>
<tr>
<td>DFT</td>
<td>36.6</td>
<td>29.1</td>
</tr>
<tr>
<td>LLE</td>
<td>190.3</td>
<td>229.6</td>
</tr>
<tr>
<td>Hu</td>
<td>143.2</td>
<td>143.3</td>
</tr>
</tbody>
</table>

$^*$Trace is defined as the sum of the elements on the main diagonal.
2.5 Motion Data Processing

Let a random person perform an activity motion sequence \( \omega_\alpha \). We use a sliding window of length \( l \), defining the size of a classifiable trajectory sub-segment \( \hat{\omega}_\alpha \), such that:

\[
\hat{\omega}_\alpha = \begin{bmatrix}
Y_{t-l} \\
\vdots \\
Y_t
\end{bmatrix}, Y \in \mathbb{R}^3
\]  

(2.12)

In previous section we addressed the performance of multiple processing algorithms [OSB99, TK09, Jol02] with respect to their separability ratio [TK09]. It was concluded that there is no method which totally outperforms the others. However, a method based on the Karhunen-Loève Transform (KLT) [Jol02] presented the best average results, thus justifying its preference in this framework. This technique is applied to generate a lower dimension representation of trajectory subset \( \hat{\omega}_\alpha \), a vector whose elements define the initial hypothesis for variable sub-space \( \Pi \), defined as:

\[
\Pi = \{ F = \text{KLT}(\hat{\omega}_\alpha) : F = [X_1 X_2 X_3 \lambda_1 \lambda_2 \lambda_3] \}
\]  

(2.13)

composed of sorted eigenvectors \( X \in \mathbb{R}^3 : X = (x, y, z) \) and eigenvalues \( \lambda \in \mathbb{R} \), in which each element \( f_i \in F \) represents an independent, identically distributed feature variable.

**Remark 1:** For 2-D trajectories, the outcome of the KLT is \( F = [X_1 X_2 \lambda_1 \lambda_2], X \in \mathbb{R}^2, \lambda \in \mathbb{R} \).

2.5.1 Feature Selection

To mitigate the degenerative classification effect caused by some variables, vector \( F \) is pruned, discarding features \( f_i \) which exhibit reduced discriminant capabilities within the variable sub-space \( \Pi \). Feature Selection Toolbox (FST) [SP02] was developed by Somol et al. with the purpose of feature selection in statistical pattern recognition. We applied the following available criterion functions to measure inter-class distances: Battacharyya distance, Mahalanobis Distance, Generalized Mahalanobis, Fisher Discriminant Ratio and Divergence (for further details address [DK82]). For selecting the sub-set which maximizes the criteria functions, we applied a class of optimal algorithms, Enhanced Branch and Bound (EBB) and Fast Branch and Bound (FBB). Herein follows a short description of the feature selection procedure.

*http://fst.utia.cz
Consider $N$ samples $\omega_\alpha$ for different activities, which are annotated based on symbol subset $\pi$. We select $N_s \leq N$ samples, which are labelled upon the same word $v_s \in \pi$, such that:

$$T_s = \{\omega_\alpha\}, \forall \omega_\alpha \leftarrow v_s \in \pi$$

(2.14)
after which a feature vector for each sample is generated as $F = KLT(T_s)$. Using the aforementioned criteria, we measure feature inter-class distances $d_{i,\text{class}}$ between different classes $v_s \in \pi$. Posteriorly, Fast, and Enhanced Branch and Bound algorithms are applied to maximize $d_{i,\text{class}}$ in the classification space $\pi$. Variables $f_i \in F$ are selected based on a predefined $d_{i,\text{class}}$ threshold.

### 2.5.2 Trajectory Feature Variables

Trajectory features are represented by discrete variables, dividing $f_i$ into a $\kappa$ equidistant classes. Let us consider that originally:

$$f_i \in \mathbb{R}: f_i \in [a,b]$$

(2.15)

where $a$ and $b$ define the minimum and maximum values for a given $f_i$ for all available training samples. The $j^{th}$ class bin corresponds to a defined interval:

$$\text{class}(j) \in \left[a + (j-1)\frac{(b-a)}{\kappa}, a + j \frac{(b-a)}{\kappa}\right]$$

(2.16)

hence, defining the trajectory low level variable space state as $f_i = \{1, \cdots, \kappa\}$.

### 2.6 Movement Notations

Cognitive sciences establish an analogy between reasoning and data processing. The relation between observed properties is formulated in models reflecting evident logically correct assumptions under different circumstances. Thus reasoning appears as an important issue of cognitive processes, where its efficiency is a critical indicator of cognitive intelligence. Human motion and behaviour modelling are complex tasks. It takes years for persons to learn how to correctly assess each others behaviours. One can argue that the establishment of relations of simple observed features and high level concepts of motion and behaviour is hard to be done directly. To deal with this problem, we propose to describe behaviour using a movement notational system as a reasoning unifying framework, thus creating a symbolic support layer. This section
2.6. Movement Notations

will now introduce some well known movement notation systems, a theoretical repre-
sentation of human movements. Such representations have been studied and developed
throughout the years.

The Beauchamp-Feuillet notation is a system developed around 1680 and published
in 1700 as a program of notated dances [LM92]. Symbols representing foot position,
the step, actions, turns and rhythm where the basic symbols for this notation. In
the late 1940’s, one other notation was invented. Joan and Rudolf Benesh developed
a notational system to document any form of dance or human movement [BB83].
Resembling a music score, it is read from left to right with bar lines to mark the
passage of time. All information about body and limb positions is shown within a
five-line stave. Movement Lines trace the paths made by the extremities. Locomotion
Lines link the positions of the feet, showing whether the performer steps, jumps or
slides from one position to the next. Rhythm, phrasing and movement quality are
shown above the stave. Figure 2.2 illustrates this notation. The Dance Writing was
first developed in 1966 by Valerie Sutton and extended to a greater body of work
called Movement Writing [Sut82]. Dance Writing is a way to read and write any kind
of dance movement. A stick figure is written on a five-lined staff. Each line of the staff
represents a specific level. Figure 2.3a represents the different levels of the Sutton’s
Dance Writing. When the figure bends its knees or jumps in the air, it is lowered or
raised accordingly on the staff. The five-lined staff acts as a level guide. Figures and
symbols are written from left to right, notating movement position by position, as if
stopping a film frame by frame (Figure 2.3b).

The Eshkol-Wachman Movement Notation was developed by the choreographer
Noa Eshkol and architect Abraham Wachman [EW58]. It intents to represent not only
dancing movements but motion in general, finding application in animal behaviour

![Figure 2.2: Benesh Movement Notation](image-url)
analysis [Gol76]. Eshkol-Wachman notated the body as a sort of stick figure. The body is divided into its skeletal joints, where each pair represented a limb. The relationship of those segments is done over the three dimensional space using a spherical coordinate system. Movements are shown as transitions between initial and end coordinates. The Eshkol-Wachman Movement Notation represents a good descriptor for spatial positions and the kinematic chains are not limited to the human body alone.

The use of spatial/geometric descriptors is a common characteristic for all above mentioned notational systems. However, human motion encompasses additional information other than its geometry. The expressive content of movements is not addressed in any of the presented notational systems. Rudolf Laban (1879-1958) was the pioneer of Laban Movement Analysis. Currently used as one of the primary movement notation systems in dance, Laban’s work, Kinetographie Laban was published in 1928. Laban Movement Analysis (LMA) evolved throughout the years by Laban’s scholars to become a notational language for understanding, observing, describing and notating all forms of movement. Devised by Rudolf Laban, LMA draws on his theories of effort and shape to describe, interpret and document human movement. Used as a tool by dancers, athletes, physical and occupational therapists, it is one of the most widely used systems of human movement analysis. LMA is divided in components: Body, Space, Effort and Shape. What distinguishes this framework from others, is its ability to describe an additional expression that accompanies the spatial trajectory. This might be the key to retrieve some evidences about the emotional state or the intention of the performer. Anne Hutchinson-Guest compares and emphasizes these differences, when comparing LMA to thirteen other historical and present-day dance notation systems, pointing the advantages and disadvantages of each system [Gue89].
2.6. Movement Notations

2.6.1 Laban Movement Analysis

Laban Movement Analysis (LMA) was developed by Rudolf Laban (1879 to 1958) as a method for observing, describing, notating, and interpreting human movement. In 1980, Irmgard Bartenieff, a scholar of Rudolf Laban described LMA’s general framework in [BL80]. It is a widely applied framework to studies of dance and physical and mental therapy [BL80]. Recently researchers from neuroscience started applying LMA to describe certain effects on the movements of animals and humans. Foround and Whishaw adapted LMA to capture the kinematic and non-kinematic aspects of movement in a reach-for-food task by human patients whose movements had been affected by stroke [FW06]. It was stated that LMA places emphasis on underlying motor patterns by notating how the body segments are moving, how they are supported or affected by other body parts, as well as whole body movement. In the engineering domain, the most notable researches started with the group of Norman Badler [CCZB00, Zha02, ZB05], who had started in 1993 to formulate Labanotation in computational models [BPW93]. Recently, Rett J. [Ret09] et al. [RSD08, RDA08, RDA10] applied Laban’s framework to the Human-Robot Interaction domain with success.

The theory of LMA consists of several major components, though the available literature is not in unison about their total number. The works of Norman Badler’s group [CCZB00, Zha02] mention five major components shown in Fig. 2.4. *Relationship* describes modes of interaction with oneself, others, and the environment (e.g. facings, contact, and group forms). As *Relationship* appears to be one of the lesser explored components. Some literature [FW06] only considers the remaining four major components, something that this work will tend to follow as a guideline also. As suggested in [FW06], components will be divided in 2 groups. *Body* and *Space* as kinematic features describing changes in the spatial-temporal body relations, and *Shape* and *Effort* the non-kinematic features contributing to qualitative aspects of the movement (Fig. 2.4).
2.6.1.1 Space

Space treats the spatial extent of the mover’s Kinesphere (often interpreted as reach-space) and what form is being revealed by the spatial pathways of the movement. The Space component is probably the most important component to distinguish one movement from another. It presents the different concepts that allows the description of the trajectories emerging of human movements inside a frame of reference [BL80]. Space presents different concepts to express movements in the specified frame and all measures are relative to the anthropometry of the performer. Despite being different, all concepts are capable of being represented in the 3-D Cartesian system. Choreutics [Lab66] yields the following definitions that are different in some aspects from those given in Labanotation [Gue70]: Levels of space, Basic Directions, the three axes, the three planes and the Icosahedron. The Kinesphere describes the space of farthest reaches in which the movements take place. Levels and Directions can also be found as symbols in modern-day Labanotation [BL80]. Labanotation direction symbols encode a position-based concept of space. Recently, Longstaff [Lon01] has translated an earlier concept of Laban which is based on lines of motion rather than points in space into modern-day Labanotation. Longstaff coined the expression Vector Symbols to emphasize that they are not attached to a certain point in space. The different concepts are shown in Fig. 2.6. The symbols of Labanotation correspond to positions in space like Left-High while the Vector Symbols describe directions. Fig. 2.6b represents a 2-D view for one of the defined planes, thus showing only a partial set of symbols (8). It was suggested that the collection of Vector Symbols provides an heuristic for the perception and memory of spatial orientation of body movements.

Figure 2.5: The Space component defines several concepts: a) Levels of Space, Basic Directions, Three Axes, and b) Three Planes and Icosahedron.
2.6. Movement Notations

2.6.1.2 Effort

The Effort component describes the dynamic qualities of the movement and the inner attitude towards using energy. The ability to describe an additional expression accompanying the spatial trajectory, allows the possibility to retrieve some evidences considering the emotional state and intention of the performer. The Effort component consists of four Effort qualities: Space, Weight, Time, and Flow. Each of the qualities has an underlying cognitive process, a subject and two extremes each of them has \[ BL80 \]. The relations are shown in Table 2.3. The Movements are described and distinguished by those qualities that are close to an extreme, e.g. a Punch has Strong Weight, Sudden Time and Direct Space. When a movement has one quality lying between two extremes, it is considered to be neutral and it is often described that quality as not been observed by simply omitting it.

Combinations of three qualities, with the fourth considered to be neutral, are considered the most natural way to perform an action. Combinations of all four qualities close to an extreme rarely occur as they produce extreme movements (e.g. tearing

![Figure 2.6: Comparison of Labanotation and Vector Symbols for the Door Plane.](image)

Table 2.3: Effort qualities and their subjects

<table>
<thead>
<tr>
<th>Effort</th>
<th>Cognitive process</th>
<th>Subject</th>
<th>Extremes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Space</td>
<td>Attention</td>
<td>The spatial orientation</td>
<td>focused or non-focused</td>
</tr>
<tr>
<td>Weight</td>
<td>Intention</td>
<td>The impact</td>
<td>strong or light</td>
</tr>
<tr>
<td>Time</td>
<td>Decision</td>
<td>The urgency</td>
<td>urgent or non-urgent</td>
</tr>
<tr>
<td>Flow</td>
<td>Progression</td>
<td>How to keep going</td>
<td>free or careful</td>
</tr>
</tbody>
</table>
Figure 2.7: The bipolar Effort qualities of the Action Drive, i.e. Flow = neutral (omitted) represented as a cube. The position of the movement M (Point) indicates its qualities, i.e. direct, sudden and light.

something apart) \cite{BL80}. Also single-quality movements are rare \cite{BL80} and even for a trained Laban performer (i.e. Laban notator) difficult to perform \cite{Zha02}. Considering Flow, Space, Weight or Time to be neutral, the related combinations are designated Action Drive, Spaceless, Timeless or Weightless, respectively. The literature on Laban Movement Analysis like \cite{Zha02, BL80} often gives some exemplary movements. The concept of combining 3 Effort qualities allows a movement to be defined by its position in one of the four 3-D spaces. The Effort space is often modelled as a cube where each vertex represents an action. The edge length represent the distance between two extremes (e.g. sudden and sustained) (Figure 2.7). Movements with only two Effort qualities are called Incomplete or Inner States as they occur often during transitions between two three-quality combinations. They can also reflect a failure to produce a certain three-quality action (e.g. an attempt to perform a Punch fails due to weakness).

2.6.1.3 Body

The Body component of LMA deals with the question which of the body parts are moving and how their movement is related to the body center. It also addresses issues concerning locomotion and kinematics. In LMA the kinematic chains are observed with relation to spatial Shaping possibilities and the dynamic qualities (Effort) accompanying them \cite{BL80}. The center of the egocentric reference system is naturally located and the center of body, approximately at the navel (see Fig. 2.8). Other locations are known to be considered the center: the sternum, near the belly-button and in the pelvis \cite{Lon96}. The body component encompasses body parts located at the lower and upper halves of the body. More precisely hip, leg and feet which are mainly related to locomotion and head, arm and hands relating to exploration, manipulation and gesturing \cite{BL80}.
2.6. Movement Notations

Figure 2.8: The Body component defines which of the body parts are moving and how their movement is related to the body center, e.g. navel or sternum.

Figure 2.9: The bipolar Shape descriptors in each of the planes.

2.6.1.4 Shape

Irmgard Bartenieff defines Shape as a set of qualities that emerge from the Body and Space components. Shape is focused on the body itself or directed to a goal in space. This component is divided in two main qualities: (1) Flow describes the movement focused on the body itself, whether it is going towards or away from the body center. It uses descriptors like shrinking and stretching. (2) The term Spatial Shaping relates to movements that are focused on a goal in space, using descriptors like Reaching. It is usually described in a Euclidean frame of reference that is aligned with an initial position of the egocentric frame of reference. Due to this, movements can be described by using the vertical, horizontal and sagittal axes and relating them to bipolar descriptors like sinking and rising, enclosing and spreading, and retreating and advancing, respectively as shown in 2.9. In [Zha02] some example movements for the Shape component
Figure 2.10: Labanotation: a) The staff is used to place the symbols. b) The horizontal placement of the symbol indicates the body part. c) Shading of the symbol is used to indicate the Level (height) of the 3-D position. d) Different shapes of the symbols indicate the position in the Table Plane $\pi_v$.

can be found. Sometimes a third quality is mentioned in literature \cite{Zha02} that is described as carving or moulding when the body with the environment (e.g. moving through a crowd).

\subsection{Labanotation and Effort Notation}

The need to develop some means of recording for the perceptions of movements led to a notation system known as Labanotation. It is built of symbols which describe the structure and progression of the movement (shown in Fig. 2.10). The spatial definitions (see \cite{Gue70}) vary from those stated in Choreutics (see \cite{Lab66}).

In Labanotation the three Levels of Space are circular causing the distances e.g. centre..L and centre..LD to be equal. Moreover, distinct frames of reference are defined for the different groups of body parts. e.g. placing the origin of the arm-hand group at the shoulder joint. The symbols reflect which body part does what in space and time and with what kind of dynamic stress. In particular it contains when the movement starts and its duration. The so called Staff organizes the body parts in columns where the time proceeds from the bottom up along the length. The placement of a symbol shows that the body part is active, its shape indicates the direction of the movement, its shading shows the level and its length, the duration of the movement. From a properly notated movement sequence, the skilled reader can see at one glance what is happening at any moment in every part of the body.
2.7 Summary

This chapter has reviewed the main approaches in the field of computer based human motion analysis and recognition. The proposed models are based on theories of motion, more specifically, on a movement notation system, Laban Movement Analysis. An introduction to Bayes theorem, the core of the proposed models, is made, covering some of its key issues. Probabilistic graphical models are an intuitive way of representing dependencies between random variables, and apply very well to Bayesian methodologies. They are useful to assist the implementation of a Bayesian model under the formalism of Bayesian Programming. Very often, available inputs have an untreatable number of variables and data (curse of dimensionality [Bel57]), for which we present different reduction techniques for representing data as usable sets variables. Variable separability criteria is also addressed.
Chapter 3

Activity Recognition
and Hierarchical Analysis

3.1 Introduction

"Body language is a powerful source of information about human emotions and intentions." - de Gelder, 2007 [dG]. Humans constantly produce unconscious body motion signalling. These non-vocal signals are used to assess physical activity, behaviour, mood, personality and social relations in a variety of situations. Interpreting human motion is a relevant scientific research topic and a key concept in a wide range of applications, such as surveillance, monitoring, human-robot interfaces, activity/motion recognition or analysis in physiotherapy and sports, to mention some. Activity recognition focuses on the association problem, in which sets of discriminant features are related to (usually restricted) symbolic spaces, mostly developed within specific scenario contexts. This common approach has made it difficult for the development of a unified and universal motion language for computational applications. Moreover, implemented algorithms are seldom applied in conditions, other than the ones they were originally designed for. In this chapter, we propose a multilayer classification framework, based on an activity invariant language, which can be composed to define more complex information. Moreover, the proposed framework has the capability to infer different types of information simultaneously, where each layer may be modelled using different, adequate methodologies.
3.1.1 Related Work on Activity Recognition

The challenge of activity recognition is addressed by the community considering two different paradigms: (1) pattern and (2) model based algorithms. There are numerous techniques and methods applied within pattern approaches. Some works, in the research field of computer vision, compute optical flow [LB98, Bla99, EBMM03], sometimes revealing to be a complex task due to image requirements. Others involve feature extraction from a data stream of images [BN06, CNC03, CK96, HE04, SWZ08, SW09] categorizing movements from a geometric perspective. Gait analysis and similar actions [NC04, WSNK10] are target of keen research, whose algorithms are often restricted by periodicity. Exploring joint angles and/or body kinematics is another popular choice [UF04], which involves complex computations and signal acquisition is not trivial. Recently, approaches which address motion from a space-time perspective have also provided interesting results [GBS+07].

An alternative approach focuses on exploiting model properties, fitting a determined concept of motion, or motion data. Applied methodologies include Hidden Markov Models (HMM) [LSS+09, LCC10], Support Vector Machines (SVM) [FVN10] or Bayesian Networks (BN) [Ret09, KCPS08], to name some popular choices. The majority of existent works, postulate high level actions to be composed of simpler activities or sequences of basic actions, e.g. [MCB+01, PA04, HNB04, AWSR05]. However, the transversal application of the developed algorithms may prove to be a frustrating challenge, as the defined variable spaces and related features are specific to narrow scoped research scenarios. The selected decompositions are usually developed to fit specific contexts, such as sports-like [KCPS08] or basic every day actions like picking-up or dropping-off objects [LHdW07]. Moreover, most works target the identification of an activity, but seldom a comprehensive analysis of motion’s characteristics.

One objective of this research work, is presenting a model capable of symbolically describe random activity sequences, based on context-free and invariant descriptors. Within this scope, there is Laban Movement Analysis (LMA) [Dav06], a language which parametrizes any form of movement into basic motion units. LMA is a flexible language and its paradigm provides an intuitive way to combine its basic information into subsequent and more complex definitions, such as gestures or behaviour. Laban’s concept has been previously explored. Rett. et al [Ret09, SPD09] computationally implemented Laban to classify a reduced set of gestures within the context of Human-Robot Interaction. Targeting Kinesiology training, a study lead to the development of a model to analyse Laban Shape Quality [STM+09]. Laban as also been noticeably
Figure 3.1: General framework for the proposed approach: The artificial System observes a motion sequence, and queries information; An activity database emerges from observations, which upon previous knowledge allows generative learning (Bayesian Model); The system, based on Bayesian inference, analyses motion using natural language semantics; The Bayesian model is iterative and makes use of previous estimations.

explored by Norman Badler’s group, e.g. [Zha02, CCZB00], and recently by Khoshhal and Dias in the area of interactive behaviours [KD13].

3.1.2 Our Approach

Activity recognition is a very active research area, mainly focusing on image-based or kinematic joint information. Moreover, research shows existent analysis models to be limited to small actions sets. We propose to address the following unsolved limitations in current state of the art approaches:

- Descriptor sets are limited and specific to determined research contexts.
- Most approaches depend on specific model characteristics and/or processing algorithms, jeopardizing their scalability and applicability.
- The expressive, invariant qualities of motion are seldom analysed, and may provide augmented information sets for extending their applicability.
- Activities are very subjective and may need large amounts of training data, for a single activity, to present accurate classifications for large groups of different persons.

A multilayer model to analyse body language using different symbolic information layers simultaneously, is developed to further extend a previous research [SD11b].
The overview of the proposed framework is presented in Figure 3.1. We consider a trajectory-based analysis, thus creating a fair degree of device abstraction, since these signals can be acquired, for example, using Motion Capture devices or image-based segmentation. Trajectories are preprocessed using the Karhunen-Loève Transform generating a reduced, representative feature set. These features are evaluated with respect to their discriminant capabilities in the classification spaces, and pruned using a class of optimal algorithmic procedures. The multilayer activity model is developed under a unified Bayesian framework, which may integrate independently developed sub-models, based on different Bayesian-based algorithms. A simulated-based performance analysis to the multilayer model is done, measuring to what extent the hierarchical topology affects accuracy and speed. To deal with different performance styles for the same actions (the generalization problem), we propose a symbolic information hierarchy based on an activity invariant symbolic language, Laban Movement Analysis. Laban descriptors can be differently combined into describing more complex activities and/or behaviours, a property which is reflected in our model structure. The experimental set-up is extended to acknowledgeable and publicly available datasets, where ground truth activity annotation is compared against the model beliefs. Analysis is performed based on different layers of symbolic descriptors, simultaneously, and our results directly compared to state of the art approaches. The selected classification approach shows real-time performance capability. From our proposed solution to the aforementioned challenges, we identify the following main contributions:

- A highly flexible, generalizable and scalable motion analysis classification framework,

- which, based on Laban Movement Analysis, provides an activity invariant symbolic analysis, describing motion’s expressive qualities,

- presenting the capability to analyse different motion information levels simultaneously.

- Laban models also show to be generalizable, without the need to be retrained to classify different motion datasets.
3.1.3 Problem Statement

Let $\Omega$ be a database with annotated motion sequences $\omega_{\alpha}$, where different trials for a given activity $\Lambda = \alpha$ are performed by different persons.

$$\Lambda = \{\alpha_1, \cdots, \alpha_m\}$$  \hspace{1cm} (3.1)

A motion sequence consists of a trajectory in Cartesian space yielding

$$\omega_{\alpha} = \begin{bmatrix} Y_1 \\ \vdots \\ Y_t \end{bmatrix}, Y \in \mathbb{R}^3$$  \hspace{1cm} (3.2)

where $t$ indexes each sample $Y$, acquired at a suitable frequency to capture human activities, for which in our experimental set-up, we consider $f = 40Hz$ [BSKK07]. Consider a hierarchic dictionary defined as $\Pi = \{\Pi_1, ..., \Pi_k\}$ with $k$ different abstraction layers such that:

$$\pi_{k,p} \in \Pi_k : \pi_{k,p} = \{v_1, ..., v_s\}$$  \hspace{1cm} (3.3)

represents the $p^{th}$ symbolic subset, with $s$ different symbols $v$. Let a random person, perform an activity which belongs to the existent database. Learning a symbolic model of human motion is defined as the problem of parametrizing motion sequences into different symbols and finding their hierarchical relations, combining them towards describing more complex activities and/or behaviours. For each symbolic representation layer, the challenge is to define an independent model, which can be posteriorly integrated in the global, unified classification framework. The analysis process is defined as the problem of automatically associating different symbols $v$ to a sub-segment of $\omega_{\alpha}$, based on the previously learned activity models, in order to provide a continuous, semantic description of the observed activity.

3.2 Motion Information Hierarchy

Movement is the result of the release of energy through muscular response to a stimulus, producing a body response in space and time, from which different types of information can be withdrawn. A comprehensive interpretation of motion is defined in this chapter, as a problem of information hierarchy, in which sets of basic descriptors combine themselves into describing complex activities, through an adequate symbolic
grammar. In Figure 3.1 a pyramidal representation (right) attempts to illustrate our proposed paradigm. On the right side, we enumerate possible dimensions (variable spaces), whereas the left shows examples of states for each dimension. Basic descriptions are provided at the bottom, whereas complexity grows on upper layers of the pyramid. We categorize variable spaces considering their ability of being reproducible into natural language (InforMotion∗) or not (Primary). The basis for InforMotion variables is defined upon LMA [BL80, Gue70], an abstract and context independent representation. Subsequent space states, e.g. gestures or behaviours, emerge as combinations of hierarchically inferior variables. At the primary layer, trajectory segments are represented through lower dimensional feature sets, as described in previous section 2.3. Our pyramidal grammar for motion description is developed based on the concepts of movement notation.

3.2.1 The Concept of Movement Notation

"The process of recording movement (...) involves the conversion of elements of space, time, energy and the parts of the body involved into symbols which can be read and converted into movement." [Gue70]. We can infer from Hutchinson’s statement that movement can be described as a composition of different elements. She identifies three kinds of descriptions from which a comprehensive understanding of body movement emerges:

- **Motif Description** provides a general statement concerning the most salient features of a movement. It also encodes the aim or intention.

- **Effort-Shape Description** investigates movement with respect to its dynamic content. Effort refers to energy and Shape to expressiveness and functional value.

- **Structural Description** addresses clearly defined and measurable terms, such as: which parts move, direction or movement duration, to enumerate some.

In this chapter, we take the liberty to further extend the aforementioned concept to other levels beyond gestures. People, in their daily routine, infer a lot of information from observing body language. Studies reveal that by observing and reasoning (combining) small details or dynamics in body language, persons usually make assumptions on context or behavioural aspects of a scene (e.g. [EW02]).

*INFORmation from MOTION.*
3.2.2 Laban Movement Analysis and Labanotation

Several systems have been proposed to annotate motion [Gue89], but supported by theories of effort and shape [Gue70], Laban Movement Analysis (LMA) provides a unique capability of describing qualitative aspects and expressiveness of movement. Laban defines movement as an intentional process of patterned and orderly changes, which can be better studied if approached at multiple information levels [MY88]. Originally, it defines five components, Body, Effort, Shape, Space and Relationship, each addressing specific properties of movement [BL80], and represented through an adequate symbolic grammar, Labanotation [Gue70].

- **Body**: Describes structural and physical properties of the moving human body.
- **Effort**: Addresses the dynamics, the way the movement is performed with respect to inner intention.
- **Shape**: Studies the connections between body and space, and body shape.
- **Space**: Focuses on spatial patterns and body part pathways.
- **Relationship**: The less studied component, it addresses the relations between a person with the surroundings.

These are regarded as the most basic elements needed to comprehensively describe human motion activities. Each individual has its own way for combining these components according to its cultural, personal and artistic preferences [Zha02]. However, these sequences can be symbolically generalized for activity description [BL80].

3.2.3 Hierarchical Variable Sub-Spaces

3.2.3.1 Activity Invariant Symbolic Representation

Laban’s theory of movement, states “*Gesture... is any movement of any body part in which Effort and Shape elements or combinations can be observed*” [BL80]. The previous statement, justifies these two components to define Laban space state, $\Pi_2$ (symbolically represented by $L$). Additionally, if these characteristics are always observable in any movement, it is reasonable to assume that, as variables, they are invariant to the performed activities. Effort is divided into four different qualities: Space, Weight, Time and Flow [CCZB00]. Combining three qualities is the most natural way of performing
Table 3.1: Effort Time and Shape qualities, cognitive process, subject, dimensions, associated planes and space state description. Acronyms Comp. and Var. stand for Component and Variable respectively.

<table>
<thead>
<tr>
<th>Comp. Quality</th>
<th>Var.</th>
<th>Dimension</th>
<th>Plane</th>
<th>Cognitive Process</th>
<th>Subject</th>
<th>Space State: {v_1, v_2}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time</td>
<td>Space</td>
<td>c_1</td>
<td>-</td>
<td>Attention</td>
<td>Orientation</td>
<td>{Direct, Indirect}</td>
</tr>
<tr>
<td></td>
<td>Time</td>
<td>c_2</td>
<td>-</td>
<td>Decision</td>
<td>Impact</td>
<td>{Sudden, Sustained}</td>
</tr>
<tr>
<td>Flow</td>
<td>Weight</td>
<td>c_3</td>
<td>-</td>
<td>Progression</td>
<td>Urgency</td>
<td>{Free, Careful}</td>
</tr>
<tr>
<td></td>
<td></td>
<td>c_4</td>
<td>-</td>
<td>Intention</td>
<td>Impact</td>
<td>{Strong, Light}</td>
</tr>
<tr>
<td>Shape</td>
<td>Shape</td>
<td>c_5</td>
<td>Depth</td>
<td>Vertical</td>
<td></td>
<td>{Rising, Sinking}</td>
</tr>
<tr>
<td></td>
<td>Flow</td>
<td>c_6</td>
<td>Width</td>
<td>Horizontal</td>
<td></td>
<td>{Spreading, Enclosing}</td>
</tr>
<tr>
<td></td>
<td>Space</td>
<td>c_7</td>
<td>Length</td>
<td>Sagittal</td>
<td></td>
<td>{Reaching, Retreating}</td>
</tr>
</tbody>
</table>

a movement, being very difficult, even for the most trained performer, to combine all four [BL80]. Each quality is a continuous between two extremes, which are enumerated in the first four rows of Table 3.1 thus defining the Effort Time variable space. Shape component is also divided in qualities: Flow, Space and Shape [BL80, LT69]. These closely relate to the length, width and depth of movement, carrying a specific terminology according to each dimension. We enumerate Shape elements and how they define a variable space in the bottom three rows of Table 3.1.

The following property characterizes Laban variables: \(c_n = \{v_1, v_2\}\), where \(v_1\) and \(v_2\) are mutually exclusive binomial states, i.e. they verify \(P(c_n = v_1) = 1 - P(c_n = v_2)\) and vice-versa. Each component is independently defined in Laban theory, therefore variables \(c_n \in \mathcal{L}, \forall n\) are considered independent as well.

### 3.2.3.2 Gesture Symbolic Representation

We demonstrate our model’s scalability, augmenting the variable space with level \(\Pi_3\), which represents more complex actions from a combination of symbols \(c_n \in \mathcal{L}\). Variable \(\Lambda\) represents an activity set for our public domain database (MRL), as abstractly considered in equation 3.1. Consider states \(\alpha_m \in \Lambda\) to be one of the following gestures.

\[
\Lambda \in \Omega_{\text{MRL}} : \Lambda = \{\text{bye, punch, point, lift, write}\} 
\]

(3.4)

As mentioned, our experimental set-up is extended to other databases, KTH, WZ and UTI (details in Section 3.5). The correspondent activity variable sets are:

\[
\Lambda \in \Omega_{\text{KTH}} : \Lambda = \{\text{box, clap, wave, run, walk}\} 
\]

(3.5)

\[
\Lambda \in \Omega_{\text{WZ}} : \Lambda = \{\text{box, jump, jack, run, walk, wave1, wave2}\} 
\]

(3.6)

\[
\Lambda \in \Omega_{\text{UTI}} : \Lambda = \{\text{hshake, hug, kick, point, punch, push}\} 
\]

(3.7)
3.3. Multilayer Activity Model

Our research identified four main methodology groups applied to activity recognition. *Deterministic Models* which are unable to deal with uncertainty. *Discriminative Models* (e.g. Neural Networks) can provide fast inference and interpolate flexibly over the trained region, however failing on novel inputs (especially when using small training datasets). Increasing activity complexity or the size of the dataset usually leads to multi-modal state conditionals. Learning such distributions is difficult task, as the majority of existent methods are uni-modal. *Descriptive Stochastic Models* support unsupervised learning, but do not allow prediction, rather focusing on the data’s intrinsic structure. *Generative Stochastic Models* overcome the aforementioned limitations and allow creating statistical models of future behaviour. In this work, we tackle the challenge of generative learning: identifying variables of interest and how they relate. We apply Bayesian methods, exploiting their flexibility. In his Ph.D. thesis [Mur02], Murphy K. demonstrates how to integrate different Bayesian methodologies into a single model. Given different Bayesian models, it is possible to develop a unified model, through the integration of independent sub-models, as their equivalent Dynamic Bayesian Networks (D.B.N.).

3.3.1 Laban Movement Analysis Sub-model

Let the Laban Movement Analysis sub-model be represented by the following joint distribution, where variables $f_i$ are defined as in Section 2.5:

$$
P(f_1,\cdots,f_i,c_1,\cdots,c_n)$$

which, considering variables $c_n \in \mathcal{L}$ to be independent and identically distributed, accepts the following decomposition:

$$
P(c_n|f_1,\cdots,f_i) \propto P(c_n) \prod_{q=1}^{i} P(f_q|c_n)$$

Each gesture $\alpha_m$ can, according to Laban theory, be described as a different sequence and/or combination of Effort and Shape quality descriptors. Despite specific sequences for $\Lambda$ actions can be predefined based on [BL80], we propose the gesture models to be learned upon Bayesian learning techniques, deriving from the experimental data $\in \Omega$. 

### Bayesian Program: Laban Movement Analysis Sub-Model

<table>
<thead>
<tr>
<th>Description</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Variables:</strong></td>
<td>$f_i \in F$: Trajectory feature variables. $c_n \in L$: Laban quality variables.</td>
</tr>
<tr>
<td><strong>Decomposition:</strong></td>
<td>$P(f_1, \ldots, f_i, c_1, \ldots, c_n) = \sum_{q=1}^{i} P(f_q</td>
</tr>
</tbody>
</table>
| **Formulation:** | $P(c_n)$: 
  - $Uniform(c_n)$ for $t = 0$ 
  - $P(c_n)_{t-1}$ for $t \neq 0$ 

$P(f_i|c_n)$: Gaussian Distribution. |

**Identification:** Gaussian parameters $\mu$ and $\sigma$ based on training dataset $\Omega$. 

**Question:** $P(c_n|f_1, \ldots, f_i)$ answered using Maximum A Posteriori (Bayesian Inference).

---

The conditional probability distributions defining each independent sub-model as:

$$P(c_n = v_s|f_i) = \frac{P(f_i|c_n = v_s)P(c_n = v_s)}{\sum_{j=1}^{s} P(f_i|c_n = v_j)P(c_n = v_j)}$$  \hspace{1cm} (3.10)

According to equation (3.10), the posterior probability (left argument) measures the degree of belief the model has about $c_n$, given a set of observable evidences $f_i$. The prior $P(c_n)$ expresses uncertainty about variable $c_n$ before new evidence is taken into consideration. The likelihood distribution $P(f_i|c_n = v_s)$ represents the model learned from previous knowledge. The denominator is a normalization factor, which is often omitted for simplification.

The definitions of the prior and likelihood distributions are of vital importance for an accurate Bayesian Inference. In the first iteration, we assume the model to have no knowledge about $P(c_n)$, therefore we maximize entropy, representing the prior through a uniform distribution. For subsequent iterations, the prior distribution is updated with the previous $c_n$ estimation, i.e. the variable belief state on instant $t-1$ (equation 3.11).

$$P(c_n)_t = \begin{cases} Uniform(c_n) & t = 0 \\ P(c_n)_{t-1} & t \neq 0 \end{cases}$$  \hspace{1cm} (3.11)

Likelihood learning is addressed in Section 3.3.3. The aforementioned mathematical description can be put into a formalism, which is a development pillar of this thesis, which is known as Bayesian Programming. In Figure 3.2 we present the Bayesian Program for the just described model.
3.3. Multilayer Activity Model

Bayesian Program : Activity Sub-Model

Variables:
\[ \alpha_m \in \Lambda_t : \text{Action variable at instant } t. \]
\[ c_n \in \mathcal{L} : \text{Laban quality variables.} \]

Decomposition:
\[ P(\Lambda_1, \cdots, \Lambda_t, c_{n,1}, \cdots, c_{n,t}) = P(\Lambda_0)P(c_{1,1}, \cdots, c_{n,1}|\Lambda_1) \prod_{i=1}^{t-1} P(\Lambda_i|\Lambda_{i-1}) \prod_{j=1}^{c_n} P(c_j|\Lambda_i) \]

Formulation:
\[ P(\Lambda_t) : \text{Stochastic Matrix.} \]
\[ P(c_n|\Lambda_t) : \text{Stochastic Matrix.} \]
\[ P(\Lambda_t|\Lambda_{t-1}) : \text{Stochastic Matrix.} \]

Identification: Baum-Welch Algorithm.\( \Omega \).

Question: \[ P(\Lambda_0, \cdots, \Lambda_{t-1}|\Lambda_t, c_{1,1}, \cdots, c_{1,t}, \cdots, c_{n,t}) : \]
Answered using Maximum A Posteriori (Bayesian Inference).

Figure 3.3: Bayesian Program for the Activity Sub-Model.

3.3.2 Activity Sub-model

Laban theoretically defines each \( \alpha_m \in \Lambda \) as a sequence of \( c_n \in \mathcal{L} \). To learn an activity \( \Lambda = \alpha_m \), we selected Hidden Markov Models (H.M.M.) [MT09], an adequate algorithm to model sequences of state changes. Let \( v_{s,t} \) and \( \alpha_{m,t} \) represent at instant \( t \), the states for variables \( c_n \) and \( \Lambda \) respectively. The initial state distribution is given by \( \Lambda = \{ \alpha_{m,\text{init}} \} \) with \( \alpha_{m,\text{init}} = P(\Lambda = \alpha_{m,0}) \). Transition and Observation probabilities are given by \( \Phi = \{ \phi_{i,j} \} \) and \( \Theta = \{ c_i \} \) respectively, which correspond to \( P(\Lambda_t = \alpha_j|\Lambda_{t-1} = \alpha_i) \) and \( P(\Lambda_t | \alpha_t, c_{n,t} = v_{s,t}) \). The representation of the proposed Markov Model yields:

\[
m_t(i,j) = P(\Lambda_t = \alpha_j, \Lambda_{t-1} = \alpha_i | \Phi, \Theta, \alpha_{m,\text{init}}, \sigma) \tag{3.12}\]

where \( \sigma \) corresponds to the sequence of actual observations \( c_{n,t} \). Parameters are considered time-invariant, which we justify with support on the definition of \( \omega_\alpha \) and properties of the KLT algorithm: shifting a sample \( \omega_\alpha \) in time will not modify the generated features \( f_i \). This property [Mur02], allows representing the Markov model in equation (3.12) as the equivalent Dynamic Bayesian Network:

\[
m_t(i,j) \propto P(\Lambda_t)P(\Lambda_{t-1}|\Lambda_t)P(c_{n,t}|\Lambda_t) \tag{3.13}\]

During the inference process, \( \Theta \) represents a latent observation space, generated at Laban sub-space, upon the iterative classification framework update process. Figure 3.3 presents the Bayesian Program for the Activity Sub-Model.
3.3.3 Learning

Learning the multilayer activity model, is a statistical process based on real experimental data. This process is done from a maximum likelihood perspective, under the assumption that activity descriptors are expected to be both repeatable and reproducible, according to our interpretation of Laban theory. Our variable sub-spaces are defined as discrete, therefore conditional multi-modal stochastic matrices are applied for likelihood representation. Consider a supervised learning approach, where segments are manually labelled, in a process whose data flow is illustrated in Figure 3.4. Symbols from Laban sub-space are associated to each consecutive, non-overlapping segment $\hat{\omega}_i$. They have a (heuristically defined) fixed temporal duration of 1 second. Next, we will formulate the learning of a specific sub-model, however assume the process to be generalizable for the remaining. The probability distribution $P(f_i|c_n)$ is defined as a stochastic matrix $M_{i \times n}$ (for dimensions $f_i$ and $c_n$). The value $f_i \in \mathbb{R}: a \leq f_i \leq b$ is discretized into $\kappa$ number of equidistant classes, has described in Section 2.5. The variable $c_n$ is discrete by definition: $c_n \in \mathcal{L}: c_n = \{v_1, v_2\}$. Each $m_{v,K} \in M$ accounts for the number of occurrences of $f_i = \kappa$ for $c_n = v$. Probability for each $m_{v,K}$ is given.

![Diagram](image)

**Figure 3.4**: The typical experimental set-up for which one of three sensors is chosen, its data processed and then depending on the system’s goal might be manually annotated for learning purposes, or fed into the model for information inference. For video camera networks, body part trajectories must be segmented from the images. The annotation staff shows an example of annotated Laban sequences with respect to the activity sequence in Cartesian Space.
by equation (3.14).

\[ P(f_i = \kappa|c_n = \upsilon) = \frac{m_{\upsilon,\kappa}}{\sum_{q=1}^{m_{\upsilon}} m_{\upsilon,q}} \]  

(3.14)

An earlier version of our model parametrization [SD11b], exploited Gaussian distributions.

In the activity sub-model, transition distributions \( P(\Lambda_t = \alpha_i|\Lambda_{t+1} = \alpha_j) \), are computed through a similar algorithm, where occurrence matrices are \( m \times m \) squared, along dimensions \( \Lambda_t \) and \( \Lambda_{t+1} \). With the appropriate normalization, the learning process is generalizable, allowing to learn the activity model based on Laban sequences.

Remark: We are representing H.M.M. as a D.B.N. equivalent, therefore standard Bayesian learning algorithms can be applied [Mur02].

### 3.3.4 Inference

Upon definition of the prior and learning distributions, the model’s joint distribution can be queried for information, performing Bayesian Inference. We perform inference over two different variables. The first addresses the most likely state \( c_n = \upsilon_s \) given an observable evidence \( f_i = \kappa \). The formulation of this Bayesian question (left argument) and inference, yield:

\[
P(c_n|f_1, \cdots, f_i) \propto P(c_n) \prod_{q=1}^{i} P(f_q|c_n)
\]  

(3.15)

The second question concerns activity estimation, given an observed sequence of Laban parameters.

\[
P(\Lambda|\Phi, \Theta, \alpha_{m,\text{init}}, \sigma) \propto P(\Lambda_t) P(\Lambda_{t-1}|\Lambda_t) \prod_{q=1}^{n} P(c_{q,t}|\Lambda_t)
\]  

(3.16)

The are a number of inference algorithms. We opted for Maximum A Posteriori (MAP), in which belief states are inferred by maximizing Bayes theorem, and are generically defined as:

\[
\text{var}_{\text{MAP}}(\text{obs}) = \arg \max_{\text{var}} P(\text{obs}|\text{var})P(\text{var})
\]  

(3.17)

where \( P(\text{obs}|\text{var}) \) and \( P(\text{var}) \) represent likelihood and prior distributions respectively. The acronyms \( \text{obs} \) and \( \text{var} \) stand for observation (evidence), and variable respectively.
Chapter 3. Activity Recognition and Hierarchical Analysis

Figure 3.5: Top: Average values for convergence time measured in model iterations for different values of $\sigma_{\text{noise}}$. Bottom: Model true positive classifications for different noise standard deviations values $\sigma_{\text{noise}}$.

3.4 Model Evaluation and Performance

The following experiments use simulated data with the purpose of evaluating model performance, subject to noise, poor discriminant features and measuring the impact of the implemented hierarchic taxonomy. Performance is measured with respect to convergence speed and accuracy. Convergence is considered when $P(\Lambda = \alpha_m) \geq 0.999$, and speed is measured in number of inference iterations. The model is accurate when the most probable class equals the ground truth annotation. To validate our experiments of statistical significance, each simulation is tested over $10^4$ trials.

3.4.1 Input Signal Noise

Devices are usually prone to interference, affecting the quality of the acquired signal. A popular way of representing interference is white noise [Die07], which is modelled using a Gaussian distribution.

Let us consider a simple Bayesian question $P(c_n|f_i)$ where $c_n = \{v_1, v_2\}$ and $f_i \in \mathbb{R}$. We define model likelihood as a kernel of Gaussian distributions, indexed by $c_n$ states, as:

$$P(f_i|c_n = v_1) = \mathcal{N}(0,1)$$  \hspace{1cm} (3.18)
$$P(f_i|c_n = v_2) = \mathcal{N}(1,1)$$  \hspace{1cm} (3.19)

Assume an observation $f_i = 0$ subject to a source of additive white noise of $\mu_{\text{noise}} = ...$
0, considering different standard deviations $\sigma_{\text{noise}} \in [0, 5]$. The average number of iterations needed to achieve convergence, are depicted in Figure 3.5 (top) for a range of different noise values. We verify an approximately constant behaviour for $\sigma_{\text{noise}} = [0, 2]$, whereas posterior values are linearly decaying. However, Figure 3.5 (bottom) shows that faster convergence is achieved at the expense of lower accuracies. Hence, graph visualization allows to define a threshold for input noise as $\sigma_{\text{noise}} \simeq 3\sigma_{\text{likelihood}}$, for which the model as experimentally exhibited good accuracy. Over the threshold boundary $\sigma_{\text{noise}} = 3$, the model takes an average 8 iterations to accurately converge over 87% trials, which we consider as an indicator of a good accuracy performance.

3.4.2 Feature Selection

Feature selection is a key stage in model development, which may induce internal noise, direct affecting model performance. Consider the following analysis tests to be with respect to the number of features and their discriminant ability (Quality). Two scenarios are defined considering Good Quality (Scn.1) and Poor Quality (Scn.2) features. Assume the following instantiation $P(c_n | f_1, \cdots, f_i)$. We define likelihood kernels of Gaussian distributions, considering quality to be directly reflected in the likelihood variable $\sigma_{\text{likelihood}}$. Kernel definitions for each scenario, are enumerated as follows:

\[
\text{Scn.1} = \begin{cases} 
P(f_k | c_n = v_1) = \mathcal{N}(0, 1) & \forall k = 1 : i \\
P(f_k | c_n = v_2) = \mathcal{N}(1, 1) & \\
\end{cases} 
\tag{3.20}
\]

\[
\text{Scn.2} = \begin{cases} 
P(f_k | c_n = v_1) = \mathcal{N}(0, \frac{k+2}{3}) & \forall k = 1 : i \\
P(f_k | c_n = v_2) = \mathcal{N}(1, 1) & \\
\end{cases} 
\tag{3.21}
\]

Observations are $f_k = 0$, $\forall k$, subject to a source of additive white noise characterized by $\mu_{\text{noise}} = 0$ and a constant $\sigma_{\text{noise}} = 1$. As the graph of Figure 3.6 (top) illustrates, adding good features exponentially improves convergence speed. Contrary to this behaviour, inconsistency is observed when poor discriminant features $f_i$ are added. The number of iterations for convergence improves considering $k \in [1, 3]$, diverges in $k \in [3, 6]$, peaking at $k = 6$ and resumes faster times at $k \in [6, 10]$. The irregular behaviour is contextualized with accuracy results depicted in Figure 3.6 (bottom). As expected, scenario 1 displays perfectly accurate results, as adding new discriminative information leads to faster convergence. In case of poor observed evidence, the model
starts diverging for $k \geq 6$, where a total accuracy incapability is observed for $k = 10$.
These results corroborate the finding from previous subsection, where for $k \in [1, 5]$ the
model tolerates the existence of noise (poor quality features). However, as quality
in $f_i \in F$ degrades, convergent results appear at the expense of a slower convergence
speed. We demonstrate that a good balance between the number of features and their
discriminant capabilities, lead to both fast and accurate results.

3.4.3 Error Propagation for Different Model Topologies

Let us consider three different hierarchic topologies as illustrated in Figure 3.7, charac-
terized with respect to the dependencies between bottom and middle nodes. Scenario
1 is characterized for the lack of shared connectivity; Scenario 2 presents partial shar-
ing; and finally Scenario 3 assumes all bottom observations connect to all middle level
variables. Each Scenario is tested against its non-hierarchical correspondent which is
defined removing all nodes $c_n$ from layer $L$. Nodes that connect to only one variable are
represented by distributions $P(c_n = v_s|f_k) = N(s, 1)$. Augmenting node connectivity
is reflected with higher values of $\sigma$ in non-hierarchical counterparts. Equation (3.22)
Figure 3.7: Directed Acyclic Graphs defined for the 3 defined scenarios; There are 3 abstraction levels F, L and G from bottom-top respectively.

Figure 3.8: Top: Average Convergence Time measured in model iterations for Hierarchic and Non-hierarchic Topologies. Bottom: Model classification accuracy with respect to the different hierarchic and non-hierarchic topologies.

defines this concept.

\[ P(c_k | \Lambda = \alpha_m) = \mathcal{N}(m,k) \]  

(3.22)

Variable \( k \) represents the number of \( f_i \) nodes connecting to a given \( c_n \) node. Nodes \( c_n \) and \( \Lambda \) are binomial variables, where \( P(\Lambda = \alpha_1 | c_1, c_2) \) is symmetric with respect to \( \alpha = \{\alpha_1, \alpha_2\} \) and defined by the histograms \( \{0.2;0.8\} \) and \( \{0.8;0.2\} \) respectively. Two different observation values are tested, which are added white noise of \( \sigma_{\text{noise}} = 1 \). Observations values 1 and 1.4 are used, implicitly forcing the model to converge for class \( \alpha_1 \). Results in Figure 3.8 (top) compare hierarchic and non-hierarchic equivalent networks in terms of their convergence speed. For Scenario 3, we observe a slower convergence for non-hierarchical model, which is consensual with previous subsection
results, considering the impact of available evidence to be reflected in Gaussian likelihood \( \sigma \), as defined in equation (3.22). The remaining scenarios 1 and 2 maintain a similar performances for both topologies. Corresponding accuracy is depicted in Figure 3.8 (bottom). When the observation is strongly biased towards class \( \alpha_1 \), i.e. \( f_k = 1 \), both topologies are perfectly accurate across all scenarios (Figure 3.8a (bottom)). When the observed signal \( f_k \) is closely in between classes \( \alpha_1 \) and \( \alpha_2 \) averages, but slightly biased towards \( \alpha_1 \), accuracy is slightly affected in both topologies. However, minimal effect is felt, as accuracy is still over 95%. The presented results show minor discrepancies between hierarchic and non-hierarchic topologies in both speed and accuracy. Situations where total connectivity between nodes exist are considered to be exceptions. Thus, we can conclude that parametrizing activities throughout a hierarchically structured model, allows to infer significantly more information, without affecting model performance.

3.5 Extended Experimental Set-up

The Mobile Robotics Laboratory (MRL) public domain motion database\(^*\) consists in a collection of 93 motion sequences of 5 persons performing the 5 gestures defined in Equation (2.14), see Figure 3.9. The data is composed of low-resolution (320 x 240 pixel) image sequences synchronized with high-resolution Cartesian trajectories acquired with MVN suit from XSens\(^†\) at a \( f = 120 \)Hz, which are posteriorly undersampled to 40Hz. Our framework is extended to acknowledgeable and public datasets: the Swedish: Royal Institute of Technology (KTH) human action dataset [SLC04a]; the Weizmann (WZ) dataset [GBS+07]; the University of Texas Interaction (UTI) dataset [RA10, RA09]. (See Figure 3.10). These datasets have a good compromise between upper and lower body gestures.

The dominant Laban symbolic description for all activities and datasets are summarized. Our results are compared to state-of-the-art approaches, using the KTH, WZ and UTI datasets, which are amongst the most used in action recognition [CCFC13]. As a consequence of our extended experimental set-up, we are able to show generalization and invariance properties of Laban symbolic descriptions.

Remark: KTH, WZ and UTI databases are video-based, therefore the models consider 2-D trajectories, which are roughly (manually) tracked, to induce noise.

---

\( ^† \)http://www.xsens.com
Figure 3.9: Example actions from the Mobile Robotics Dataset and URL layout, which highlights its main features: Scenario and Device listings. The database was developed under the scope of the European FP6 Bayesian Approaches to Cognitive Systems (BACS) Project.

Figure 3.10: Example actions from external, publicly available datasets.

Remark: Laban sub-model \((P(f_k|c_n) \in \Pi_2)\) is learned based on data \(\Omega_{MRL}\) only.

Remark: The activity sub-model for the \(\Omega_{KTH}, \Omega_{WZ}\) and \(\Omega_{UTI}\) datasets, is an unsupervised learning process (for which only half of the trials are used to train the model), based on automatically classified activity sequences, upon inference from sub-model \(\Pi_2\), on each aforementioned dataset.

### 3.5.1 Experimental Results on Laban Sub-model

In Figure 3.11, we can observe the classification accuracy for Laban sub-model with respect to the ground-truth annotated MRL dataset. The model converged accurately
in 83.66% of the analysed sequences, improving when compared to our previous approach [SD11a], which yielded an average of 67.79%. However, some inconsistency is observed across some variables, where the max and min accuracy range from 94.1% to 50.0%, respectively for Free and Rising states. We recall a previous statement from sub-section 2.5, which states KLT might not be ideal to characterize all variables. In fact, variables associated to geometric properties are prone to ambiguity, given KLT algorithm characteristics. Moreover, Shape Space component is only observable at beginning and end phases for most actions, which may indicate an insufficient number of learning data, and consequently under-average accuracy. Overall results show what we consider a good accuracy ratio for the Laban sub-model.

![Laban Sub-Model Classification Accuracy](image)

Figure 3.11: Classification accuracy for Laban space variables across MRL-3D full dataset.

### 3.5.2 Experimental Results on Activity Recognition

For this set of experiments we apply a sliding window approach, of length $l$ corresponding to a 1 second interval. Contrary to the training stage, where no overlap was considered, the window time shift is now of 0.25 seconds, generating a segment overlap per consecutive $\omega_{t,m}^l$ and $\omega_{t-1,m}^l$. In our approach, the activity classification emerges from a combination of LMA components, from which experimental results on all datasets are summarized in the confusion tables presented in the following subsections.

*Remark:* For a fair comparison, benchmark is made by directly using activity classification accuracies between different methods when applied to the same datasets.
3.5.2.1 MRL Dataset

Activity classification accuracy on the MRL dataset shows a residual number of misclassified sequences. This fact may be justified due to similarities between different gestures, i.e. during the acquisition sessions, a small number of movements are sluggishly performed in order to create a set of noisy samples. In addition, one should take into consideration the possibility of wrongly classified Laban states to propagate from the Laban classification level. However, even considering these factors, results show an average accuracy of 94.62%, were all sequences have accuracies over 90.00%. To complement our experiments and provide a fair benchmark basis, the MRL dataset has also been tested using 2-D data, based on the trajectories segmented from the acquired image sequences. When comparing with the 3-D results, we observe a similar accuracy performance. However, as it would be expected, results are slightly under the ones using 3-D data, due to the noisy nature of the applied tracking method, contrasting with the high resolution MVN data. In addition, consider the loss of one dimension in the Cartesian space. Despite, average accuracy is still over 90%.

3.5.2.2 KTH Dataset

The KTH dataset encompasses 6 different actions, from which we have not considered "Jogging", using the remaining "Walking" (Walk), "Running" (Run), "Boxing" (Box), "Hand Waving" (Wave) and "Hand Clapping" (Clap). KTH has 25 different
<table>
<thead>
<tr>
<th>Classification Algorithm</th>
<th>Reference</th>
<th>per-seq.(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>K-Nearest Neighbour</td>
<td>Dondera et al. [DDD09]</td>
<td>90.00</td>
</tr>
<tr>
<td>Variational Inference</td>
<td>Wang and Mori [WMct]</td>
<td>91.20</td>
</tr>
<tr>
<td>Dynamic Bayesian Network</td>
<td>Proposed</td>
<td>91.50</td>
</tr>
<tr>
<td>Nearest Neighbour</td>
<td>Bregonzio et al. [BGXne]</td>
<td>93.17</td>
</tr>
<tr>
<td>Efficient Nearest Neighbour</td>
<td>Yuan et al. [YLW09]</td>
<td>93.30</td>
</tr>
<tr>
<td>Support Vector Machines (best)</td>
<td>Zhang and Tao [ZTch]</td>
<td>93.50</td>
</tr>
<tr>
<td>N.B.M.I.M</td>
<td>Zhang et al. [ZLLL11]</td>
<td>93.98</td>
</tr>
<tr>
<td>Maximization of Mutual Information</td>
<td>Liu et al. [LASne]</td>
<td>94.16</td>
</tr>
<tr>
<td>Gaussian Mixture Model</td>
<td>Tian et al. [TCLZ12]</td>
<td>94.50</td>
</tr>
</tbody>
</table>

Figure 3.13: Experimental results on KTH dataset: Confusion matrix for per-sequence classification (overall accuracy = 91.50%); Comparison of per-sequence (per-seq.) accuracies with previous methods on KTH Dataset. Acronym: N.B.M.I.M. Native Bayes Mutual Information Maximization.

performers, each allowed 4 trials per action, however we have discarded one of the trials per individual due to constant zoom changes and corresponding lack of calibration information. The overall activity classification accuracy yields a percentage of 91.5%. The major focus of confusion is observed in the Boxing and Waving actions, where the biggest percentage of misclassified samples is observed. In fact, some of the Clapping trials are performed with hands spread wide, which are confused by the model with a number of Waving similar trials. The same phenomenon is observed, where some of the performed Clapping sequences are very short and vigorous, generating a very similar pattern to some of the recorded Punching sequences.

### 3.5.2.3 Weizmann Dataset

The Weizmann (WZ) dataset is composed of 10 different actions, performed by 9 different persons with 1 trial per action. For this experiment we have discarded the "Galloping Sideways”, “Skip” and “Jump in Place”, classifying between the remaining "Bend” (Bend), "Walk” (WK), "Run” (Run), "Jump” (Jump), "Jumping Jack” (J.Jack), "One-Hand Wave” (Wave1) and "Two-Hand Wave” (Wave2). The reason for discarding the aforementioned 3 actions is due to two factors: (1) Skip and Gallop are two gesture
3.5. Extended Experimental Set-up

<table>
<thead>
<tr>
<th>Bending</th>
<th>J.Jacking</th>
<th>Jumping</th>
<th>Running</th>
<th>Walking</th>
<th>Wave1</th>
<th>Wave2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bend</td>
<td>1.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>J.Jack</td>
<td>0.00</td>
<td>1.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Jump</td>
<td>0.00</td>
<td>0.23</td>
<td>0.67</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Run</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>1.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Walk</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.11</td>
<td>0.89</td>
<td>0.00</td>
</tr>
<tr>
<td>Wave1</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.89</td>
<td>0.11</td>
</tr>
<tr>
<td>Wave2</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>1.00</td>
</tr>
</tbody>
</table>

Classification Algorithm Reference per-seq.(%)

- Support Vector Machine: Huang et al. [HWTM09] 82.37
- Cluster Transition Maps: Sharma et al. [SVM11] 90.00
- Fiedler Embedding: Liu et al. [LASne] 90.40
- 3-D Gradients: Klaser et al. [KMS08] 90.70
- Dynamic Bayesian Network: Proposed 92.06
- Support Vector Machines (best): Zhang and Tao [ZTch] 93.87
- K-Nearest Neighbour: Shabani et al. [SZC10] 93.50
- Nearest Neighbour: Bregonzio et al. [BGXne] 96.66
- Nearest Neighbour: Zhong and Stevens [ZS10] 98.60
- Variational Inference: Wang and Mori [WMct] 100.00

Figure 3.14: Experimental results on Weizmann dataset: Confusion matrix for per-sequence classification (per-seq.) (overall accuracy = 92.06%); Comparison of Classification accuracies per-sequence, with previous methods on Weizmann Dataset.

which are not commonly performed in our daily life and (2) to avoid an excess of Jumping actions. Results show the model performance on the Weizmann dataset, achieving an overall accuracy of 92.06%. The results are considered positive as the majority of observed confusion is restricted to Jumping actions, whereas the remaining appear very consistently classified.

3.5.2.4 UT-Interaction Dataset

The UT-Interaction (UTI) segmented dataset is a recent and acknowledgeable dataset [CCFC13], composed of 6 different interaction actions: "Shake Hands" (H.Shake), "Point", "Hugging" (Hug), "Punching" (Punch), "Pushing" (Push) and "Kicking" (Kick). There are a total of 120 video sequences, divided between 2 datasets, encompassing a single action per video. Results show a good recognition accuracy, however for fairness, one must mention that some of the trials are classified with probabilities around 70%. In fact, some of the performed gestures are performed at a slow-motion pace, e.g., there are punch sequences which are performed at slow speeds, making
### Classification Algorithm Reference per-seq. (\%)  
<table>
<thead>
<tr>
<th>Classification Algorithm</th>
<th>Reference</th>
<th>per-seq. (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non-Linear S.V.M.</td>
<td>Ryoo and Aggarwal [RA09]</td>
<td>70.08</td>
</tr>
<tr>
<td>Team BIWI (best)</td>
<td>Ryoo and Aggarwal [RA10]</td>
<td>88.00</td>
</tr>
<tr>
<td>Dynamic Bayesian Networks</td>
<td>Proposed</td>
<td>92.50</td>
</tr>
<tr>
<td>Support Vector Machines (best)</td>
<td>Zhang and Tao [ZTch]</td>
<td>98.30</td>
</tr>
</tbody>
</table>

Figure 3.15: Experimental results on UT-Interaction segmented datasets 1 and 2: Confusion matrix for per-sequence classification (per-seq.) (overall accuracy = 92.50\%); Comparison of Classification accuracies per-sequence, with previous methods on UT-Interaction Dataset.

it hard for the Laban sub-model to converge confidently to states like "sudden" our "strong". To avoid modifying execution speed via simulation, we implicitly imposed an additional challenge to our classifying framework. Despite this fact, the model exhibits an overall accuracy performance of 92.50\%.

#### 3.5.3 Comparison with Related Works

For an accurate and reliable comparison, the cited works are known to have their methodologies tested either on the KTH, WZ and/or UTI datasets. From comparison tables in Figures 3.12 to 3.15 the biggest visible improvement in model accuracy actually comes when compared to our previous approach [SD11a]. Our novel parametrization improved accuracy from 67.16\% to 90.32\% and 94.62\% using 2-D and 3-D data respectively. When testing the model on KTH, WZ and UTI datasets, results within the expected range of state of the art performances.

However, there are advantages present on our proposed methodology. It provides different symbolic analysis of activities simultaneously, and is presented as a flexibly scalable framework, while maintaining high accuracy performance in the activity discrimination. Another relevant fact is that activities from KTH, WZ and UTI datasets are symbolically described with Laban parameters, without their trajectory sequences...
3.5. Extended Experimental Set-up

being used to train the Laban sub-model. An accurate and apparently generalizable activity invariant description (Laban) is observed for different persons and datasets as depicted in Table 3.3. Results show our method to extend high accuracy to a multi informative framework based on Laban notation with no need to be re-trained for new datasets, keeping current state of the art accuracy ranges.

3.5.4 Discussion on LMA Generalization

3.5.4.1 Generalization Indicators

We consider two indicators of generalization: Repeatability and Similarity. Take into consideration motion sequences are temporally aligned, i.e. we consider the initial instant when at least one body part presents significant initial displacement, for all action sequences. Consider the following definitions:

- **Repeatability** is a desired property. In fact, Laban sequences need to be consistently repeatable (at the symbolic level) for different trials $\omega_\alpha$ of the same action. Some mathematical methods are usually presented as indicators to measure repeatability: standard deviation, absolute difference or correlation. However, we perform visual analysis to assess and discuss this property.

- **Similarity** between sequences performed by different persons are also a relevant indicator. Relaxing strict repetitions, we rather expect to observe similar Laban sequences for same actions performed by different persons. For this experiment, we compute average trajectory for the person and activity $\Lambda$, which are posteriorly compared other persons $p$. Using a popular signal processing methodology to measure similarity between generated sequence signals, Standard Deviation (SD), we have a quantitative measure to assess how different are Laban sequences.

We selected three random trials (one per different person) of a bye-bye gesture, and a visual analysis (Figure 3.16) immediately allows to visualize a pattern for Effort Time probability signal. As visible, there is a large time interval where Effort Time is sudden, followed by two ending peaks, common to all three performances, indicating the sequence to be somehow repeatable. This repeatability phenomenon has been observed for the majority of gestures and persons, intuitively suggesting that Laban sequences generate visible patterns. However, cases happened where Laban components presented no patterns, specifically for Effort Weight, Shape Space and some instances of Shape
Chapter 3. Activity Recognition and Hierarchical Analysis

Figure 3.16: Laban sequence example for one motion type performed by three different persons.

Flow. This may be justified by the under-performance of the classifier accuracy for those components.

To increase readability, the similarity results are presented using topological indicators. Table 3.2 considers the following topological definitions:

- ✓ High Similarity $SD \in [0, 15]$
- $\approx$ Medium Similarity $SD \in [15, 30]$
- X No similarity $SD > 30$

Results show that 6 of the 17 gestures are not similar when performed by different persons. The majority is observed in the WZ and KTH datasets. Despite classifying the correct states, there are substantial differences in the confidence (probability value) of estimated Laban states, which is posteriorly reflected in the presented SD values. For the MRL dataset 3 activities show high similarity. The different performers are given a specific gesture performance script and might be influenced by previous observation of other people’s performances, which may suggest an induced mimicking effect. Nonetheless, these are considered positive indicators to be explored in future research. Other 2 actions, from other datasets also exhibit high similarity. In the category of moderate similarity are 6 actions, which was somewhat expected, due to each person’s different performance styles for similar actions. These preliminary results show positive indicators, which demonstrate LMA’s generalization and invariance capabilities.

3.5.4.2 Symbolic Description Statistics

Statistics over the dominant Laban symbols for each gesture in all datasets are presented, which are expected to illustrate the generalization properties that the model exhibits when compared to classic, usually narrowed state-of-the-art approaches.
Table 3.2: Standard Deviation (SD) with topological indicators for similarity. Different activities are on the right column, and N=r is the number of r persons performing that activity.

<table>
<thead>
<tr>
<th>Action</th>
<th>(Dataset) N Different Performers/Action</th>
<th>(MRL)</th>
<th>(MRL)</th>
<th>(MRL)</th>
<th>(WZ)</th>
<th>(KTH)</th>
</tr>
</thead>
<tbody>
<tr>
<td>N=3</td>
<td>N=4</td>
<td>N=5</td>
<td>(N=9)</td>
<td>N=25</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lift</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Write</td>
<td>≈</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Punch</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Point</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bye-bye</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Box</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Clap</td>
<td>≈</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wave</td>
<td>≈</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Run</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Walk</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bend</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jack</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jump</td>
<td>≈</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Run</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Walk</td>
<td>≈</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wave1</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wave2</td>
<td>≈</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 3.3: Dominant Laban states statistics across all datasets considering 2-D data, i.e. the number of times they appear as output of the Laban sub-model. States where dominance is under 60% or do not have enough samples for an accurate conclusion are considered undefined (Undef.). **NOTE:** The Sh.Shape component results are omitted as for most actions, Rising and Sinking states only occur during the initial and end phases generating insufficient number of samples with exception of actions Lift and Bend whose results are Rising - 74.35% and Sinking 94.02% respectively.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>State</td>
<td>%</td>
<td>State</td>
<td>%</td>
<td>State</td>
</tr>
<tr>
<td>MRL</td>
<td>Lift</td>
<td>Sustained</td>
<td>76.42</td>
<td>Direct</td>
<td>86.79</td>
<td>Light</td>
</tr>
<tr>
<td></td>
<td>Write</td>
<td>Undef.</td>
<td>-</td>
<td>Indirect</td>
<td>60.87</td>
<td>Light</td>
</tr>
<tr>
<td></td>
<td>Punch</td>
<td>Sudden</td>
<td>72.17</td>
<td>Direct</td>
<td>88.75</td>
<td>Strong</td>
</tr>
<tr>
<td></td>
<td>Point</td>
<td>Sustained</td>
<td>66.35</td>
<td>Direct</td>
<td>81.63</td>
<td>Strong</td>
</tr>
<tr>
<td></td>
<td>Bye</td>
<td>Sudden</td>
<td>63.44</td>
<td>Undef.</td>
<td>-</td>
<td>Light</td>
</tr>
<tr>
<td>KTH</td>
<td>Box</td>
<td>Undef.</td>
<td>-</td>
<td>Direct</td>
<td>100.00</td>
<td>Strong</td>
</tr>
<tr>
<td></td>
<td>Clap</td>
<td>Undef.</td>
<td>-</td>
<td>Direct</td>
<td>100.00</td>
<td>Light</td>
</tr>
<tr>
<td></td>
<td>Wave</td>
<td>Sudden</td>
<td>100.00</td>
<td>Indirect</td>
<td>67.66</td>
<td>Light</td>
</tr>
<tr>
<td></td>
<td>Run</td>
<td>Sudden</td>
<td>88.78</td>
<td>Direct</td>
<td>100.00</td>
<td>Strong</td>
</tr>
<tr>
<td></td>
<td>Walk</td>
<td>Undef.</td>
<td>-</td>
<td>Direct</td>
<td>100.00</td>
<td>Undef.</td>
</tr>
<tr>
<td>WZ</td>
<td>Bend</td>
<td>Sustained</td>
<td>100.00</td>
<td>Direct</td>
<td>75.34</td>
<td>Light</td>
</tr>
<tr>
<td></td>
<td>Jack</td>
<td>Sudden</td>
<td>100.00</td>
<td>Indirect</td>
<td>69.52</td>
<td>Light</td>
</tr>
<tr>
<td></td>
<td>Jump</td>
<td>Sudden</td>
<td>100.00</td>
<td>Direct</td>
<td>100.00</td>
<td>Light</td>
</tr>
<tr>
<td></td>
<td>Run</td>
<td>Sudden</td>
<td>67.15</td>
<td>Undef.</td>
<td>-</td>
<td>Light</td>
</tr>
<tr>
<td></td>
<td>Walk</td>
<td>Undef.</td>
<td>-</td>
<td>Direct</td>
<td>67.95</td>
<td>Light</td>
</tr>
<tr>
<td></td>
<td>Wave1</td>
<td>Sudden</td>
<td>72.09</td>
<td>Undef.</td>
<td>-</td>
<td>Strong</td>
</tr>
<tr>
<td></td>
<td>Wave2</td>
<td>Sudden</td>
<td>94.76</td>
<td>Direct</td>
<td>90.56</td>
<td>Strong</td>
</tr>
<tr>
<td>UTI</td>
<td>H.Shake</td>
<td>Undef.</td>
<td>-</td>
<td>Direct</td>
<td>97.00</td>
<td>Light</td>
</tr>
<tr>
<td></td>
<td>Hug</td>
<td>Sustained</td>
<td>69.88</td>
<td>Indirect</td>
<td>86.27</td>
<td>Light</td>
</tr>
<tr>
<td></td>
<td>Kick</td>
<td>Sudden</td>
<td>73.45</td>
<td>Direct</td>
<td>95.87</td>
<td>Strong</td>
</tr>
<tr>
<td></td>
<td>Point</td>
<td>Undef.</td>
<td>-</td>
<td>Undef.</td>
<td>-</td>
<td>Light</td>
</tr>
<tr>
<td></td>
<td>Punch</td>
<td>Sudden</td>
<td>77.93</td>
<td>Direct</td>
<td>76.78</td>
<td>Strong</td>
</tr>
<tr>
<td></td>
<td>Push</td>
<td>Undef.</td>
<td>-</td>
<td>Direct</td>
<td>96.62</td>
<td>Light</td>
</tr>
</tbody>
</table>
Let us start this discussion over the Effort Time component. It is possible to observe that gestures traditionally more "energetic" are associated with "Sudden" states, e.g. Punch, Boxing, Jumping or Running. Whereas actions like Bending or Lifting an object, are performed at a more slower pace, therefore classified as "Sustained". Actions in which body part trajectories are not geometrically straight are considered to be "Indirect" such as Writing, Waving or Hugging. The Jumping Jack$\in\Omega_{KTH}$ is performed by jumping forward creating a bowed, non-linear trajectory, thus justifying the Indirect dominance. All other actions are tendentiously prone to linear (Direct) trajectories. Activities like Running, Punching, Pushing, Boxing, kicking or even some of the Waving are commonly considered to have some Strength applied, whereas others are more "subtle" and performed Lightly. Apart from Waving actions and some instances of Running, all others appear to have a purpose, a goal, performed without hesitations. This occurrence might be associated to the fact all performers were instructed on how to perform. States where predominance ratio are under 60% or do not have enough classified samples to define a dominant state, are considered to be undefined.

We argue, supported by table 3.3 that describing motion in Laban Movement Analysis symbolic space demonstrates generalization capabilities. As observed, similar actions from different datasets share the same Laban symbolic description, independently of being performed by different persons.

### 3.6 Applicability

Supported by achieved accuracy results and generalization indicators, we have pinpointed two application scenarios extending our proposed work: Identifying persons on activity invariant spaces and further generalize LMA to hand manipulation and grasping gestures.

#### 3.6.1 Person Identification

Results show that Laban sequences for the same actions, are similar between different trials and persons. However, for some components they are not. This is a natural consequence of a person’s own moving characteristics, which are influenced by psychological and physical properties. Hence, we expect those characteristic Laban sequence differences to be highlighted from their component quality outputs. Those differences may present themselves either topologically (different states) or different probabilistic amplitudes for the same state. Our research will explore these observed differences in
Laban space towards person identification within a monitoring scenario, i.e. via observation of a persons motion a system will automatically discriminate between different person identities.

3.6.2 Hand Grasping and Manipulation Characterization

Laban semantics have been specifically designed for body motion. Our experiments determined that Laban can be generalized for different persons as a generic body motion descriptor. We are currently extending that generalization to Hand grasping and manipulation tasks. Hand motion has specific grammars, which somehow allow a semantic description of motion. However, we will try to demonstrate that Laban can be adapted to other motion types, encompassing the sense of emotion of whoever is performing. Simultaneously, we expect this newly adapted Laban for hand motion to be able to discriminate between different hand activities, while simultaneously qualifying them.

3.7 Conclusions and Discussion

We have presented a scalable multilayer model for analysis of different body language information, based on activity body part trajectories, whose presented research allows withdrawing the following conclusions. The implemented variable qualities in the Laban sub-model, show symbolic analysis to be repeatable for similar actions when performed by different persons, demonstrating activity invariance. The previous conclusion is reinforced with generalization, if one takes into consideration that the Laban sub-model is learned using only a single dataset and classified against all others. The global model has a real-time capability to provide different symbolic analysis simultaneously. A simulated performance analysis shows the model topology not to influence accuracy and convergence speed. Additional tests showed the model to withstand considerable input noise. We demonstrate scalability and flexibility, by modelling additional information levels and using different Bayesian methodologies respectively. Despite its complex taxonomy, our multilayer model does not lose performance when compared state of the art approaches, and showing that higher level activity descriptions can be modelled as a combination of activity invariant Laban components. The model shows analysis capabilities in both 2-D and 3-D Cartesian Spaces. We identify as future work, growing our public database with new sequences including multi-person scenarios, which are targeted to be annotated with Laban descriptors.
Chapter 4

Motion-Based Person Identification

4.1 Introduction

Computational analysis and recognition of human actions focuses on modelling movements’ specific properties, generalizing them to random performers. However, factors like anatomical structure or emotional state are naturally and unconsciously combined during activity execution, imprinting a set of unique and discriminant characteristics. In this manuscript, we capture each person’s specific characteristics exploiting generalized motion properties. To perform person identification, a three stage methodology is proposed:

1. Project 3-D Cartesian motion trajectories onto an activity invariant symbolic space (Section 4.2);

2. Encode the symbolic information into so called Laban signatures (Section 4.3);

3. Associate signatures to identities towards autonomous person identification (Section 4.4).

This work is an extension to a 3-D motion trajectory-based action analysis and recognition research, where a model was developed to retrieve different types of information, using hierarchical symbolic levels [RDA10]. We address the challenge of person identification exploiting one of those levels, which describes motion properties using an action invariant notation, Laban Movement Analysis (LMA) [BL80, Gue70]. Our research targets three main purposes:

- Identify persons based on random actions (gait or non-gait);
• Use 3-D motion trajectory information;
• The subject is not required to explicitly cooperate or interact with the system.

4.1.1 Related work on Person Identification

Table 4.1: Precision for several activity-based person identification approaches, identifying the number of different persons, the experimented dataset and dominant activities.

<table>
<thead>
<tr>
<th>Method</th>
<th>Dataset</th>
<th>Persons</th>
<th>Activity</th>
<th>Activity per-sequence (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gkalelis et al. [GTP09]</td>
<td>Weizmann</td>
<td>9</td>
<td>6</td>
<td>83.58</td>
</tr>
<tr>
<td>Lu et al. [HZS12]</td>
<td>Weizmann</td>
<td>9</td>
<td>6</td>
<td>93.28</td>
</tr>
<tr>
<td>Iosifis et al. [ITP12]</td>
<td>AiiA Mobiserv</td>
<td>12</td>
<td>Eating/Drinking</td>
<td>87.33</td>
</tr>
<tr>
<td></td>
<td>i3DPost Multiview</td>
<td>8</td>
<td>Gait</td>
<td>94.34</td>
</tr>
<tr>
<td></td>
<td>Casia Gait Multiview</td>
<td>124</td>
<td>Gait</td>
<td>93.37</td>
</tr>
<tr>
<td></td>
<td>i3DPost Multiview</td>
<td>8</td>
<td>Gait</td>
<td>94.34</td>
</tr>
<tr>
<td></td>
<td>Casia Gait Multiview</td>
<td>124</td>
<td>Gait</td>
<td>93.37</td>
</tr>
<tr>
<td>Iwashita and Kurazume [IK09]</td>
<td>Southampton Gait Database</td>
<td>20</td>
<td>Gait</td>
<td>92.00</td>
</tr>
<tr>
<td>Kale et al. [KCCay]</td>
<td>CMU MoBo</td>
<td>25</td>
<td>Gait</td>
<td>92.00</td>
</tr>
</tbody>
</table>

Note: The experiments of Kale et al. use a different classification metric, for which we present the achieved accuracy ranges.

Researchers have been addressing the problem of person identification, by adapting action recognition methods to analyse how different persons move, exploiting image-based or joint space information. It is a branch of biometrics, where approaches are dependent on the observation of specific actions, in which gait is still dominant [NC04, WSNK10]. We identify two main research categories, that either explore gait or non-gait actions.

Gait-based Analysis and Identification: Babski et al. [BBT96] proposed a method that would optimally fit an ellipsoid to all leg markers and study the periodic behaviour of its parameters across time. Little and Boyd [LB98] use optical flow, fitting an ellipse to dense optical flow of a person’s motion, using the ellipse characteristics as features. Different domains are also explored, using Fourier descriptors [CNC03, HE04, BN06], Wavelets [CK96] or Curvature functions [MAK96]. Yam et al. [YNC04] extended Cunado et al. system [CNC03] to walk and running actions using temporal template matching. An approach based on Tensor analysis and Gabor features was presented by Tao et al. [TLWM07]. Urtasun and Fua [UF04] present a model using implicit surfaces directly related to an articulated skeleton. To enforce invariance, they used 3D point clusters from a stereo camera. Liu et al. [LS06] adopted dynamics-normalized shape cues applying Hidden Markov Model (HMM) classifiers. Other popular approaches are based on geometrical descriptors such as contour or similar mathematical functions [Kin03, ZL04, CGS02, NA94]. Zhang et al. [ZZX10] project active energy images to
a feature subspace, using two-dimensional locality preserving projections (2DLPP) in
gait recognition. Sequences of uniformly scaled binary silhouettes for one gait cycle
were applied by Murase and Sakai [MS96]. They proposed a template matching method
in a parametric eigenspace created from the images. Sarkar et al. [SPL+05] proposed
to perform spatio-temporal correlation of silhouettes. BenAbdelkader et al. [BCND01]
characterize gait as 2D signature computed from a sequence of silhouettes. Kale et al.
have used Frame to Exemplar Distance (FED) derived from body silhouettes, classifying
different types of gait actions using HMM [KCCay]. Han et al. described walking
and running activities using gait energy image (GEI) [HB06]. Inside the invariant fea-
tures category, Iwashita et al. [IP08 IK09 IBOK10 IUKS12] use 2D and 3D affine
moment invariants to characterize a person’s gait together with K-Nearest Neighbours.
Boulgouris and Chi [BC07], propose a method where different body parts are analysed separately, weighted and encoded based on their relative distances towards gait
recognition.

Non-Gait Analysis and Identification: Vasilescu [Vas02] proposed to analyse and
synthesize motion, using motion signatures based on n-mode analysis [Tuc66]. Signa-
tures are organized in high-order arrays or tensors, defining multi-linear operators over
a set of vector spaces. Gkalelis et al. [GTP09] use Fuzzy C-means and K-means with
the purpose of generating motion signatures for identifying different persons. Lu et
al. [LHZS12] exploit silhouette sparse coding with mean pooling to perform identifi-
cation based on different activities. Iosifidis et al. [ITP12] present another interesting
study using either gait and non-gait activities, which classifies people based on dynemes
[GTP08]. Binary images of body regions are vectorized and clustered using K-Means,
posteriorly applied in a Bayesian classifier.

A short-list of relevant works on activity-based person recognition, which have
recently shown high identification precision in both gait or non-gait based approaches,
is presented in Table 4.1. It is a fact that gait-based approaches are still dominant
and consistently achieve precisions over 90%, for large numbers of different persons.
Non-gait approaches’ most recent results show similar accuracy ratios, ranging from
83.58% to 93.28%, considering an average number of ≈ 10 different persons.

4.1.2 Our Approach

State of the art methods are usually supported by 2-D, image based data, either im-
plcitly relying on the specific properties of gait motion, for example, periodicity, or
requiring persons to perform specific activities in order to present accurate identifi-
cations. Our approach is based on 3-D motion trajectories of different body parts, focusing on observing generalizable characteristics rather than a single action’s properties. Moreover, it does not restrict a person’s performance to a given action, in the sense that it does not depend on specific activity knowledge.

A global overview of the proposed framework is illustrated in the diagram of Figure 4.1. We propose to encode the symbolic motion expressive qualities using two different techniques: Pearson Product-Moment Correlation Coefficients and Topological Adjacency Matrix. A third method is proposed, based on Gaussian Mixture Models. These techniques encode a unique signature for each person, exploiting the existent relations between different, LMA-based, action-invariant qualities. The Bayesian-based person identification models are trained using supervised learning, where variable spaces are related through conditional probability distributions. The main contributions identified from our solution are:

- An activity invariant symbolic space for representing generalisable qualities of motion trajectories;
- Signature techniques encoding symbolic information to retain a person’s unique motion expressive properties;
- Two different Bayesian-based classifiers which, using the proposed signatures, perform computational inference over person identities.

![Figure 4.1: Block diagram of the proposed system, from motion sensing to person identification, whose inference phases are done using Bayesian-based models. Variable List: Trajectory $\omega_p$; Identity $\zeta$; Trajectory Feature $F = \{f_b\}$; Laban Symbol $c_n$; Signature Feature $\Gamma = \{\gamma_r\}$.](image-url)
4.1. Introduction

Our main experimental set-up uses the publicly available University of Coimbra 3-D Motion Database, developed in the context of this work. We demonstrate that our method is also applicable to 2-D motion trajectories, extending our experiments to two acknowledgeable and publicly available video-based motion datasets.

4.1.3 Laban Variable Space

In this section we recall some concepts of Laban Movement Analysis and summarize the selected components and qualities used in this chapter. A key concept relies on labanotation’s ability to describe activities using generic sequences of symbols [BL80], but most importantly "each individual has its own way for combining these components according to its cultural, personal and artistic preferences" [Zha02]. Effort and Shape components represent expressive properties and are always observable in any movement, therefore they define the Laban variables $c_n \in \mathcal{L}$, that will be used to define the dimensions of Laban space $\chi$. Identically to the exposed in the previous chapter, and maintaining formal coherence, the formal definition for each Laban variable yields $c_n = \{q_1,q_2\}$, where $q_1$ and $q_2$ are mutually exclusive binomial states, verifying $P(c_n = q_1) = 1 - P(c_n = q_2)$ and vice-versa. Independent component definitions in Laban theory are reflected by independent and identically distributed variables $c_n$.

The first three rows of Table 4.2 illustrate the qualities and correspondent extreme factors defining the Effort Time variable space, while the bottom two enumerate the qualities used to define the Shape component variable space.

Table 4.2: Summary of implemented Effort and Shape qualities and their respective characteristics.

<table>
<thead>
<tr>
<th>Comp.</th>
<th>Quality</th>
<th>Var.</th>
<th>Dimension</th>
<th>Plane</th>
<th>Cognitive Process</th>
<th>Subject</th>
<th>Space State={q_1,q_2}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Effort</td>
<td>Time</td>
<td>$c_1$</td>
<td>-</td>
<td>-</td>
<td>Decision</td>
<td>Impact</td>
<td>{Sudden,Sustained}</td>
</tr>
<tr>
<td></td>
<td>Space</td>
<td>$c_2$</td>
<td>-</td>
<td>-</td>
<td>Attention</td>
<td>Orientation</td>
<td>{Direct,Indirect}</td>
</tr>
<tr>
<td></td>
<td>Flow</td>
<td>$c_3$</td>
<td>-</td>
<td>-</td>
<td>Progression</td>
<td>Urgency</td>
<td>{Free,Careful}</td>
</tr>
<tr>
<td>Space</td>
<td>Space</td>
<td>$c_4$</td>
<td>Length</td>
<td>Sagittal</td>
<td>Momentum</td>
<td>-</td>
<td>{Reaching,Retreating}</td>
</tr>
<tr>
<td></td>
<td>Flow</td>
<td>$c_5$</td>
<td>Width</td>
<td>Horizontal</td>
<td>Body Span</td>
<td>-</td>
<td>{Spreading,Enclosing}</td>
</tr>
</tbody>
</table>

4.1.4 Problem Statement

Let $\Omega$ be a database of annotated 3-D motion trajectories, recorded from multiple action sequences, performed by persons $p$ belonging to a set $\zeta$ of known identities.
Consider a trajectory $\omega_p$ defined as:

$$\omega_p = \begin{bmatrix} Y_1 \\ \vdots \\ Y_S \end{bmatrix}, Y_{i=1:S} \in \mathbb{R}^3$$ (4.1)

which represents a sequence of coordinates $Y_i$ in Cartesian space, where $i = 1, \cdots , S$ is the sample index. Each trajectory $\omega_p \in \Omega$ is characterized with a set of symbols $c_n \in \mathcal{L}$, based on $n$ Laban components [SD11a].

Let $\Omega_\Gamma$ be an associative space, relating persons $p$ to motion signatures $\Gamma$, where a signature is a mathematical representation encoding symbolic qualities $c_n$ to characterize each person’s unique expressive style. Let a person perform a random action generating a correspondent signature. We propose different techniques allowing to perform inference over person identities by means of conditional probability distributions, in a Bayesian based approach.

The remaining of this chapter is organized as follows. A brief introduction to our problem and previous work using Laban as a parametrization language is described in Section 4.2. Signatures based on different methods, their discriminant comparative studies and discussions are presented in Section 4.3. The Bayesian-based identification models, learning and inference are formulated in Section 4.4. Section 4.5 presents experimental results and related discussion. Conclusions and extension of this work are presented in Section 4.6.

### 4.2 Activity Invariant Symbolic Space

To define the activity invariant descriptor space, we exploit the generalized symbolic descriptors from our previous chapter. As shown, Laban variables consistently showed to be classified with the same states, for similar actions even when performed by different actors. However, the probability values with which those symbols are classified, differed from person to person, a property which the present chapter aims to exploit. Consider that for developing Laban Space, we use the outputs inferred from the Laban Movement Analysis Sub-Model.
### 4.2.1 Laban Space Definition

Consider Laban Space $\chi \in \mathbb{R}^n$ as an *n-dimensional* unified representation space for the different Laban variables $c_n \in L$. Let $\tau_n$ represent the $n^{th}$ dimension in $\chi$, associated to $c_n$. Therefore, the coordinate vector $R \in \chi : R = (\tau_1, \cdots, \tau_n)$ allows representing the symbolic Laban properties of the trajectory sub-segment $\hat{\omega}_p$ in Laban Space, such that a value along a given dimension is defined as $\tau_n = P(c_n = q_1)$. We exploit the mutual exclusivity property, using the probability for a single state $q \in c_n$ to avoid redundancy, while simultaneously reducing the dimension of $R$. Laban Space is defined in equation (4.2).

$$R \in \chi = \{ c_n \in L \mapsto \tau_n = P(c_n = q_1) \in \mathbb{R} : 0 \leq \tau_n \leq 1 \} \quad (4.2)$$

In an intuitive interpretation, each dimension $\tau_n$ quantifies *how much* of a Laban quality is observed for a given motion trajectory segment, e.g. an absolute value for $\tau_n = 0$, means the trajectory segment $\hat{\omega}_p$ is exhibiting properties which are dominantly associated to quality $q_2$. Conceptually, analysing how a given component $c_n$ relates to the other variables $c_i, i \neq n$, when the observed state is $c_n = q$, involves the observation over a region in Laban Space, that is bounded by the hyperplanes defined as $0.5 < P(c_n = q) \leq 1$.

### 4.3 Laban Signatures

We present three different approaches to generate signatures, encoding information from Laban Space: the Pearson’s Correlation Coefficient approach; the Topological (or Adjacency Matrix) and Gaussian Mixture Model approaches, which are identified by $P_{sig}$, $T_{sig}$ and $M_{sig}$ respectively.

#### 4.3.1 Pearson Correlation Approach

This approach attempts to characterize the existing component relations in Laban space, by measuring the linear dependency between any two dimensions $\tau_i$ and $\tau_j$. Let $\Pi_{t \times n} = [R_1 \cdots R_t]^T$, be a matrix of coordinate vectors $R \in \chi$, where $T_i$ is the $i^{th}$ column vector of $\Pi$, representing a sequence of $t$ observations for coordinate $\tau_n \in R$. Consider $P_{sig}$ the Pearson’s coefficient matrix, where each correlation element $\rho_{i,j}$ is
Chapter 4. Motion-Based Person Identification

defined as:

\[
P_{\text{sig}} = \begin{bmatrix}
\rho_{0,0} & \cdots & \rho_{0,n} \\
\vdots & \ddots & \vdots \\
\rho_{n,0} & \cdots & \rho_{n,n}
\end{bmatrix}, \text{ for } \rho_{i,j} = \frac{\text{cov}(T_i, T_j)}{\sigma_i \sigma_j}
\]  

(4.3)

with \(i, j = 1, \ldots, n\). The values \(\sigma_i\) and \(\sigma_j\) are the variance in \(T_i\) and \(T_j\) respectively, while \(\text{cov}(T_i, T_j)\) is their covariance. The existence of \(n^2\) coefficients \(\rho_{i,j}\) in \(P_{\text{sig}}\) will, expectedly, generate a wide variety of discriminant motion signatures.

The sample size of \(T\) is selected with two key issues in mind. Should not be short so to generate highly volatile measures (influenced by noisy measures), nor should it be long enough to stabilize for a given value of \(\rho\), preventing a signature to correct itself from sequences of misclassified \(c_n\). In our experiments, we set \(t = 40\), where samples \(R\) are updated according to a First In First Out paradigm.

### 4.3.2 Topological Approach

The topological approach, establishes variable relations by means of a sorted chain, with respect to a given parent node (see Figure 4.2). Let \(G\) define a topological space, where each node represents an unique symbol corresponding to one, and only one variable \(c_n\). The distance between two nodes \(c_i\) and \(c_j\) is \(d_{i,j} = \tau_i - \tau_j\). In this work, ascending sorting was applied (Algorithm 1). Ideally we could represent the sorted chain using a vector if all \(d_{i,j}\) were different, however, we cannot guarantee this property for all \(i, j = 1, \ldots, n\). Therefore, we use a \(n \times n\) squared adjacency matrix \(T_{\text{sig}}^n\), where

\[
T_{\text{sig}}^n = \begin{bmatrix}
0 & 0 & d_{1,4} & 0 \\
0 & d_{1,3} & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & d_{1,2} & 0 & d_{1,5} \\
0 & 0 & d_{1,3} & 0
\end{bmatrix}
\]

Figure 4.2: Projection of variables \(c_n \in L\) onto the topological space \(G\). By means of a sorting function, a parent node \(c_{\text{parent}}\) is selected and the remaining nodes sorted by their ascending scores \(d_{\text{parent},j} = \tau_{\text{parent}} - \tau_j, \forall j \neq \text{parent}\).
Algorithm 1: (Sorting Function) $\text{Sort}(G, \text{parent}_n)$

Require: $G = [c_1 \ldots c_n]$ and Parent Node: $c_n$

1: for $j = 1$ to $n-1$ & $j \neq i$ do
2: \hspace{1em} if $d_{i,j} > d_{i,j+1}$ then
3: \hspace{2em} Switch $c_j$ with $c_{j+1}$ ;
4: \hspace{1em} end if
5: end for
6: return Sorted $G$

the superscript $n$ identifies the parent node $c_n$. The values of elements $a_{i,j} \in T^{n}_{\text{sig}}$ reflect directed connectivity on a "From-To" basis, i.e. from node $i$ to node $j$. An initial hypothesis was considered, using boolean logic TRUE or FALSE for values of $a_{i,j} \in \{0,1\}$ to represent the existence of connectivity or not respectively. Experiments using this approach showed low discriminant matrices. Motivated by the upcoming sub-section 4.3.4 we devised an alternative approach for selecting the values for $a_{i,j}$, which is presented in equation (4.4).

$$a_{i,j} = \begin{cases} 0, & \text{No Connectivity} \\ d_{i,j}, & \text{Connectivity} \end{cases}$$ (4.4)

Consider an example, where the parent node is $c_1$ and scores obey the following inequalities $d_{1,4} < d_{1,2} = d_{1,5} < d_{1,3}$. The adjacency matrix presented on the right side of Figure 4.2 represents the sorted topological graph presented on top, where score values are defined as $\{d_{i,j} \in \mathbb{R} : -1 \leq d_{i,j} \leq 1\}$. An independent matrix $T^{n}_{\text{sig}}$ is generated for each parent node $c_n$.

4.3.3 Gaussian Mixture Model Approach

During an activity performance, we expect the projection of each person’s characteristics in $\chi$, to be differently distributed. Gaussian Mixture Models are applied to find a probabilistic representation of those patterns. Consider a set of trajectories $\omega_p$ for person $p \in \zeta$, that generate a training point cloud $\Pi_p$ in Laban Space. We propose to model $\Pi_p$ using a set of GMM parameters $\Theta$ defined as follows:

- $N_p = \text{number of samples of } R \in \Pi_p$
- $K \approx \sqrt{\frac{\min\{N_p\}}{2}}, \forall p \in \zeta = \text{no. of components}$\textit{[MKB79]}
- $\mu_i=1\ldots K = \text{mean of component } i$
- $\sigma^2_i=1\ldots K = \text{variance of component } i$
• $\phi_{i=1...K} = \text{weight of component } i$, $\sum_{i=1}^{K} \phi_i = 1$

The probability density function of the Gaussian mixture model can be represented by:

$$F(c_n|\Theta_k) = \sum_{i=1}^{K} \phi_i f(c_n|\theta_i) \quad (4.5)$$

where $\theta_k$ is the set of parameters $\{\mu_k, \sigma^2_k\}$ of component $k$, which jointly with the weights, define vector $\Theta_k = \{\phi_1, \phi_2, ..., \phi_k, \theta_1, \theta_2, ..., \theta_k\}$. The Gaussian density distributions for each $\theta_k$ can be expressed as:

$$f(c_n|\theta_k) = \frac{1}{(2\pi)^{n/2}|\Sigma_k|^{1/2}} e^{-\frac{1}{2}(c_n-\mu_k)^T \Sigma_k^{-1}(c_n-\mu_k)} \quad (4.6)$$

where $\Sigma_k$ is the covariance matrix assumed positive definite. According to the previous equations (4.5) and (4.6), the density of the Gaussian mixture is expanded, yielding

$$F(c_n|\Theta_k) = \sum_{i=1}^{K} \phi_i f(c_n|\mu_i, \Sigma_i).$$

Given a sample dataset $\Pi_p$, the optimal parameters $\Theta_k$ are estimated from a maximum likelihood approach, using the following log-likelihood function, based on the Expectation Maximization algorithm.

$$L(\Pi_p|\Theta_K) = \log p(\Pi_p|\Theta_K)$$

$$= \log \prod_{j=1}^{n} f(c_j|\Theta_K)$$

$$= \sum_{j=1}^{n} \log \sum_{i=1}^{k} \phi_i f(c_j|\theta_i) \quad (4.7)$$

In order to get a good initialization for the iterative process, we use the K-Means algorithm. The signature for this approach corresponds to the set of parameters yielding $M_{\text{sig}} = \{\Theta_i\}, i=1, ..., k$.

### 4.3.4 Signature Probabilistic Representation

Pearson and Topological signature approaches are represented by $n \times m$ matrices $P_{\text{sig}}$ and $T_{\text{sig}}$, which are now abstractly named $A$. With the purpose of reducing signature dimension, we apply Singular Value Decomposition (SVD) [WRR03]. For a signature matrix $A$, solving equation (4.8) gives a matrix $U$ with $n$ columns representing $m$ dimensional eigenvectors $X_n$, and each diagonal element of $\Sigma$, the correspondent
4.3. Laban Signatures

eigenvalue.

\[ A = U \Sigma V^* \] \hspace{1cm} (4.8)

Assume a sorted \( \Sigma \) where the first element \( \Sigma_1 \) represents the highest eigenvalue, corresponding to eigenvector \( X_1 \). These parameters represent a reduced signature as in equation (4.9).

\[ \Gamma = [\Sigma_1 \ X_1]^T \] \hspace{1cm} (4.9)

In the particular case of the topological approach, there are \( n \) signatures \( \Gamma_n \), one for each \( T^n_{sig} \), which still holds a large number of variables. We propose to generate a global matrix \( A_G \), such that

\[ A_G = [\Gamma_1 \cdots \Gamma_n] \] \hspace{1cm} (4.10)

which, upon solving SVD for \( A_G \), gives a \( \Gamma_G \) with an identical structure to the one presented in equation (4.9). The recursive application of SVD, could result in a two layer signature, in a similar concept to what was presented in [SWZ08]. However, in this work, both layers \( \Gamma \) and \( \Gamma_G \) are used separately in the identification model. We expect to measure the impact of multiple signature dimension reduction, in the process of person identification.

The generated signatures yield a set of parameters, which abstractly represent the variable space \( \Gamma \) defined as:

\[ \Gamma = [\gamma_1, \cdots, \gamma_r] \in \mathbb{R}^r, \text{ with } \begin{cases} \gamma_i = x_i, & i \neq r \\ \gamma_i = \Sigma_1, & i = r \end{cases} \] \hspace{1cm} (4.11)

where \( \gamma \)'s are independent and identically distributed. Variables \( \gamma_r \) are modelled as Gaussian distributions of average \( \mu_r \) and standard deviation \( \sigma_r \), computed upon a set of \( Q \) signatures \( \Gamma \), belonging to the same class \( p \in \zeta \). In the particular case where \( \gamma = 0 \) for all \( Q \) samples of \( \Gamma \), we consider it as "non informative" therefore modelling it as a uniform distribution. The following equation (4.12) represents the definition of the signature probabilistic model for Pearson and Topological approaches,

\[ P(\gamma_i) = \begin{cases} N(\mu_i, \sigma_i) , \text{if } \exists \gamma_i \in \Gamma_1: \gamma_i \neq 0 \\ \text{Uniform}(\gamma_i) , \text{if } \forall \gamma_i \in \Gamma_1: \gamma_i = 0 \end{cases} \] \hspace{1cm} (4.12)

computed for \( i = 1, \cdots, r \).
4.3.5 Signature Discriminant Evaluation

We propose to evaluate how discriminant each $\gamma_r$ is, via the comparison between class pairs $[X, Y] \in \zeta$. The Kullback-Leibler Divergence (KLD) is a metric in probability and information theory that quantifies the difference between two probability distributions through the following integral,

$$KLD_{X,Y}^r = \int_{-\infty}^{\infty} P(\gamma_r^X) \ln \left( \frac{P(\gamma_r^X)}{P(\gamma_r^Y)} \right) dX$$ \hspace{1cm} (4.13)

where $P(\gamma_r^X)$ and $P(\gamma_r^Y)$ are the learned conditional distributions for the same variable $\gamma_r \in \Gamma$, for classes $p = X$ and $p = Y \in \zeta$ respectively. The discriminative capability of variable $\gamma_r$ between classes $X$ and $Y$ is proportional to its absolute $KLD_{X,Y}^r$ value.

We present the minimum and maximum observed values of $KLD_{X,Y}^r$ amongst all class pairs $[X, Y] \in \zeta$. The average value across all variables $\gamma_r \in \Gamma$ is also computed for each class $p \in \zeta$. These metrics are defined as

$$\text{MIN} = \arg \min_{\forall \gamma_r \in \Gamma} (KLD_{X,Y}^r)$$
$$\text{MAX} = \arg \max_{\forall \gamma_r \in \Gamma} (KLD_{X,Y}^r)$$
$$\text{AVG} = \frac{1}{r} \sum_{i=1}^{r} KLD_{X,Y}^i$$ \hspace{1cm} (4.14)

computed upon all pairs $[X, Y] \in \zeta$. We compare both Pearson and Topological methods in Table 4.3.

Table 4.3: Main statistical KLD ratio measures for both signature approaches. Results do not take into consideration situations where $KLD=0$.

<table>
<thead>
<tr>
<th>Approach</th>
<th>MIN</th>
<th>MAX</th>
<th>AVG$_{MIN}$</th>
<th>AVG$_{MAX}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_{sig}$</td>
<td>0.002</td>
<td>0.016</td>
<td>0.002</td>
<td>0.011</td>
</tr>
<tr>
<td>$T_{sig}$</td>
<td>0.001</td>
<td>179.979</td>
<td>0.165</td>
<td>5.142</td>
</tr>
</tbody>
</table>

Results indicate approach $T_{sig}$ to outperform $P_{sig}$, for which the existing difference in MIN is seen as minor. An extended statistical assessment is presented as occurrence ratios, computed as equation (4.15).

$$\text{Ratio}_{min} = \frac{\Sigma (\text{min} KLD_B > \text{max} KLD_A)}{\Sigma KLD}$$
$$\text{Ratio}_{max} = \frac{\Sigma (\text{max} KLD_B > \text{max} KLD_A)}{\Sigma KLD}$$ \hspace{1cm} (4.15)

They refer to the percentage ratio for when the minimum and maximum KLD values in approach $T_{sig}$ are bigger than the maximum in $P_{sig}$. Values for the mini-
4.4. Methods for Person Identification

We develop different strategies for the person recognition classifier, considering \( \Gamma \) or \( \Gamma_G \), here named as parametric and compressed signature model respectively. A novel classifying measure is proposed as an alternative method to identify the most probable person according to the GMM signature approach, here named mixture signature model. Laban \( c_n \in \mathcal{L} \), signature \( \gamma_r \in \Gamma \) and person \( p \in \zeta \) are independent and identically distributed variables corresponding to abstraction variable spaces, laban, signature and identification respectively (see Figure 4.3).

4.4.1 Parametric Signature Model

The relevant variables for the Parametric Signature Model are identified in the Bayesian Program of Figure 4.4: signature vector variables \( \Gamma_n = [\gamma_r^n, \ldots, \gamma_r^n] \) (as defined in equation (4.11)), the Laban components \( c_n \) and the identity \( \zeta \). Let identity \( \zeta \) depend on each feature \( \gamma_r^n \in \Gamma_n \), where the superscript \( n \) indexes \( \gamma_r \) with respect to a given \( \Gamma_n \).
Bayesian Program: Parametric Signature Model

<table>
<thead>
<tr>
<th>Variables:</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\gamma^n_r \in \Gamma_n): Signature variables.</td>
</tr>
<tr>
<td>(c_n \in L): Laban quality variables.</td>
</tr>
<tr>
<td>(\zeta): Identity variable.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Decomposition:</th>
</tr>
</thead>
<tbody>
<tr>
<td>(P(\zeta, \gamma^1_r, \ldots, \gamma^n_r, \ldots, \gamma^r_r, c_1, \ldots, c_n) = P(c_n</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Formulation:</th>
</tr>
</thead>
<tbody>
<tr>
<td>(P(\zeta): Stochastic Matrix.)</td>
</tr>
<tr>
<td>(P(c_n</td>
</tr>
<tr>
<td>(P(c_n, \gamma^r_n</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Identification:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gaussian parameters (\mu) and (\sigma) based on training dataset (\Omega).</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Question:</th>
</tr>
</thead>
<tbody>
<tr>
<td>(P(\zeta</td>
</tr>
</tbody>
</table>

Figure 4.4: Bayesian Program for the Parametric Signature Model.

and that an intrinsic dependence of \(\gamma^r_n\) towards \(c_n\) also exits, because signatures are computed as a function \(f(c_n, \tau_n)\). These dependencies are reflected in the decomposition phase, where the joint distribution (left term of the equation) is decomposed into simpler conditional probability distributions (right terms). The real valued variables \(\gamma^r_n\), given a sufficient number of training samples, are expected to be normally distributed, therefore represented as Gaussian distributions. Identity \(\zeta\) represent our estimation for a given identity, and follows no particular known parametric form. Given that it is discrete, we represent it as a stochastic matrix. Inference over \(\zeta\) if given by weighing the relative uncertainty about signatures and Laban parameters, and is computationally calculated using a Maximum a Posteriori algorithm, where the continuous belief update yields

\[
\hat{\zeta}_{MAP} = \arg\max_{\zeta} \prod_{j=1}^{n} \prod_{i=1}^{r} \left( P(c_j | \gamma^j_i)P(c_j, \gamma^j_i | \zeta) \right) P(\zeta) \tag{4.16}
\]

and distributions \(P(c_n, \gamma^r_n | \zeta)\) and \(P(c_n | \gamma^r_n)\) are likelihood distributions, trained a priori with real experimental data.

### 4.4.2 Compressed Signature Model

The compressed signature model approach considers each signature variable \(\gamma_r \in \Gamma_G\) to depend on all \(c_n \in L\), since they are computed from a mixture of all \(\Gamma_n\). Similarly to the Parametric Signature Model, we present the Bayesian Program of the Compressed
Figure 4.5: Bayesian Program for the Compressed Signature Model.

version in Figure 4.5. The decomposition is obtained from the DAG in figure 4.3, formally establishing the aforementioned dependencies between variable spaces.

Analogously to what has been previously exposed and considering independent and identically distributed variables, inference is given by equation (4.17).

\[ \hat{\zeta}_{MAP} = \operatorname{argmax}_\zeta \prod_{i=1}^r \left( \prod_{j=1}^n P(c_j | \gamma_i) \right) P(\gamma_i | \zeta) P(\zeta) \]  

(4.17)

where the parameters \( \mu \) and \( \sigma \) of the likelihood Gaussian distributions are computed upon supervised learning.

4.4.3 Learning

Our approach to learn the identity models hypothesizes that the same person exhibits the same behavioural dynamics, even when performing different actions, e.g.: the symbol "sudden" may be identified, characterizing two completely different gestures, but its projection in Laban Space will be repeatable (similar) in both actions, which naturally generates similar signatures. This property suggests that signature variables are normally distributed. Let us address distributions \( P(\zeta, c_n | \gamma_r) \). Variables \( c_n \) and \( \zeta \) are discrete variables, which define the dimensions of a kernel map of probability distributions. Each variable \( \gamma_r \) has an indexed array \( T \), where each of the array elements yields

\[ T_{\zeta, c_n} = \mathcal{N}(\mu_{\zeta, c_n}, \sigma_{\zeta, c_n}) \]  

(4.18)
where $\mu_{\zeta,cn}$ and $\sigma_{\zeta,cn}$ are computed from a set of $Q$ samples, relatively to each $\gamma_r$ for all $\zeta = p$ and $cn = q$.

$$\mu_{\zeta,cn} = \frac{1}{Q} \sum_{\gamma_r \in Q} \gamma_r; \quad \sigma_{\zeta,cn} = \sqrt{\frac{1}{Q} \sum_{\gamma_r \in Q} (\gamma_r - \mu_{\zeta,cn})^2}$$ (4.19)

The distribution $P(\gamma_r | \zeta)$ has a similar build up, however average $\mu$ and standard deviation $\sigma$ are calculated without taking into account the dimension $cn$. The kernel map of distributions, for this case, has one dimension $\zeta$, which is represented as $T_{\zeta} = N(\mu_{\zeta}, \sigma_{\zeta})$.

### 4.4.4 Mixture Signature Model

An alternative, Bayesian-based approach is developed for classification using the the Mixture signature approach. Let an unknown person generate a sample $R \in \chi$. Consider a set of classifiable persons $p \in \zeta$, whose signature models are represented by a set of GMM parameters $\Theta_p = \{\phi_{p,i}, \mu_{p,i}, \Sigma_{p,i}\}$ for $i = 1, \ldots, k$ mixture components. The following equation measures the Mahalanobis distance from a single vector $R$ to a distribution of parameters $\{\mu_{p,k}, \Sigma_{p,k}\}$.

$$D^k_{p}(R, \mu_{p,k}; \Sigma_{p,k}) = (R - \mu_{p,k})^T \Sigma_{p,k}^{-1} (R - \mu_{p,k})$$ (4.20)

Computing the distance $D^k_{p}$ from $R$ to all components $k$ will result in a vector $v = \{D^1_{p}, \ldots, D^k_{p}\}$. We propose to compute the weighted Mahalanobis composed distance from a vector to a given dataset of GMM signature parameters $\Theta_p$ as in the following equation (4.21).

$$\eta(R_p) = \sum_{i=1}^{k} \phi_i \frac{1}{D^i_{p}}$$ (4.21)

The variable $(D^i_{p})^{-1}$ means that, smaller distances from vector $R$ to the components in the identity GMM, represent higher probabilities of $R$ to have been generated by that specific person. The parameter $\phi_k$, similarly to the Mixture Model, is a weight factor representing the relative significance of the measured distance. Given a composed distance from $R$ to all classes $p$, represented by $\Theta_i, i = 1, \ldots, p$, we propose the following probabilistic metric.

$$P(R_p) = \frac{\eta(R_p)}{\sum_{i=1}^{p} \eta(R_i)} \frac{\eta(R_p)}{P(\hat{R}_i)}$$ (4.22)
The element $P(R_p)$ represents the probability of a sample $R$ belonging to a class $p \in \zeta$, represented by $\Theta_p$, and $P(\hat{R}_p)$ the prior probability, computed at the previous iteration. The expected output is a probabilistic distribution vector $\Psi = [\psi_1, \ldots, \psi_p]$, where $\psi_p = P(R_p)$. The normalization factor in equation (4.22) ensures $\sum_{i=1}^{p} \psi_i = 1$.

The proposed methodology presents a classifying metric, yielding the probability of a multivariate vector belonging to a given class, modelled using multiple Gaussian distributions.

### 4.5 Experiments

The proposed methodologies are tested on the University of Coimbra 3-D (UC-3D) motion dataset (Appendix B). Our experiments using 3-D data are divided two-fold: in one case all actions are tested and, in the other we use only the gait-based actions (Walk and Run).

- In the first case, the classification models are assessed using cross validation. We employ a Leave-One-Out Cross-Validation (LOOCV), where a single activity is used for validation, and the remaining used as training data.
- For gait-based experiments, given that all actions are of the Gait category, we learn subsets of random samples from both walk and running actions, ensuring the existence of signature samples for a wider variety of $c_n$.

The per-frame classification results are presented in confusion tables, whilst per-sequence precision (as defined in equation (4.23) is present in bar-charts.

$$\text{precision(\%)} = \frac{\sum \text{Correctly Classified Sequences}}{\sum \text{Total Classified Sequences}} \times 100 \quad (4.23)$$

Posteriorly, we extend our validation to show that our method is also applicable to 2-D trajectories, by testing it on two popular action video datasets.

### 4.5.1 Experiments Using 3-D Data

The University of Coimbra 3-D Motion Dataset is a public domain database, developed in the context of this work. It is divided into single person and interactive (two person) activities, totalling 11 different actions, currently recorded by 13 different persons, identified from p01 to p13. Each action is performed 3 times by each person, in a total of 429 action sequences. The motion data is acquired using the high resolution
Figure 4.6: An example of available data from the University of Coimbra 3-D Motion Dataset, publicly available at [http://mrl.isr.uc.pt/experimentaldata/public/uc-3d/](http://mrl.isr.uc.pt/experimentaldata/public/uc-3d/).

(120Hz) MVN motion capture suit from XSens, equipped with 17 Inertial Measuring Units. This data is published using the proprietary MVN XSens XML format files. Simultaneously, we acquire RGB and depth image sequences (30Hz) using a Microsoft Kinect, which are also made available. A sequence for a Walk movement is presented in Figure 4.6 along with a sample of RGB and depth image frames. The MVN suit also provides linear, angular and gravitational acceleration as well as velocities for each sensor.

Experimental results using all actions from the UC-3D dataset are summarized in Table 4.7. The parametric model shows a better identity classification performance when compared to the compressed version. Results indicate that the two stage signature compression applying SVD exhibits less discriminant capabilities, in our case, considering that variables \(\gamma_r\) are modelled using Gaussian distributions. A thorough, step-by-step result analysis showed the discriminant ability to be lost at the stage where the SVD algorithm is applied to matrix \(A_G\), whose first row values are significantly larger that those of the remaining rows, i.e. \(M_{1,j} \gg 1\) and \(-1 \leq M_{i,j} \leq 1\), \(\forall i \neq 1\). The generated signature \(\Gamma_G\) showed similar values for \(\gamma_r \in \Gamma_G\) across all learned identity classes \(p \in \zeta\) and, for those specific cases, the posterior is unable to converge to the correct identity.

Because experimental results are presented as a sum of all LOOCV experiments, each leaving out a different action from the training set, we demonstrate that the proposed method can accurately identify persons independently of the performed actions.
Figure 4.7: Per-frame confusion tables for person identification, using all actions, on the UC-3D motion dataset.
The bar chart in Figure 4.8 presents the per-sequence precision comparison for experiments with the Parametric approach using all actions and the ones using only Gait-based actions. These results show that, independently of which action category is used, we can accurately identify the different persons, with an average precision of 97.84% and 98.67% for non-gait and gait-based approaches respectively.

Figure 4.8: Classification precision for experimental set-ups using both non-gait and gait-based actions using the Parametric Model.

The mixture signature model encompasses a random initialization process, reason for which we have tested the classification experiments over a $Q = 100$ number of trials. At the beginning of each trial, the new set of parameters $\Theta$ is learned and classified using LOOCV. The parameter estimation uses the EM algorithm, forced to iterate until the weights converge to $\phi_t - \phi_{t-1} < \text{threshold}$. Precision results are presented in terms of their average over $Q$.

As Figure 4.7c shows, identity classification accuracies range from 54.02% and 98.89%, which are under the results observed for the other two approaches. These results demonstrate that using signature techniques to encode data in Laban Space, have the advantage of increasing its discriminant properties.

In the graph of Figure 4.9, we show the impact of the number of components $k$ in the classification global precision. Results improve with the number of components up to $k = 5$, which visually defines a local optimum value upon which accuracy stabilises, showing no further improvement.

With this approach, Gait-based actions have a slightly better performance than non-Gait (See Figure 4.10). This is justified by the reduced number of actions, leading to the signature models be better separated, as opposed to using multiple actions, where models are more likely to overlap. Close accuracies for both approaches show that there is little or no difference when using Gait-based or non-Gait actions for person identification, reinforcing the property of activity invariance characterizing the
4.5. Experiments

Figure 4.9: Per-sequence classification precision for all identities considering different number of components $k$ in the Mixture Signature Model $\Theta_k$.

Figure 4.10: Per-sequence classification precision for non-gait and gait-based actions using the Mixture Model.

4.5.2 Experiments Using 2-D Data

The experimental set-up is also validated using 2-D trajectories, demonstrating the proposed method robustness. To this purpose, two public and acknowledgeable motion datasets are used.

- The *KTH (Kungliga Tekniska Högskolan) Royal Institute of Technology 2-D Motion Dataset* [SLC04b], from which $\approx250$ motion sequences from different activities are used, performed by 10 different persons.

- The *WZ (Weizmann Institute) 2-D Motion Dataset* [GBS+07] contributes with 9 different persons executing 9 different actions, in a total of 81 activity sequences.

Their selection is motivated by their popularity and the existent variety of action sequences, from upper body gestures to gait activities. These video-based datasets were used in our previous research work [SD], were body parts are roughly tracked, and whose sequence duration ranges from $2 − 5$ seconds. Note that for these 2-D
experiments, we are using the Parametric Signature Model (best in 3-D), comparing it to the Mixture Model in all actions.

### 4.5.2.1 Experiments on KTH Motion Dataset

The confusion table in Figure 4.11a shows the identification accuracy per-frame for the KTH motion dataset, using all actions in a LOOCV approach. Results show an accurate model, except for performer '02'. Detailed analysis, revealed the signatures of both performers '01' and '02' to be non-discriminant. Their mean $\mu$ values for $\gamma_r$ are similar and the standard deviation $\sigma$ values are bigger for '01' than for '02', which makes this identity easily divergent to '01' in the presence of noise. The global accuracy is 82.80%, increasing to 91.11% if we disregard identity '02'.

We present the per-sequence precision using the non-gait action experiments in Figure 4.11b for the Parametric and the Mixture (benchmark) Models. Results show that identity classification is consistent for the Parametric, with a global precision over 90%. However, the Mixture Signature Model showed low accuracy, due to the excessive overlap of different person’s characteristics in Laban Space.

![Confusion Table](image)

(a) Per-frame confusion table using Parametric Signature Model.

![Precision Accuracy](image)

(b) Per-sequence precision using the Parametric Signature approach in experimental cases for non-gait and gait-based actions.

Figure 4.11: Results for identification on KTH dataset.
4.5.2.2 Experiments on WZ Motion Dataset

The results for WZ data also exhibit good classification performance (Table 4.12a). During a given sequence, classifying each subset \( \hat{\omega}_p \) is harder than in KTH dataset, a fact visible by the low per-frame accuracy. However, per-sequence classification (Figure 4.12b) still holds highly precise identity classification, showing an overall ratio of 95.06% across all action sequences \( \omega_p \in \Omega \). The slower convergence when compared to the KTH dataset is, in part, justified by the different acquisition frequencies, reflected in the number of classified samples per second.

A detailed analysis, allows concluding that 'Ido' presents the highest rate of misclassified sequences, which can be explained by the reduced amount of sequences \( \omega_p=Ido \) when compared to the remaining performers. Actor 'Ira' also showed an under par, per-frame accuracy, due to its signatures being similar to the ones generated for 'Denis'.

<table>
<thead>
<tr>
<th></th>
<th>Daria</th>
<th>Denis</th>
<th>Eli</th>
<th>Ido</th>
<th>Ira</th>
<th>Lena</th>
<th>Lyova</th>
<th>Moshe</th>
<th>Shahar</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.83</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.14</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>2</td>
<td>0.00</td>
<td>0.96</td>
<td>0.01</td>
<td>0.00</td>
<td>0.00</td>
<td>0.01</td>
<td>0.00</td>
<td>0.01</td>
<td>0.00</td>
</tr>
<tr>
<td>3</td>
<td>0.00</td>
<td>0.06</td>
<td>0.60</td>
<td>0.00</td>
<td>0.00</td>
<td>0.01</td>
<td>0.29</td>
<td>0.04</td>
<td>0.00</td>
</tr>
<tr>
<td>4</td>
<td>0.00</td>
<td>0.07</td>
<td>0.03</td>
<td>0.26</td>
<td>0.00</td>
<td>0.01</td>
<td>0.55</td>
<td>0.02</td>
<td>0.06</td>
</tr>
<tr>
<td>5</td>
<td>0.05</td>
<td>0.38</td>
<td>0.00</td>
<td>0.00</td>
<td>0.49</td>
<td>0.00</td>
<td>0.07</td>
<td>0.01</td>
<td>0.00</td>
</tr>
<tr>
<td>6</td>
<td>0.00</td>
<td>0.30</td>
<td>0.01</td>
<td>0.00</td>
<td>0.00</td>
<td>0.03</td>
<td>0.09</td>
<td>0.01</td>
<td>0.90</td>
</tr>
<tr>
<td>7</td>
<td>0.00</td>
<td>0.28</td>
<td>0.01</td>
<td>0.00</td>
<td>0.00</td>
<td>0.01</td>
<td>0.01</td>
<td>0.02</td>
<td>0.00</td>
</tr>
<tr>
<td>8</td>
<td>0.00</td>
<td>0.23</td>
<td>0.01</td>
<td>0.00</td>
<td>0.00</td>
<td>0.01</td>
<td>0.05</td>
<td>0.70</td>
<td>0.00</td>
</tr>
<tr>
<td>9</td>
<td>0.00</td>
<td>0.07</td>
<td>0.01</td>
<td>0.00</td>
<td>0.00</td>
<td>0.01</td>
<td>0.11</td>
<td>0.02</td>
<td>0.78</td>
</tr>
</tbody>
</table>

(a) Per-frame confusion table using Parametric Signature Model.

![Graph](image)

(b) Per-sequence precision using the Parametric Signature approach in experimental cases for non-gait and gait-based actions.

Figure 4.12: Results for identification on Weizmann dataset.

As in the previous experiments using the KTH dataset, per-sequence results exhibit good identification precisions (see Figure 4.12b) for Parametric Signature Model,
demonstrating our identification methodology can be effectively applied to 2-D based data, using short sequences. Results for Mixture Signature Model also show poor performance, due to the same reasons explained for KTH dataset experiments.

4.5.3 Discussion

In Table 4, we present a summary of achieved results and experimental conditions. Results show that the Topological encoding together with the Parametric Signature Model to exhibit the best overall performance in our UC-3D motion database. The Compressed Model also presents interesting results. Comparison of both methods with the Mixture approach experiments, indicate advantages about using signatures to encode Laban Space information. For the 2-D experiments, at the Laban symbolic analysis step, components were classified with converging probability values for most persons, a fact that interfered with the encoding process.

Overall classification accuracies show that our model is, in fact, capable of identifying persons without restricting them to perform a previously known action. The proposed identification methods are validated using a Leave-One-Out Cross-Validation approach, showing a high ratio of correctly classified identities, when classifying actions that have not been used to train the model. This fact is true for both 3-D and 2-D data experiments. In the specific case of action-based person identification using the Weizmann Institute Motion Dataset, our methodologies exhibit a higher precision than the currently best performance using the same action dataset, 95.06% comparing to 93.28% in [LHZS12].

The experimental set-up using 3-D data, has been extended to perform identification

<table>
<thead>
<tr>
<th></th>
<th>NON-GAIT LOOCV</th>
<th>GAIT-BASED Random Sampling</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P (%)</td>
<td>C (%)</td>
</tr>
<tr>
<td>3-D</td>
<td>UC-3D</td>
<td>97.84%</td>
</tr>
<tr>
<td>2-D</td>
<td>KTH</td>
<td>91.11%</td>
</tr>
<tr>
<td></td>
<td>WZ</td>
<td>95.06%</td>
</tr>
</tbody>
</table>

**Acronyms:** LOOCV (Leave-One-Out Cross Validation; UC-3D (University of Coimbra 3-D Motion Dataset; KTH (KTH Motion Dataset); WZ (Weizmann Institute Motion Dataset); P (Parametric Signature Model; C (Compressed Signature Model); M (Mixture Signature Model).

**Note:** For the 2-D experiments, only the best performing Signature Model was tested for comparison with the benchmark approach, Mixture Signature Model.
using only a subcategory of actions. The near perfect accuracy, demonstrates that our methodology’s can also be applicable with success to gait-based scenarios.

4.6 Conclusions and Future Work

In this manuscript, we propose a person recognition model using motion signatures, generated from 3-D trajectories in random actions. Results demonstrate the model to be action invariant, not restricting performers to execute specific movements in order to be correctly identified. Highly accurate identity convergence in short action sequences, indicates the proposed framework to fulfil real-time requirements for most applications. The developed signature methods demonstrate to encode unique expressive characteristics, defined upon symbolic Labanotation, which show to be discriminant with respect to whom is performing a given movement. The achieved results allow concluding that the encoding methods increase accuracy when compared GMM-based approach in the symbolic descriptive space. Additionally, our model has also been validated using 2-D trajectories.

Two main future research challenges are identified, which would extended this work, envisioning two applications: (1) monitoring restricted access environments and (2) identifying populations with similar behaviour patterns. Restricted environments usually encompass a limited number of allowed personnel. These are usually video-surveilled, where the major challenge would be segmenting body part trajectories from the acquired images, or alternatively associate image-based features (e.g. Silhouettes) to Laban symbolic descriptors. Solving this problem, would make our framework straightforward applicable. Grouping large populations by behaviour patterns would implicate a modification the proposed identification model. Rather than learning identities, the association process would refer to Laban symbolic descriptors and behaviour classes. A similar concept was already applied to gestures in [SD11b] and interpersonal behaviours [KD13]. Here, the challenge would be verifying if the same categorical classes for behaviours would generate similar Laban descriptions for large groups of different persons.

We would like to highlight another key contribution of this work, the new University of Coimbra 3-D Motion Dataset, which at the moment encompasses 11 different activities performed by 13 different performers. It is publicly available, encompassing data from different sources: the XSens MVN suit provides Inertial Measuring Unit data; image sequences from a Microsoft Kinect, acquire both RGB and Depth images.
This data makes it easier for different approaches on action or person recognition to be more fairly benchmarked.
Chapter 5

Case Study:
Intelligent Video-Surveillance

5.1 Introduction

"Video cameras are increasingly prevalent in society in both public and private spaces. At the same time, the quality of video surveillance continues to improve. This is especially true of intelligent video surveillance technology, which can recognize or track objects as well as identify human faces and behaviour patterns." - Held in [HKMST12].

In our previous chapters, we have presented two frameworks for trajectory-based action analysis and for activity invariant person identification. Our findings show a highly accurate framework which, using trajectory-based motion signatures, is capable of classifying between different person identities. This is a study of special interest in the area of video-surveillance. However, as has been opportunely stated, to make our framework straightforward applicable, there is an unsolved challenge to be addressed: associating image-based features to the set of activity invariant descriptors based on Laban Movement Analysis (LMA) [BL80].

5.1.1 Related work

Person Identification research can be broadly divided in two distinct categories: invasive and non-invasive biometrics. The majority of existent biometric works focus on fingerprint, iris or face analysis, which are invasive techniques, requiring some sort of cooperative behaviour by the identified person. Non-invasive research addresses
motion analysis in order to discriminate between different persons, focusing on gait as the primary activity, e.g. [LMS04, HL09, LLC07, BCD02]. Kobayashi and Otsu [KO04], more recently Iosifidis et al. and Santos and Dias address activities other than gait, which present interesting results in the area of person recognition. Iosifidis et al. [ITP12] address eating and drinking, while Santos and Dias [SD] present an activity invariant solution, testing up to 9 different activities. An overview on the state of the art shows that action-based person recognition systems are still an open problem, which is increasingly receiving attention from the scientific community.

LMA is a motion notation language, developed in the context of dance and choreography by Rudolf Laban, which in the past decade, has found its way in the field of computational motion analysis. A kinematic based Expressive MOTion Engine (EMOTE) has been developed by Norman Badler’s Group [CCZB00]. Swaminathan et al. [STM+09] propose a probabilistic model which uses a body kinematic model and joint velocities to model Shape qualities. Santos and Dias address trajectory-based Laban models [SD11b], while Kamrad and Dias focus on body part acceleration signals for Laban-based behaviour understanding [KD13]. Kim et al use an RGB-D camera to extract joint velocities to model the Effort component [KPL+13]. Zhao [Zha02] and Rett [Ret09] both investigated LMA in the context of communicative gestures. Zhao explored inverse kinematics, while Rett exploited vectorial information of limb velocity and acceleration signals. The use of classical visual cues has not yet been addressed, from which an unsolved challenge is identified and pointed as a valuable contribution.

5.1.2 Our Approach

This chapter is presented as a case study, contributing to computational LMA and Person Recognition areas. As previously argued, most existent applications are based on biometric properties such as fingerprint, eye scanning or face recognition technologies. Non-invasive, motion-based biometric systems are still very dependent on a specific activity, gait. This work is an extended solution which integrates both themes from chapters 3 and 4. We address two unsolved problems in the current state of the art: (1) Existent LMA models are mostly based on kinematic or motion dynamic features, whereas classic visual cues have not yet been applied to this purpose; (2) Despite the advances in motion based person recognition frameworks in recent years, gait is still the dominant exploited activity. We illustrate our approach in Figure 5.1. We propose to acquire a set of image sequences from a calibrated camera network, from which silhouettes are segmented. Given their high dimensionality, we propose an alternative
representation, which combines two signal processing techniques: Principal Component Analysis and Fast Fourier Transform. Posteriorly, we apply a supervised learning strategy to associate the generated silhouette features to sets of training data, which are manually labelled with the dominant LMA characteristics. Upon learning the Visual LMA models, we use a classifier based on Dynamic Bayesian Networks, which is applied to autonomously analyse motion sequences using LMA symbolic descriptors. This motion analysis framework aims to be integrated with an adapted version of the person identification system developed in chapter 4. By solving the aforementioned challenge, we identify the following major contributions: (1) A LMA model based on visual cues; (2) Establish an adapted Laban signature model for action-based person identification. Validation is done on two publicly available datasets, through adequate evaluation metrics. Additional experimental information will be found on a support web page.

5.2 Visual Cues and Variables

In this work, we are encoding classical visual features, which are then used to learn motion models. For the purpose of extracting silhouettes in our experimental set-up, we have applied existing image processing algorithms, from which we mainly enumerate the popular Gaussian filtering, background subtraction and Canny Edge Detector.

http://www.isr.uc.pt/~luis
5.2.1 Silhouette Features

Consider a binary image $I_{BW}$, containing a silhouette contour $P$, which is represented by the coordinates of every white pixel, such that:

$$P = \left\{ (u_1, v_1), \ldots, (u_s, v_s) \right\}, \forall (u_s, v_s) \in I_{BW} : I_{BW}(u_s, v_s) = 1$$  \hspace{1cm} (5.1)

Given the potential high dimensionality of $P$, we use an alternative representation based on Principal Component Analysis (PCA) algorithm. PCA is used to uniquely characterize the internal geometrical structure of the scatter data, which best explains its variance. The defined orthogonal PCA space is here hypothesized as a set of geometrical cues. In practice, we use the eigenvector coordinates $v_r = (x_r, y_r)$, (which define the axis in the PCA component space) as independent silhouette features for a static image $I_{BW}^n$ at instant $n$. We compute the ratio between the first and second eigenvalues as $\lambda_1/\lambda_2$, which is then multiplied by the first component coordinates, such that $v'_1 = v_1(\lambda_1/\lambda_2)$. This additional step makes the first eigenvector to represent implicit eigenvalue information, which mitigates the impact of silhouette scale. The second component coordinates are used directly.

$$\hat{P}^t = \text{pca}(P) : \hat{P} = [v'_1 \quad v_2] \equiv [p^n_1, \ldots, p^n_4]$$  \hspace{1cm} (5.2)

This procedure is applied twice, in both upper and lower body sections of the silhouette, which is roughly divided using the information about its center of mass. We consider this approach to provide better information in cases where actions are dominantly performed by one of the body halves (e.g. run or wave).

A part of LMA components address movement expressiveness, which describe motion dynamics. These properties require temporal characterization, rather than using a single image. Let an image sequence $I$ be divided into sub-sequences of duration $\hat{n}$, such that $\hat{I} = \{I_{n-\hat{n}}, \ldots, I_n\}$. For each $\hat{I}$ we have a corresponding times series for each feature $p_j$, such that $p_j[\hat{n}] = (p_j^{n-\hat{n}}, \ldots, p_j^n)$. To characterize motion dynamics we apply the Fourier Transform, a popular technique to analyse time series. We propose represent $p_j[\hat{n}]$ in the frequency domain (eq. (5.3)), from which the computed coefficients constitute a set of features, implicitly representing the dynamics of $\hat{I}$.

$$P_j(\omega) = \sum_{n} p_j[\hat{n}] e^{-i\omega n}$$  \hspace{1cm} (5.3)
5.2. Visual Cues and Variables

(a) Examples of Static features.

(b) Processing diagram of a sequence \(\hat{I}\).

(c) Dynamic features for \(p_1(\hat{t})\) and \(p_3(\hat{t})\), in 4 different perspectives.

Figure 5.2: Static (a), Dynamic (b) visual cues examples and an example of Dynamic for 4 different perspectives.

Within the set of Fourier coefficients, we select the maximum value, such that \(F = \{f_1, \cdots, f_j\} : f_j \in \max P_j(\omega)\), and its correspondent fundamental frequency index \(n\). The last considered feature is based on the silhouette displacement vector \(\vec{d}\) for two consecutive images, such that \(\vec{d}_n = (u^n_n - u^{n-1}_c, v^n_n - v^{n-1}_c)\), where \((u^n_c, v^n_c)\) is the center of mass at instant \(n\). The displacement feature is \(\theta_n = \tan^{-1}(v^n_n - v^{n-1}_c, u^n_n - u^{n-1}_c)\), which will be specially relevant for components which are direction based, rather than orientation, being complementary to the information given by the PCA features.

5.2.2 Experimental Results

In this section, for simplicity and easier visualization, results are presented for a representative subset of features \(\{x'_1, y'_1, \lambda_1/\lambda_2\} \in \hat{P}, \theta\) and \(\{f_1, f_2\} \in F\), for the different gestures \(g_{\text{index}}\) in the selected datasets. Figure 5.3 presents the mean of measured values for each feature in each different gesture. Results show features to exhibit discriminant capabilities with respect to different gestures and, consequently, different
properties of the performed motion. These differences indicate that features encompass interesting geometric and dynamic properties to be explored by our model, in Section 5.3. We also highlight an interesting fact observed during a preliminary study on action observation from different perspectives. The bottom image in Figure 5.2 shows that silhouettes are naturally different, however their feature dynamic behaviour is similar in the various perspectives.

5.3 Visual Laban Model

The computed feature variables are used to define the LMA model, which is learned using a supervised mixture model approach. Each component will be modelled from adequate feature types. A Bayesian classifier is used to evaluate the model with respect to its analysis capabilities, which is required to accurately characterize short activity sequences.

Definitions: The proposed LMA model is a probabilistic representation of its components, learned from a set of observable visual-based cues \( \{f_i, \theta, p_i\} \), which are extracted from a combination of static images and sequences. It is parametrised into as many sub-models as the number of components \( c_n \), where the initial hypothesis for the space state is defined in equation (5.4) based on the concepts of Labanotation [Gue70] (details in subsection 5.3.1). In light of the conclusions withdrawn from the feature experiments, the model should be prepared to support an arbitrary number of cameras \( S_m \), therefore bear in mind the proposed model is assumed to be running independently for each
different sensor. In practice, each camera will show a different silhouette perspective.

\[
\mathcal{L} = \begin{cases}
  c_1 : \text{Effort Time} & \in \{\text{sudden, sustained}\} \\
  c_2 : \text{Effort Space} & \in \{\text{direct, indirect}\} \\
  c_3 : \text{Effort Flow} & \in \{\text{free, bound}\} \\
  c_4 : \text{Shape Form} & \in \{\text{wall/pin, ball}\} \\
  c_5 : \text{Direction Shape} & \in \{\text{spoke, arc}\} \\
  c_6 : \text{Shape X} & \in \{\text{spreading, enclosing}\} \\
  c_7 : \text{Shape Y} & \in \{\text{rising, sinking}\} \\
  c_8 : \text{Shape Z} & \in \{\text{advancing, retreating}\}
\end{cases}
\]  

(5.4)

Formulation:

The Bayesian Program in Figure 5.4 shows our proposed visual LMA model. The first step is to state and define the relevant variables:

- \( \mathcal{L} = \{c_n \equiv \{q_1, q_2\}\} \) is a variable denoting a LMA component representing a specific motion characteristic, admitting two mutually exclusive states, as defined in equation (5.4). States are assumed to be dynamically propagating through a given sequence.

- \( F = \{f_i \in \mathbb{R}_0^+\} \) are a set of random variables representing dynamic information.

<table>
<thead>
<tr>
<th>Variables:</th>
</tr>
</thead>
<tbody>
<tr>
<td>( c_n \in \mathcal{L} ): represents the ( n )th Laban component;</td>
</tr>
<tr>
<td>( p_j \in \hat{P} ): represents the ( j )th static silhouette feature;</td>
</tr>
<tr>
<td>( f_i \in F ): the ( i )th dynamic silhouette feature;</td>
</tr>
<tr>
<td>( \theta \in [-\pi, \pi] ): represents the relative displacement of the silhouette;</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Decomposition:</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P(c_n, p_j, f_i, \theta) = P(c_n)P(p_j</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Formulation:</th>
</tr>
</thead>
</table>
| \( P(c_n) = \begin{cases}
  \text{uniform, } t = 0 \\
  P_{t-1}(c_n), \quad t > 0
\end{cases} \) |
| \( P(p_j, f_i, \theta|c_n) \): independent Gaussian distributions; |

<table>
<thead>
<tr>
<th>Identification:</th>
</tr>
</thead>
<tbody>
<tr>
<td>none.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Question:</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P(c_n</td>
</tr>
</tbody>
</table>

Figure 5.4: Bayesian Program for estimating the most probable state for each Laban component \( c_n \). It considers a single component, for simplicity purposes.
from an image sequence $\hat{I}$.

- $\hat{P} = \{p_j \equiv \{u_1, \cdots, u_x\}\}$ are a set of random variables representing the geometrical information over static silhouettes $\in I$. It is discretised into a number of $x$ equidistant intervals, representing a possible state (or bin) $u_x$.

- $\theta \in [-\pi, \pi]$ is a random variable which represents the displacement orientation between to consecutive silhouettes $P^t$ and $P^{t-1}$.

The model decomposition is detailed in the Bayesian Program in Figure 5.4, from which the questions are formulated. To estimate of a given state in the model, Bayesian inference is applied. Inference in this work considers feature variables to be independent and identically distributed, and can be formulated, from a Maximum a Posteriori perspective, as follows:

$$ P(c_n|p_j, f_i, \theta) \propto P(c_n) \Pi_{i,j} P(f_i, p_j, \theta|c_n = q_j) $$
$$ \propto P(c_n) P(\theta|c_n) \Pi_i P(f_i|c_n) \Pi_j P(p_j|c_n) $$

5.3.1 Laban Component Models and Feature Types

LMA is a symbolic notation language used for a comprehensive understanding of human motion, which has the unique capability to describe expressiveness. It was developed around the concepts of movement notation and integrates studies from anatomy, kinesiology, psychology and Labanotation. It was originally designed for dance choreography, and is today one of the most used systems for movement analysis in a wide range of areas. It defines movement as an intentional process of patterned and orderly changes, which are better studied if divided in different levels. It is divided in four main different components, each describing a different motion property, using an adequate symbolic grammar, Labanotation [Gue70]. Body and Space components describe the structural or physical properties of the body and spacial patterns along with body part pathways respectively. The Effort component addresses dynamics and inner intention, while the Shape deals with the connections between the body and space and the changes in body shape. We hypothesize a generalization of LMA to full body analysis rather than specific body parts, which may, for some activities, present undefined states (e.g. rising symbolizes a state which may not even be observable for some activities). In this work, we model the two components addressing motion expressiveness, Effort and Shape, which are always observable in any movement sequence [Zha02].

Effort addresses the dynamic properties of motion with respect to inner intention. It is
5.3. Visual Laban Model

divided into four different qualities: Space, Time, Weight and Flow. We discard Effort Weight as it is usually associated to strength and we consider that visual cues are not adequate for its characterization.

\(c_1\) Effort Time characterizes the cognitive process of decision, which is tightly related to time. Therefore it is associated to dynamic features \(f_i\) and the Bayesian question yields \(P(c_1|f_i)\).

\(c_2\) Effort Space is focused on the attention with respect to orientation with a purpose. We hypothesize this to be a combination of geometric, dynamic and orientation features, such that \(P(c_2|f_i,p_j,\theta)\).

\(c_3\) Effort Flow characterizes motion continuity, which is related to performance along time and whether or not it is contained to a single action. Dynamic and displacement are the selected feature types, hence \(P(c_3|f_i,\theta)\).

Shape emerges from the Body and Space components. As the name implies it address the geometrical form the body takes and how it changes in time. It is used to integrate different categories into movement.

\(c_4\) Shape Form is one of the categories of Shape, is as the name implies is the form the body takes, which is mostly geometric, such that \(P(c_4|p_j)\). We simplified the space state considering wall and pin as a single state, where the performer is dominantly standing.

\(c_5\) Directional Shape represents the existent relation between the body and the environment. It divides movements into spoke-like (e.g. point) and arc-like (e.g. waving bye bye). It is mostly geometric and \(P(c_5|p_j)\).

\(c_6,7,8\) Shape also has qualities, which describe body extensions or how its form changes with respect to specific spacial orientations. Geometric features are relevant, as is the way these change in time, with respect to the body center. Thus, the Bayesian question formulates as \(P(c_6,c_7,c_8|f_i,p_j,\theta)\).

5.3.2 Learning the Models from Experimental Data

The likelihood distributions \(P(\theta,p_j,f_i|c_n)\) represent the actual LMA model based on sets of training motion sequences. The first step in the learning process is the manual annotation of the image sequences that will be used to train the model. Let a sequence
be labelled with a set of dominant LMA states, one for each corresponding component \( c_n \). For each \( \hat{I} \in I \) we compute a set of features \( \{p_j, f_i, \theta \} \). For every component \( c_n \), we cluster the different features with respect to each possible state, i.e. two different sets of features are associated to classes \( q_1 \) and \( q_2 \). Hence, considering features as independent and identically distributed, for a given state \( q_k \) we have a likelihood distribution defined from the association process as:

\[
P(\beta|c_n = q_k) = \mathcal{N}(\mu_{\beta}, \sigma_{\beta}), \beta \in \{f_i, \theta\}
\]  

(5.6)

In the specific case of \( p_j \) we have discretized it in a number \( x \) equidistant intervals, between observed range for \( p_j \). Each interval corresponds to a single possible state \( u \), such that, \( p_j = \{u_1, \cdots, u_x\} \). Hence, the likelihoods using this variable formulate as a stochastic matrix \( M_{x,r}^{c_n} \), each cell’s probability is given by:

\[
P([p_j = u_x]|c_n = q_k) = \frac{\sum \text{observations } u_x \text{ for state } q_k}{\sum \text{total observations } u \text{ for state } q_k}
\]  

(5.7)

### 5.3.3 Experiments on Motion Characterization with Laban Notation

The top two rows of Table 5.1 summarize the percentage of correctly identified states per component when compared to ground truth annotation and the average model

<table>
<thead>
<tr>
<th>c1</th>
<th>c2</th>
<th>c3</th>
<th>c4</th>
<th>c5</th>
<th>c6</th>
<th>c7</th>
<th>c8</th>
</tr>
</thead>
<tbody>
<tr>
<td>f.a.r.%</td>
<td>94.02</td>
<td>93.17</td>
<td>91.43</td>
<td>67.02</td>
<td>80.34</td>
<td>81.91</td>
<td>63.17</td>
</tr>
<tr>
<td>a.m.c.%</td>
<td>87.42</td>
<td>93.18</td>
<td>90.58</td>
<td>82.88</td>
<td>89.24</td>
<td>98.91</td>
<td>90.46</td>
</tr>
<tr>
<td>jump</td>
<td>sudden</td>
<td>direct</td>
<td>free</td>
<td>spoke</td>
<td>pin</td>
<td>n.d.</td>
<td>rising</td>
</tr>
<tr>
<td>pjump</td>
<td>sudden</td>
<td>direct</td>
<td>free</td>
<td>spoke</td>
<td>pin</td>
<td>n.d.</td>
<td>rising</td>
</tr>
<tr>
<td>run</td>
<td>sudden</td>
<td>direct</td>
<td>free</td>
<td>spoke</td>
<td>pin</td>
<td>enclosing</td>
<td>rising</td>
</tr>
<tr>
<td>side</td>
<td>sudden</td>
<td>direct</td>
<td>free</td>
<td>spoke</td>
<td>pin</td>
<td>spreading</td>
<td>n.d.</td>
</tr>
<tr>
<td>skip</td>
<td>sudden</td>
<td>direct</td>
<td>free</td>
<td>spoke</td>
<td>pin</td>
<td>n.d.</td>
<td>rising</td>
</tr>
<tr>
<td>walk</td>
<td>sudden</td>
<td>direct</td>
<td>free</td>
<td>spoke</td>
<td>pin</td>
<td>n.d.</td>
<td>rising</td>
</tr>
<tr>
<td>wave1</td>
<td>sustained</td>
<td>indirect</td>
<td>free</td>
<td>arc</td>
<td>wall</td>
<td>spreading</td>
<td>n.d.</td>
</tr>
<tr>
<td>wave2</td>
<td>sustained</td>
<td>indirect</td>
<td>free</td>
<td>arc</td>
<td>wall</td>
<td>spreading</td>
<td>n.d.</td>
</tr>
<tr>
<td>boxing</td>
<td>sudden</td>
<td>direct</td>
<td>bound</td>
<td>spoke</td>
<td>n.d.</td>
<td>spreading</td>
<td>n.d.</td>
</tr>
<tr>
<td>handwaving</td>
<td>sustained</td>
<td>indirect</td>
<td>free</td>
<td>n.d.</td>
<td>ball</td>
<td>spreading</td>
<td>rising</td>
</tr>
<tr>
<td>running</td>
<td>n.d.</td>
<td>n.d.</td>
<td>free</td>
<td>spoke</td>
<td>n.d.</td>
<td>n.d.</td>
<td>n.d.</td>
</tr>
<tr>
<td>walking</td>
<td>n.d.</td>
<td>n.d.</td>
<td>free</td>
<td>spoke</td>
<td>pin</td>
<td>enclosing</td>
<td>n.d.</td>
</tr>
</tbody>
</table>

Table 5.1: Classified dominant LMA states. Each state is considered dominant, if and only if they are classified in at least 2/3 of the frames in a sequence \( I \). The acronyms \( f.a.r \) and \( a.m.c. \) stand for Frame Accuracy Ratio and Average Model Confidence respectively. The acronym \( n.d. \) stands for Non-Dominant state.
confidence, i.e. the average probability with which states are classified. The average f.a.r. percentage is 87.67%, whereas the average confidence is over 90%. However, for component $c_8$, the accuracy is low with high confidence, which means the model is converging to a false positive for that particular characteristic. Isolating $c_8$ from the remainder, results show an accurate model when comparing ground truth annotation and estimated LMA states, showing that visual features can be applied for characterizing motion sequences using LMA descriptors. To further extend our result analysis, Table 5.1 additionally presents the dominant characteristics for each of the analysed activities. Gestures are usually characterized by specific properties, which using our model, are accordingly represented by dominant symbols, e.g. actions traditionally associated to fast movement are associated to the symbol sudden. Figure 5.5 presents an example of the generated symbolic output of the visual-based LMA model.

5.4 Application on Person Recognition

In chapter 3, we argue that LMA can be generalized, at least symbolically. Results from that and this chapter show LMA states to be repeatable for similar actions when performed by different persons. However, the confidence with which the model classifies each state, differed from person to person. This property indicates that LMA space can be discriminant with respect to whom is performing an observable sequence. The presented identification model had to be adapted to cope with the generalization characteristics of the proposed visual LMA model.
5.4.1 Laban Motion Signature and Identification Models

*Laban Space Definition:* We briefly summarize the signature encoding process which presented the best results in the previous chapter. Let LMA Space $\chi \in \mathbb{R}^n$ be a $n$-dimensional unified representation for all LMA variables $c_n$. Consider a vector $R \in \chi : R = (\tau_1, \cdots, \tau_n)$ representing a combination of LMA properties for a motion sub-sequence, where $\tau_n = P(c_n = q_1)$. Consider each LMA variable as a topological node, measuring inter-node distances as $d_{i,j} = \tau_i - \tau_j$. The signature is generated by defining the adjacency matrix $A$ of the topological graph and computing its single value decomposition. The signature variables $\Gamma = \{\gamma_i\}$ are independent and defined from the computed eigenvalues and corresponding eigenvectors.

*Signature and Person Identification Model:* In chapter 3, Laban is applied to analyse different body parts rather than the body as a whole, which naturally augments the signature discriminant capabilities. The initial experiments applying our visual LMA model to the identification model in chapter 4 presented poor recognition accuracy. Result evaluation showed that, generalizing LMA to a full body, lead to similar estimates, in Laban space, for the different persons in the same action categories, which propagated to the identification model, causing a high rate of misclassified identities. To overcome this issue, we propose a modified version of their identification model, adding an extra variable to the signature model, activity $\alpha \in \Lambda$, with the purpose of increasing discrimination. Let each $\gamma_i \in \Gamma$ be an independently and identically distributed motion signature feature. Consider the following joint distribution for the identification model:

$$P(\zeta, \alpha, \gamma_i, c_n)$$

where $\zeta$ represents a recognition variable, whose states correspond to different identities. We consider the following decomposition:

$$P(\alpha | c_n) P(c_n, \alpha | \gamma_i) P(\alpha, \gamma_i, c_n | \zeta) P(\zeta)$$

The prior distribution $P(\zeta)$ starts as an uniform distribution in the first iteration. The activity $\Lambda$ is estimated combining the different LMA variables. In our adapted approach, we learn a signature kernel model $P(\gamma_i | c_n, \alpha)$, which is now indexed to specific activities. This means that, for each activity $\alpha \in \Lambda$, a signature $\Gamma$ is computed for each different person, creating a kernel upon computation for all $\alpha \in \Lambda$. Using
5.4. Application on Person Recognition

Bayesian inference, the posterior density yields:

\[ P(\zeta|\alpha, \gamma_i, c_n) \propto P(\zeta) \prod_{q=1}^{n} P(c_q|\alpha) \prod_{q=1}^{n} P(\alpha, c_q|\gamma_i) \prod_{q=1}^{n} \prod_{p=1}^{i} P(c_n, \gamma_p, \alpha|\zeta) \]  (5.10)

The normalization factor is omitted for simplification purposes. The distributions \( P(\alpha, \gamma_i, c_n|\zeta) \), \( P(c_n|\alpha) \) and \( P(\alpha, c_n|\gamma_i) \) are the likelihood distributions, representing the identity model trained from real experimental data. The distributions \( P(c_n|\alpha) \) are Gaussian, while \( P(\alpha, c_n|\gamma_i) \) is a kernel of Gaussian distributions for \( \gamma_i \) generated from the probability values of \( c_n \) and indexed by \( \alpha \). The identity likelihood is a multi-variate stochastic matrix where signatures are associated to identities by means of activity and LMA state indexing.

5.4.2 Experimental Set-up and Results

Our identification experimental set-up encompasses 2 acknowledgeable datasets in motion analysis, KTH\(^\dagger\) and WZ\(^\dagger\). A LMA description is generated for each motion sequence, which is used to generate signatures and identity classification. We are interested in knowing the identification rate, i.e. how many times is a person correctly identified. Classification results are presented on a per frame and per sequence basis. The KTH dataset has 6 different actions (walking, jogging, running, boxing, hand waving, hand clapping) acquired at 25 fps frame rate and a 4 second average length. We have used 12 different actors. Identification results for the KTH dataset show an average per-frame accuracy over 90% and a perfect per sequence accuracy (Fig.5.6). Convergence is typically reached in 1 and 2 seconds time, considering the last instant

---

\( ^\dagger \) [Online] http://www.nada.kth.se/cvap/actions/


Figure 5.6: Results for identification on KTH database: Confusion matrix for per-frame classification accuracy; the last column shows per-sequence classification accuracy. (overall accuracy = 100%).
Figure 5.7: Results for identification on Weizmann database: Confusion matrix for per-frame classification accuracy; the last column shows per-sequence classification accuracy. (overall accuracy = 100%).

Figure 5.8: Average convergence time (seconds), considering a video sampled with fps = 25 and fps = 50 frames per second for KTH and WZ datasets respectively.

where a misclassified frame was observed. In fact, analysing per action results, we see the major confusion focus lies in similar actions, for which the classifier takes longer to converge. The WZ dataset is a collection of 90 video sequences of low resolution (180 x 144), recorded at 50 fps frame rate. It has 9 different people, performing 10 different actions (run, walk, skip, jumping jack, jumping, jump in the place, side, waving with one and two hands, bend). As in the previous dataset, per-sequence classification achieved 100% accuracy (Fig. 5.7). The per-frame ratio is still high, which is positive due to the number of similar actions. The convergence times are faster than in KTH, which is justified by an increased fps acquisition rate.
5.5 Conclusions and Discussion

In this chapter we have developed a model for a comprehensive characterization of motion sequences, which uses visual cues and is supported by an adequate descriptive grammar, based on the principles of LMA and notation. The analysis framework was integrated in a previously developed person recognition framework, for which the models were adapted to cope with the LMA generalization to the body as a whole. The generated LMA symbolic description was applied into activity identification and to develop motion signatures, which are combined into showing discriminating capability between different persons, using a Bayesian classifier. Results are promising and suggest further exploitation and model development. We intent to explore the feature behaviours with respect to different acquisition perspectives, by augmenting the model to multiple cameras simultaneously. It is our expectation to improve the signature model into relaxing the specific activity dependency. Furthermore, we aim to continue validating our identity estimation accuracy in more complex datasets, improving image segmentation and provide a working prototype.
Chapter 6

Cognitive Skills for Autonomous Action Learning and Synthesis

6.1 Introduction

So far we have presented methods for recognizing actions, with the ability to simultaneously infer different levels information from motion activities. While we have demonstrated that our basic symbolic descriptors can be generalizable, at the recognition stage we usually require a set of previously learned actions. In this chapter, we propose a cognitive framework, which gives an artificial system the ability to autonomously build its own action memory, by either refining existing knowledge or learning unknown actions, characterized by known properties. Moreover, we also present a methodology which allows the system to synthesize those learned actions, allowing them to be easily reproduced. With this aim, we propose a four stage cognitive process:

1. Sensing the environment;
2. Data interpretation;
3. Action learning and generalization;
4. Autonomous reproduction of learned actions;

We present a task oriented framework, in which upon scene interpretation, the system will probe the memory into learning and/or retrieving the appropriate action to solve a task in hand. Initial action memory is built based on "Learning by Demonstration" principles. Some experimental examples are presented to illustrate the propose
methods. The opportunity to collaborate with the Computer Science and Engineering Department, from University of Jaume-I, Castellón, Spain, lead to an experimental use case exploiting the developed models, in the context of Autonomous Underwater Vehicles for Intervention Missions, which is presented in Chapter 7.

6.1.1 Related Work

One trend in the current state of the art concerns learning based on geometric properties of manipulation movements to identify the different manipulation stages [FMLD12], or continuously learning constraints in a Gaussian Mixture Model approach [CGB07]. Kondo [KUO08], Bernardin [BOID05] and Kruger [KHB+10] use symbolic representations encoding hand-object contacts states, temporally represented in Hidden Markov Models (HMM) or Markov Decision Processes. Bekiroglu [BLJ+11] proposes an approach, using Support Vector Machine (SVM) and HMM to learn and assess robotic grasping stability. Lin [LRC12] applies GMM to learn required fingertip force and pose, to obtain a stable grasp during dexterous manipulation tasks. A different approach is presented in [LMM09], which applies inverse reinforcement learning techniques to infer the underlying task, which is being executed by the demonstrator. Another example comes from Jetchev [JT11] which adapts inverse optimal control techniques [RBZ06] to a single grasping task on a real robotic platform. Beyond the terrestrial applications, there are manipulation platforms working on space to fix satellites [Gui], where the robot is taught remotely by human operators using an immersive interface with sensorial feedback. Underwater scenarios have only recently been addressed, e.g. in [PGF+11] autonomous mobile manipulation in shallow water using a single robotic arm is presented. Recently, Carrera et al. [CAA+12], propose a learning solution for autonomous robot valve turning, using Extended Kalman Filtering and Fuzzy Logic to learn manipulation trajectories via kinaesthetic teaching.

6.1.2 Our Approach

Some action learning methods are supervised, which require a human expert to guide/teach the robot which actions are being performed. Others use unsupervised techniques, where the model is unknown and the system will explore the observable space state until it finds an adequate solution. In this chapter, we propose a hybrid approach, in the sense that, the system is required a set of initial knowledge, but at a given point, it will be capable of autonomously expand its own memory, using a set of cognitive
6.1. Introduction

Assume a robot has been taught, through a supervised learning process, to interpret a set of basic properties, which allow for a symbolic characterization of both the scene and its own actions, during task execution. The robot’s memory is categorical and follows a tree structure, in which actions are categorized according to combinations of those symbols. When performing a given task, the robot executes computational inference over those parameters, and uses such information to retrieve a specific memory location. By evaluating the expected action impact and the real outcome, the system will decide whether or not to memorize the just performed task, for solving future tasks.

In autonomous execution, the robot assesses and locates memory content to retrieve a satisfiable solution which fulfils the estimated task conditions. This process requires actions to be encoded, so as to allow for efficient storage, but also to permit retrieving and synthesizing them into low-level control parameters. Our approach is summarized in the block diagram of Figure 6.1. This cognitive process will be validated in an Underwater Intervention Mission scenario, where an Autonomous Underwater Vehicle (AUV) will autonomously assess, learn and execute a black box retrieval mission.

Figure 6.1: Cognitive Framework.
6.2 Cognitive Skills Model

Let an artificial system be given an unknown mission, for which it has to perform initial assessment and decide the appropriate course of action. During execution, the system will continuously perform environment and self analysis using a decision process based on Bayesian Inference. Acquired data is stored in a memory buffer until the system has made a decision whether or not to memorize it. At the end of each action phase, the success of the performed action will be evaluated, measuring the real performed action against the expected outcome. In case of correct execution, measured by a success quantitative threshold, the action is generalized and memorized in the correct memory location. In cases where a memory leaf contains existing knowledge, a fusion algorithm is proposed to merge the new action with memorized equivalents. The proposed memorization process is detailed in Figure 6.2. As experience accumulates, the system is expected to develop a consistent and accurate skill memory, adequate for solving increasingly complex tasks.

6.2.1 Action Generalization

Consider that our action data is composed of trajectories, that is, functions of time and/or distance to a target objective, which herein forth will be named as object,
considering that our validation scenario involves object manipulation. We propose to encode those actions using Gaussian Mixture Models.

The initial knowledge is acquired under the paradigm of “learning by demonstration”. An expert user teaches an artificial system into performing an action over an object, considering a fixed based manipulation scenario. This is done, to allow the user to focus on the manipulation task, rather than other factors, such as positioning itself with respect to the object.

Each function ($\theta_n$) is modelled independently, assuming it is a function of time ($t$), and constrained by a function of the distance ($d$), where these are the parameters of our GMM. Consider the example illustrated in Figure 6.3a, where we have a function of time, such that $\theta_2 = f(t)$. This function is encoded into GMM parameters, where we defined a number of $k$ Gaussian components and the parametrization is given as in equation (6.1), where $\Gamma = \{\theta_n, d, t\}$ represents the variable space.

$$p(\Gamma|\chi) = \sum_{i=1}^{k} \phi_i f(\Gamma|\mu_i, \Sigma_i) \quad (6.1)$$

Each function is represented by a set of weights and Gaussian Parameters given by $\chi$.

![Figure 6.3a](image1)

(a) Function for joint $\theta_2 = f(t)$.

![Figure 6.3b](image2)

(b) GMM representation of raw data function using $k = 3$.

Figure 6.3: Action encoding process using Gaussian Mixture Models for action: **Pick a Box from the Top, at Approach Phase.**
Chapter 6. Cognitive Skills for Action Learning and Synthesis

(φ₁, ⋯, φₖ, m₁, ⋯, mₖ, σ₁, ⋯, σₖ) for a number of k different Gaussian distributions. Mean and variance vectors are given by μ = (m₁, ⋯, mₖ) and Σ = (σ₁, ⋯, σₖ). Figure 6.3b presents the Gaussian Mixture Model of the function in Figure 6.3a, considering k = 3.

### 6.2.2 Action Synthesis

In this chapter, we are proposing an approach where, given a task to be executed, the system will retrieve the required learned Gaussian parameters to perform it. To decode these parameters into generalized configuration function sequences ˆω, we apply a Gaussian Mixture Regression (GMR) method. Let the joint data samples (Γ, ω), where Γ and ω are observations and target motion functions respectively, follow a Gaussian Mixture distribution as in equation (6.1). The parameters for the model are given by χ, and the joint distribution can be expressed as a sum of the products of the marginal density over Γ and the probability density function of ω conditioned on Γ:

\[
P(Γ, ω) = \sum_{i=1}^{k} φ_i P(ω|Γ, m_i, σ_i)P(Γ, μ_k, Σ_k).
\]

The marginal distribution is given by:

\[
P(Γ) = \sum_ω P(Γ, ω) = \sum_{i=1}^{k} φ_i P(Γ, μ_i, Σ_i).
\]

The regression function can be obtained from (6.2) and (6.3):

\[
P(ω|Γ) = \frac{\sum_{i=1}^{k} φ_i P(ω|Γ, m_i, σ_i)P(Γ, μ_i, Σ_i)}{\sum_{i=1}^{k} φ_i P(Γ, μ_i, Σ_i)}
\]

Where the mean and covariance of the conditional distribution P(ω|Γ) can be computed as:

\[
m_k = μ_k + Σ_k Σ_k^{-1}(Γ - μ_k)
\]

\[
σ_k^2 = Σ_k - Σ_k Σ_k^{-1}Σ_k
\]

The graphs in Figure 6.4 illustrate the covariance space after the regression (color blue) and the synthesized function ˆω (color red), obtained upon a Maximum-Likelihood Estimation (MLE) [CGB07].

\[
ˆω = MLE(m_k, σ_k^2)
\]

*To perform this process we have used a MatLab code developed by Calinon et al. [CGB07].
Each of these functions is computed for all $\theta_n$, which are naturally compliant with the system’s actuators, generating a synthesized adequate configuration along time and constrained by range. The data is then given to the system, in order to be executed by the kinematic control.

6.3 Categorical Skill Memory

Consider a memory whose organization is based on a B-Tree structure. In our work, we are considering a set symbolic variables: $X = \{X_1, X_2, \ldots, X_c\}$, where $X_c = \{x_c', x_c'', x_c''', \ldots\}$. The states that each of these variables can take, respectively $\{x_1, \ldots, x_c\}$, define the indexes of a B-Tree of order $c$ (see Figure 6.5). These are determined by scene and self analysis, for which Bayesian Inference is applied to a classification process using Dynamic Bayesian Networks. Let $F = \{f_1, \ldots, f_m\}$ be a feature vector, containing data perceived by the system’s sensors. Let the system self assessment information be represented by $\Theta = \{\theta_1, \ldots, \theta_n\} \in \mathbb{R}$. In our previous research stage, we developed a Bayesian-based analysis model, which based on a Maximum A Posteriori (MAP) method, allowed performing inference over each of the aforementioned symbolic variables, such that the Bayesian Program [BAMM12] leads to a generic Bayesian question $P(X_1, \ldots, X_c|\theta_1, \ldots, \theta_n, f_1, \ldots, f_m)$. The memory location is given by the estimate of the indexes upon applying the MAP method.

\[ \text{location index} = (\hat{X}_{1MAP}, \ldots, \hat{X}_{cMAP}) \]  

(6.7)

The MAP inference process is a point estimate in which retrieves the state exhibiting maximum probability upon weighing uncertainty of the evidences against the prior
probability. In equation (6.8) we present the MAP estimate for a variable $X_c \in X$.

$$\hat{X}_{c\text{MAP}} = \arg\max_{X_c} \prod_{i=1}^{m} P(\theta_i|X_c) \prod_{j=1}^{m} P(f_j|X_c) P(X_c)$$  \hspace{1cm} (6.8)

Our approach relies on the fact the system is able to robustly perform correct estimation of states for those indexing variables. Parameter information resulting from inference of the classification models, will allow for the system to find the index of the correct memory leaf. Upon a precise classification, we can state that our system can robustly and accurately estimate the correct memory leaf to memorize the currently performed action.

6.4 Incrementing Knowledge

When adding knowledge to the memory, the system will probe for existence of data or not. In the case it finds an empty leaf, it will simply add the new generalized data for the performed action parameters. Otherwise it will have to fuse existing knowledge with the one already stored. We propose two different approaches for knowledge fusion. One considers a Bag of Trajectories, while the other will fuse the generalized Gaussian parameters. In this latter case, the new knowledge will be given a weight, which is
computed by measuring the difference between the new generated trajectory and the one that is already generalized in memory.

6.4.1 Augmenting the Bag of Trajectories

This approach relies on a memory which stores raw trajectories rather than generalized knowledge, where the latter only requires space for a set of Gaussian parameters for a $k$ number of pre-defined components.

The disadvantage of Bag of Trajectories method, is that increasing knowledge will lead to increasing memory capacity. The method is indeed simplistic, which after deciding the memory leaf, it will add a new trajectory and a new set of generalized parameters will be computed. In addition, each new trajectory is continuously less important, as its weight to the Gaussian Mixture is $\frac{1}{m}$, where $m$ is the total number of memorized trajectories. After a long run of trials, the impact will be greatly reduced, leaving no room for short term adaptive capabilities.

6.4.2 Gaussian Parameters Fusion

When fusing Gaussian parameters, we have the advantage that we will not require additional memory and also, we have better means of controlling the weight of new trials. Let us have a set of existing memorized parameters $\chi \equiv \{m_1, \cdots, m_k, \sigma_1, \cdots, \sigma_k\}$, which represent the generalized action knowledge, for a given memory leaf. When a new generalized trial is proposed for a leaf, it is presented as $\tilde{\chi} \equiv \{\hat{m}_1, \cdots, \hat{m}_k, \hat{\sigma}_1, \cdots, \hat{\sigma}_k\}$.

![Image](image.png)

(a) Generalized representation with a Bag of 9 Trials  
(b) Generalized unknown trial (Green).

Figure 6.6: Generalized and Unknown action GMMs.
We are going to assume that our multivariate mixture models can be seen as a sum of multiple univariate distributions. Hence the new parameters resulting from the combination yield a mean:

$$m'_i = \frac{\alpha \mu_i + \beta \hat{\mu}_i}{\alpha + \beta}, \quad i = 1, \ldots, k$$

and a covariance given by:

$$\hat{\sigma}'_i = \frac{(\sigma_i + m_i)\alpha + (\hat{\sigma}_i + \hat{m}_i)\beta}{\alpha + \beta} - m'_i, \quad i = 1, \ldots, k$$

where $\alpha$ and $\beta$ are presented as weighing factors, which measure the contribution of the existing and new knowledge. In our approach, we consider $\beta = \alpha - 1$, which is a simplified version using normalized weights. This is done, in order to give the possibility to model the weights via probabilistic inference. In Figure 6.7, we present the resulting distributions upon fusing the new unknown trial (the green curves in Figure 6.6b) with the existing knowledge (the black curves in Figure 6.6a). As expected, the less impact the new trial has, the closer to the existing knowledge our new generalized action will be. In extreme cases, where $\alpha = 1$ and $\alpha = 0$, the resulting parameters will be the same as the existing ones, or vice-versa respectively.

### 6.4.2.1 Weighing New Knowledge

Let us address the computation of the weighing factors $\alpha$ and $\beta$. In this work, we hypothesize that the closer an new executed action is to the existing knowledge, the higher it should be weighted. However, the system should also consider the confidence with which the analysis model as determined the location of the memory leaf.

The first measure suggests we need a similarity measure between new and existing generalized knowledge, taking into consideration that samples may be shifted in time, so this measure should be time invariant. A popular measure that fulfils such requirements is Dynamic Time Warping. Let a new $\hat{\theta}_n = \hat{f}(t)$ and existing generalized $\theta_n = f(t)$ be two time-series to be compared:

$$c_p(\hat{f}(t), f(t)) := \sum_{l=1}^{L} c(\hat{f}(l), f(l))$$

The lower the total warping cost $c_p$ is, the similar both series are.
The second weighing measure, which measures the model confidence, can be given by the entropy value of the classified states. For a variable \( V = \{v_1, ..., v_z\} \), the entropy can be computed as follows.

\[
h = -\frac{1}{\log_b(z)} \sum_{i=1}^{z} p(v_i) \log_b p(v_i)
\]  

(6.12)

Where \( \log_b(z) \) corresponds to the maximum entropy value for \( z \) different states, therefore ensuring a normalized entropy value \( 0 < h < 1 \).

Having the two independent measures \( c_p \) and \( h \), we propose to compute the weighing factor \( \alpha \), such that its absolute value will reflect the following set of rules:

1. If both series are similar and the model shows high confidence, both new and existing knowledge should have shared weight;

2. If both series are similar, but the memory leaf is at a borderline location, then
the newly acquired knowledge, should have low impact;

3. In cases where the memory leaf is accurately identified, but series are not similar, the fusion should give more weight to existing knowledge to avoid memory degeneration. However, new knowledge should carry relative (lower) weight, allowing for short term adaptive behaviour;

4. The final case considers uncertain location and different series, for which the just performed action should be discarded, that is, its relative weight should be tendentiously zero.

Given this set of rules, we proposed to compute the weighing factor such that:

\[
\alpha = \gamma \left( 1 + \frac{(c_p + 2h)}{3} \right)
\]

(6.13)

where \(\gamma\) is a constant factor, which limits the minimum value for \(\alpha\). In our case, \(\gamma = 0.5\). We are giving the entropy more importance than the similarity measure to discriminate between rules 2 and 3. Reasoning behind this choice is that similar trajectories present less possibility of memory degeneration than that of wrongly located memory leafs.

6.5 Discussion

The final aim of the proposed cognitive skills model is to have an Autonomous Artificial System performing tasks autonomously, with minimum human intervention. To that purpose, autonomous execution will be used to incrementally update existing knowledge with new trials (Figure 6.8). While executing, the acquired information will be interpreted, and forwarded to a module which will decide whether the trial will be added to memory for future interventions, or not. This decision will be based on a comparison between expected and real mission outcomes. This continuous process, the
generalized action information is synthesized into low-level control primitives acting on the system itself.
Chapter 7

Case Study: Underwater Autonomous Manipulation

7.1 Introduction

This chapter discusses a case study where the previously presented cognitive skills are applied in the context of autonomous underwater intervention missions. Current research in the underwater robotics intends to increase autonomy for all kinds of robotic intervention operations requiring physical interaction. Despite the fact that autonomous robotic intervention on land remains in development and with some valuable achievements, the current state-of-the-art in underwater intervention missions is currently in a very primitive stage where the majority of the systems are tele-operated by an expert user. This case study intents to address this challenge, through research that stills under development, within the context of a project, funded by the Spanish Ministry, titled GRASPER. GRASPER (under the responsibility of University of Jaume-I, UJI, and addressing the problem of the “Autonomous Manipulation”) represents only a sub-project inside a Spanish Coordinated Project, entitled: TRITON∗ “Multisensory Based Underwater Intervention through Cooperative Marine Robots”, which includes two other sub-projects: COMAROB (“Cooperative Robotics”, under the responsibility of University of Girona, UdG), and VISUAL2 (“Multisensorial Perception”, under the responsibility of University of Balearic Islands, UIB). In summary,

TRITON is a marine robotics research project focused on the development of intervention technologies really close to the real needs of the final user and, as such, it can facilitate the potential technological transfer of its results. The specific objectives for GRASPER are the following:

1. To develop the user interface and simulation capabilities needed for TRITON.
2. To generate all the mechatronics and sensor improvements to succeed in the autonomous manipulation requirements.
3. To develop new planning and control strategies, making use of range and visual information, finally leading to visual free floating manipulation.

This chapter highlights the potential benefits of including a new approach based on the “learning by demonstration” paradigm, in order to increase autonomy in the required grasping and manipulation skills. Because initially the experimental validation will be carried out in virtual reality (i.e. by using the 3D simulator UWSim [PPFS12] described below), some contributions are expected in the aforementioned objectives (1) and (2).

### 7.1.1 Initial Strategy and Roadmap

The activities developed in this research activity follow a methodology where the core techniques can be designed, developed and prototyped with support of a simulator named UWSim [PPFS12]. The research results generated by this activity are after, tested on real scenarios with different levels of complexity. The Figure 7.1 provides a graphical perspective of this strategy.

This methodology and the modular computational architecture is based on the Robot Operating System (ROS) and provides the support for prototyping a solution based on a simulator that can be used to target the real robot, in different real scenarios. The architecture allows us to switch from the simulated environment to a real scenario at any moment and test the prototyped system (manipulation, new algorithms, learning, etc.). The real test scenarios include different physical complexities with increasing degree of realism and hard conditions, when compared with open sea conditions:

- Testbed 1: Water Tank (described below) (UJI, Castellón, Spain)
- Testbed 2: CIRS pool at Girona (UdG, Girona, Spain)
- Testbed 3: Roses Harbour (Roses, Spain)
7.1. Introduction

Simulation
Real Scenario 1
Increasing
Complexity
Real Scenario 2
Real Scenario 3

Figure 7.1: Development strategy: the core techniques can be designed, developed and prototyped inside the UWSim simulator. Then, the research results generated by this activity are tested on real scenarios with increasing scales of complexity.

For each development step or research outcome it is possible to introduce more complex scenarios by simulating them on UWSim system and test the results in different test-beds that convey real hardware in real environments with increasing number of uncontrolled variables (disturbances, visibility, noise, etc.).

7.1.2 Related Work

In the field of the underwater intervention it is worth mentioning previous projects like SAUVIM [MCY09], intended for deep interventions, which demonstrated the autonomous recovery of seafloor objects by using a very bulky and expensive system; and TRIDENT* [SRO+12], that demonstrated the first multipurpose object search and recovery strategy in 2012, able to operate in shallow waters. Nowadays, two ongoing projects are running in the underwater intervention context funded by European Commission: MORPH† and PANDORA‡. It is also noticeable, that the ongoing TRITON

†Marine Robotic System of Self-Organizing, Logically Linked Physical Nodes (FP7-MORPH), available: [http://morph-project.eu/](http://morph-project.eu/)
‡Persistent Autonomy through LearNing, aDaptation, Observation and Re-plAnning (FP7-PANDORA), available: [http://persistentautonomy.com/](http://persistentautonomy.com/)
project is an extension of the previous Spanish founded project RAUVI\footnote{Reconfigurable Autonomous Underwater Vehicle for Intervention (RAUVI), available: \url{http://www.irs.uji.es/rauvi/}}\cite{SPR+10}. RAUVI was the origin of TRIDENT, demonstrating in 2011 a successful approach for the search and recovery problem but in a more limited manner.

### 7.1.3 Our previous approach

With the aim of increasing the autonomy levels of the underwater manipulation systems, we have recently been working in a multi-sensory based manipulation approach\footnote{Underwater semi-autonomous grasping experiments using laser 3D reconstruction can be seen on-line: Experiment 1: \url{http://youtu.be/VOLNBWfeoLo}, Experiment 2: \url{http://youtu.be/c62FTTyGxsQ}, Experiment 3: \url{http://youtu.be/42ZklVwNaqc}}. This approach allows the grasp of different known-a-priori objects in a water tank, but still requires the user intervention in order to specify the grasp. Some important pieces of this approach are now described:

#### 7.1.3.1 UWSim: the underwater simulator:

UWSim is a software tool for visualization and simulation of underwater robotic missions \cite{PPFS12}. The software is able to visualize an underwater virtual scenario that can be configured using standard modeling software. Controllable underwater vehicles, surface vessels and robotic manipulators, as well as simulated sensors, can be added to the scene and accessed externally through network interfaces. UWSim do the interface with external control programs through the \textit{Robot Operating System (ROS)}. UWSim has been successfully used for simulating the logics of underwater intervention missions and for reproducing real missions from the captured logs \cite{PPFS12}. UWSim is currently used in different ongoing projects funded by European Commission (MORPH and PANDORA) in order to perform HIL (Hardware in the Loop) experiments and to reproduce real missions from the captured logs.

#### 7.1.3.2 3D Reconstruction of the Scene:

The aforementioned approach requires the reconstruction of the geometry of the objects laying on the floor. To achieve this, a scan of the scene is performed using a structured laser beam attached to the forearm of the manipulator. The scan is done by moving the elbow joint of the manipulator at a constant velocity. At the same time, a digital video
camera is used to capture the scene with the laser beam projected on the object. A visual processing algorithm runs in parallel: the laser peak detector, which is in charge of segmenting the laser stripe from the rest of the image and computing the 3D points \cite{PFS12}. With these points, a 3D point cloud of the scene is built and represented on the simulator.

### 7.1.4 Our Approach

In this chapter, we will address three different problems that contribute for the development of techniques which will find its application in the aforementioned projects and specified scenarios: learning and generalizing actions by human demonstration; develop a model for analysing scene and actions being performed by the manipulator; and increment the knowledge database via memorization of autonomously performed actions. A human demonstrator will use the UWSim platform to teach the manipulator how to perform a specific action. Such action will be recorded and its parameters generalized into action knowledge. These action parameters are moved into an action memory, which may have to perform data fusion in case memory leafs already contain previously acquired knowledge. Memory index is computed from the characteristics inferred from scene and self analysis, using a Bayesian-based classifier. The manipulator should then exhibit a high degree of autonomy, where it should be able to analyse, memorize and execute different actions, with minimum human intervention.

This chapter presents two main result branches: the system’s analysis capability and the results of memory fusion of existing and newly performed action skills.

### 7.1.5 Definitions

Assume there is an object \( O \) represented by a set of characteristics \( C_O \), located in a 3 Dimensional underwater space \( U \in \mathbb{R}^3 \). Consider a \( n \) DoF manipulator \( M \), operating in \( U \). The challenge is to give \( M \) a set of skills \( S \), such that \( M \) is able to reach, grab and manipulate \( O \) into reaching a user specified goal \( G \). At a first stage, this knowledge \( S \) is taught by a human expert. Posteriorly, the manipulator exploits the skill space \( S \), in order to operate autonomously into solving \( D(G, O) \). From this scenario (Figure 7.2), we are able to identify the following main problems:

1. The development of a realistic simulation environment, that a human operator can control and, simultaneously, from which it is able to get realistic feedback
while teaching, via “tele-operation”, a virtual representation of $M$.

2. Find a suitable, probabilistic knowledge representation, which accurately models the relation between a set of manipulator sensed information $I$ and a set of skills $S$, such that $M$ can interpret scene information, identify objects of interest and decide the best course of action $D$ into satisfying $G$. The solution $D$, should be updated every time new information is available, so to be able to cope with dynamic and difficult underwater operation conditions.

3. Given a solution for $G$, project $D$ into a set of motor primitives, allowing the mechanical system $M$ to operate the different steps of its intervention mission.

4. Define a metric to evaluate the success of each intervention, so the system has the capability to decide whether or not the new proposed solution for $G$ should be incrementally added to existent knowledge $S$.

The proposed solution to this problem can be easily stated: a autonomous manipulator $M$ should be able to decide the best solution $D$ for a given user specified goal $G$, based on the information $I$ its sensors are able to acquire from the environment $U$. Such information is, at its most basic forms, identity and pose of objects $O$, obstructions and its relative End-Effector $M$ pose towards a specified goal $G$. The solution and integration of these problems are expected to provide an intelligent system, capable of autonomously perform underwater tasks, with minimum human intervention, while being able to constantly adapt to the difficult underwater conditions.

7.1.5.1 Manipulation Skills: Phases, Information and Tasks

In our manipulation scenario we parametrize the execution in 4 different stages:
1. **Initial**: The initial stage is a stage where the robot acquires initial information from the scene, and starts the first iterations to identify scene properties and find initial solution $D$ for the proposed user defined task $T$.

2. **Approach**: in this stage, the system will refine its assessment of the environment conditions and gather extra scene information, adjusting its behavior during the reach to grasp trajectory.

3. **Reach/Contact**: once in the neighborhood of the target object, the manipulator needs to decide the best pose and force parameters to enter in contact with the object.

4. **Contact/Manipulation**: at this stage, the manipulator needs to operate the gripper in order to move the object from an initial to a final position, i.e. a second goal $G$, which is defined by a user specified goal OR automatically assessed from the available sensed information.

We propose a log-spherical intermediate defined key points, at which the manipulation should verify its own attitude towards intermediate and final goals. An example is show in Figure 7.3. We define **Attitude** as the End-Effect pose, velocity and gripper state, with respect to a specific goal. This attitude should be inferred based on information acquired from the laser scanner and vision system.

**Solutions** are addressed from different system perspectives. At each of these stages a supervised learning process is applied, using a tele-operated realistic simulator envi-
Figure 7.4: On the left we have a user operating the UWSim during a training sessions. On the right we have a sample frame to exemplify the visualization screen.

...
The UWSim [PPFS12] is a software tool for visualization and simulation of underwater robotic missions. The software is able to visualize an underwater virtual scenario that can be configured using standard modeling software, and do the interface with external control programs through the Robot Operating System (ROS). UWSim is currently used in different ongoing projects funded by European Commission (i.e. MORPH [FP7] and PANDORA) in order to perform HIL (Hardware in the Loop) experiments and to reproduce real missions from the captured logs.

The simulated robotic arm is a virtual representation of the real arm considered for the validation scenario (CSIP Light-weight ARM5E). It has 5 D.O.F. and can be equipped with different kind of grippers, which can also be sensorized to provide contact information, being this information useful when grasping an object. An example of a successful reactive tactile sensor test recently performed in laboratory conditions (water tank) can be seen on-line [Sen].

The low-level control architecture, including the arm kinematics, was implemented in C++ and makes use of ROS for inter-module communications. The kinematic module accepts either Cartesian or joint information (i.e. pose, velocities). The ARM5Control module uses joint velocities in order to compute motor RPM [FPG+12].

Implementation of the HRI simulator, ensuring it will acquired the necessary data for learning the manipulation skills. The data acquired from the user controlling the simulator will be used to develop a filter for assessing what is considered a good trial or not, deciding which trials can be included in the learning.

UWSim do the interface with external control programs through the Robot Operating System (ROS). This architecture provides message-passing and communication between nodes in a transparent manner, thus allowing both local and remote localization of executing nodes (the simulator itself, the learning and database modules, the user interface, etc.). As a consequence of this, we are able to run the whole system in a single computer but also in a distributed system, allowing thus remote learning.

7.3 Environment and Action Analysis Model

The system continuously performs inference about the scene and its own state, based on information acquired from built-in sensing capabilities. A Dynamic Bayesian Network is proposed to perform estimation about the various states, in an interpretation process, developed using Bayesian Programming [BAMM12, CDB10, FLB+13]. The relevant
variables are defined in the first step of Bayesian Programming, such that:

- $A \in \{\text{pick-up, push}\}$ is a random variable denoting the different actions our manipulator can perform.

- $G \in \{\text{top, lateral, front}\}$ is a random variable denoting the approaching region of the end-effector to the object of interest.

- $P \in \{\text{approach, reach, manipulation}\}$ is a random variable which identifies the current phase of a given action. We divide an action into three different phases: the approach phase in which the manipulator identifies the object of interest and starts moving in its direction; in the reach-to-contact the end-effector is required to take the grasp configuration needed to perform the action; the manipulation stage happens when the end-effector is in contact with the object to perform a specific goal.

- $O \in \{\text{box, stone, vessel}\}$ is a random variable defining the known object classes.

- $\Theta \equiv [\theta_1, \cdots, \theta_n] \in [-\pi, \pi]$ is a random variable representing the angle in radians of a given joint $n$.

- $F \equiv (f_1, \cdots, f_b) \in \mathbb{R}^b$ represents the sensed information as a vector of processed laser range measures.

- $d \in \mathbb{R}$ is a random variable measuring the distance from the end-effector to the target object.

To estimate the joint state of an action and its corresponding characteristics, we use Bayesian Inference on the decomposition equation of Figure 7.5. The most likely candidate object class is estimated upon weighing the uncertainty of each evidence $\in F$. Each different action phase $P$ depends on the existing knowledge about the manipulator configuration $\Theta$ together with the relative distance to the target object $d$. The geometric configuration is used as evidence for inference over the Grasp Type state. These dependencies are reflected as conditional probabilities, elements of the decomposition of the joint distribution, which are also reflected in the Direct Acyclic Graph (DAG) in Figure 7.6. The DAG is divided into different abstraction levels for easier comprehension. There is the action space, the characterization space encompassing the phase and the grasp type, the object space with information about the target object and the feature space, which holds the observable evidence.
Likelihood distributions, which are function of random variables $\in \mathbb{R}$, are formulated upon Gaussian distributions, as they are expected to be normally distributed. The density functions that depend on discrete variables, as is the case of the variables whose space state is symbolic, do not follow any particular known parametric form. They represent statistical information of the different symbols being observed for a given state, and are encoded using stochastic matrices.

Once the model is fully specified we can now inquire it for information. To obtain an estimate of the most probable action state given observable evidence, we apply the Maximum A Posteriori (MAP) method, in which inference over the action variable is given as in equation (7.1).

$$
\hat{A}_{MAP} = \arg \max_{A} P(\Theta|G)P(O|G)P(d|P)
$$

$$+ P(\Theta|P)P(F|O)P(P|A)P(O,G|A)P(A) \tag{7.1}
$$

The MAP method is a point estimate, which will return the most probable state from each of the symbolic variables. These states, which maximize the inferred distribution for each variable, are used by the system to locate and probe a specific memory leaf,

<table>
<thead>
<tr>
<th>Variables:</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A, G, P, O$: scene and state symbolic information;</td>
</tr>
<tr>
<td>$\Theta \in [-\pi, \pi]$: real values of each joint $n$;</td>
</tr>
<tr>
<td>$F \in \mathbb{R}^b$: laser range finder characteristic vector;</td>
</tr>
<tr>
<td>$d \in \mathbb{R}$: distance from the end-effector to the object;</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Decomposition:</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P(A,G,P,\Theta,d,O,F) =$</td>
</tr>
<tr>
<td>$P(\Theta</td>
</tr>
<tr>
<td>$P(F</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Formulation:</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P(A) = \text{Histogram}$</td>
</tr>
<tr>
<td>$P(\Theta</td>
</tr>
<tr>
<td>$P(\Theta</td>
</tr>
<tr>
<td>$P(P</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Identification:</th>
</tr>
</thead>
<tbody>
<tr>
<td>parameters learned from training data $F, \Theta, D$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Questions:</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P(A</td>
</tr>
</tbody>
</table>

Figure 7.5: Bayesian Program description: (1) enumerates the relevant variables; (2) joint distribution decomposition; (3) formulation of the conditional distributions in parametric forms; (4) Identification stage where parameters of the Gaussian distributions are estimated from experimental data.
Figure 7.6: Directed Acyclic Graph of the proposed Dynamic Bayesian Network. Nodes represent variables and directed arcs represent variable dependencies.

as described in previous chapter.

7.4 Experimental Results

7.4.1 Experimental Setup using the UWSim

In this section, the scene setup inside the simulation environment and the developed modules to allow user teleoperation and data acquisition are described. This information is used to continuously feed the proposed memory structure to allow autonomous execution using the learned skills. To perform the autonomous execution experiments, a scenario with a box lying on the floor (a mockup of a flight recorder, or black box, from a crashed aircraft) has been defined and loaded into the simulator (see Figure 7.7a). The goal is for the I-AUV to grasp the object autonomously after a learning process, from the user teleoperation. The human-robot interaction involves: (1) the use of a gamepad for teleoperation and feedback, (2) the complete 3D visual information of the scene provided by the simulator, and also (3) several sensor information, such as the distance to the target object, collision detection, target reachable, etc, provided by a specifically developed module (User Monitor module) to display this information, connected to the simulator through a Robot Operating System (ROS) interface (see Figure 7.7b). The module, apart from the provided visual information, also sends vibro-tactile information to the gamepad. This is done to give the user a sense of contact between the arm and/or the end-effector and the target object while in the teleoperation stage. The first step in the learning process conveys data acquisition from several trials, upon user demonstration by using the simulator. The acquisition was done for different ac-
7.4. Experimental Results

(a) UWSim Scenario

(b) UWSim user feedback.

Figure 7.7: User Monitor module connected to UWSim simulator to provide feedback to the user during the learning stage and the considered scenario: the recovery of a black box laying on the seafloor.

In this section we present the experiments which demonstrate the interpretation capabilities of our system. Classification for Action, Object and Grasp types are extremely positive, showing highly precise results. This fact might also be a reflex of a reduced number of classes, however, the target Underwater Manipulation scenarios are not expected to have high dimensional variable spaces, for which we can confidently assume that our approach for scene interpretation to be adequate. With respect to Action Phases, we can see that the results in Table 7.1 (d) suggest a delay when detecting the Reach-to-Contact. After a thorough step-by-step analysis, we found this confusion to come from the distance $d$ thresholds in the model training. In fact, the Reach-to-Contact phase is defined by short ranges and time periods, generating Gaussian Distributions of low variance (see Figure 7.8). Hence, when in the presence of noise, it easily diverges to the nearest phase class, more specifically, Approach. However, the global precision of the analysis model, per symbolic variable presents promising values...
Table 7.1: Confusion Table of Symbolic Classifications. Acronyms: (A) Approach; (R) Reach-to-Contact; (M) Manipulation; (T) Top; (L) Lateral; (F) Front; (V) Vessel; (B) Box; (S) Stone; (Pu) Pick-Up; (P) Push.

(a) Action Confusion Table.

<table>
<thead>
<tr>
<th></th>
<th>Pu</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pu</td>
<td>1.00</td>
<td>0.00</td>
</tr>
<tr>
<td>P</td>
<td>0.10</td>
<td>0.90</td>
</tr>
</tbody>
</table>

(b) Object Confusion Table.

<table>
<thead>
<tr>
<th></th>
<th>V</th>
<th>B</th>
<th>S</th>
</tr>
</thead>
<tbody>
<tr>
<td>V</td>
<td>0.95</td>
<td>0.00</td>
<td>0.05</td>
</tr>
<tr>
<td>B</td>
<td>0.00</td>
<td>1.00</td>
<td>0.00</td>
</tr>
<tr>
<td>S</td>
<td>0.01</td>
<td>0.00</td>
<td>0.99</td>
</tr>
</tbody>
</table>

(c) Grasp Type Confusion Table.

<table>
<thead>
<tr>
<th></th>
<th>T</th>
<th>L</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>1.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>L</td>
<td>0.00</td>
<td>0.95</td>
<td>0.05</td>
</tr>
<tr>
<td>F</td>
<td>0.10</td>
<td>0.00</td>
<td>0.90</td>
</tr>
</tbody>
</table>

(d) Phase Confusion Table.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>R</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.93</td>
<td>0.04</td>
<td>0.03</td>
</tr>
<tr>
<td>R</td>
<td>0.32</td>
<td>0.55</td>
<td>0.13</td>
</tr>
<tr>
<td>M</td>
<td>0.00</td>
<td>0.03</td>
<td>0.96</td>
</tr>
</tbody>
</table>

(see Table 7.2), indicating that the proposed analysis models can be tested, within the scope of the ongoing project, into a more complex experimental phase, the underwater tank.

Table 7.2: Global precision per symbolic variable.

<table>
<thead>
<tr>
<th>Symbolic Variables</th>
<th>Precision (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>action</td>
<td>95.00</td>
</tr>
<tr>
<td>object</td>
<td>99.12</td>
</tr>
<tr>
<td>grasp</td>
<td>96.66</td>
</tr>
<tr>
<td>phase</td>
<td>86.50</td>
</tr>
</tbody>
</table>

Figure 7.8: Learned Gaussian Distributions for \( P(\theta_1|P=p) \), with \( p = \{\text{approach, reach, manipulation}\} \).
7.4.3 Memory Fusion

For this set of experiments we will consider a Cross Validation procedure. Given a set of \( m \) performances, we will learn \( m - 1 \) and store the generalized knowledge into memory, and then fuse the generalized representation of the remaining performance. We will then compare the \( m \)-fold fusion process to the Bag of Trajectories approach, where all \( m \) are considered. Similarly to Figure 6.7, we essay different values for \( \alpha \).

Results from Figure 7.9 show that when \( \alpha \) becomes lower, the impact of the just executed functions increases. This is an expected behaviour for when we want existing memory knowledge to adapt to new conditions in the short-term. In cases where the memory is already found adequate, the fused knowledge will be only slightly modified. Results present the generated fusion time series and compared with the one generated from all \( m \) data available. In the Table 7.9e we present these results under the form of Mean Squared Error (MSE), where we measure MSE of each fused function against the one generated with Bag of Trajectories.

\[
MSE = \frac{1}{q} \sum_{i=1}^{q} (\hat{f}(i) - f(i))^2
\] (7.2)

The above equation, measures the MSE for two different learned functions with \( q \) number of samples.

The MSE is applied to this specific case, because are interested in maintaining the time dependence, as it is critical to evaluate the generalized fused knowledge. Values, as expected from Figure 7.9, show that MSE is inversely proportional to \( \alpha \), an expected behaviour. In fact, most of the MSE minimum values are shown for the highest value of \( \alpha \), with a reduced number of exceptions. Results show, that the proposed memory fusion has an adaptive behaviour, which is dependent on both memory indexing accuracy and how much impact should the newly performed action have in the existing memory.

7.4.4 Autonomous Execution Experiments

To perform the autonomous execution experiments, a scenario with a box lying on the floor (a mockup of a black box from an aircraft) has been defined and loaded into the simulator. The goal is for the AUV to grasp the object autonomously after a learning process, from the user teleoperation. The human-robot interaction involves: (1) the use of a gamepad, (2) a complete 3D visual information display, and also (3) sensor
Figure 7.9: Fusion results (colored lines) in a Cross Validation approach, fusing $m - 1$ learned trials with the one that has not been learned. In black is the generalized results for the $m$ trials. The MSE Table presents a clearer measure of how different are synthesize trajectories for both approaches.

<table>
<thead>
<tr>
<th>$\sqrt{\text{MSE}}$ in ($\times 10^{-1}$)</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha = 0.5$</td>
<td>4.753</td>
<td>5.026</td>
<td>2.349</td>
<td>2.884</td>
<td>4.438</td>
<td>1.595</td>
<td>0.797</td>
<td>4.131</td>
<td>3.097</td>
<td>1.317</td>
</tr>
<tr>
<td>$\alpha = 0.6$</td>
<td>5.031</td>
<td>5.011</td>
<td>1.869</td>
<td>3.477</td>
<td>4.229</td>
<td><strong>1.314</strong></td>
<td>1.656</td>
<td>4.157</td>
<td>3.400</td>
<td>1.218</td>
</tr>
<tr>
<td>$\alpha = 0.8$</td>
<td>3.426</td>
<td>4.739</td>
<td><strong>0.188</strong></td>
<td>3.970</td>
<td><strong>4.480</strong></td>
<td>1.625</td>
<td>0.242</td>
<td>3.981</td>
<td>4.007</td>
<td>0.691</td>
</tr>
<tr>
<td>$\alpha = 0.9$</td>
<td><strong>2.249</strong></td>
<td><strong>2.981</strong></td>
<td>1.137</td>
<td><strong>2.354</strong></td>
<td>4.835</td>
<td>1.393</td>
<td><strong>0.141</strong></td>
<td><strong>2.076</strong></td>
<td><strong>3.821</strong></td>
<td><strong>0.672</strong></td>
</tr>
</tbody>
</table>
information, such as the distance to the target object and collision detection. A User Monitor module was specifically developed to display this information, connecting to the simulator via ROS interface (see Figure 7.7b). The module also sends vibro-tactile information to the gamepad. This is done to give the user a sense of contact between the arm and/or the end-effector and the target object. The first step in the learning process conveys data acquisition from several trials, upon user demonstration.

The second step involves associating data to task properties and store the generalized representation into the action memory. On execution, the memory will be probed whenever an action phase change occurs. The adequate motion sequence functions (one for each joint) are synthesized and sent to the specifically developed kinematics module (low-level controller). At this stage, the proper input to the simulator is generated considering a specified rate, resolution and velocity ranges. Discontinuities are automatically solved using smooth interpolation. This way, the training and learning loop is closed and the simulator is able to display the automatically generated grasp execution after the user training stage. Video examples for both the training and execution synthesis are available on-line at the following website: 

https://sites.google.com/a/uji.es/learning. This website is intended to be a growing dataset available to the scientific community with data acquired for different actions, approaches and target objects. Right now it is possible to find the acquired data for the described experiments, the developed software modules and the documentation for properly downloading and installing them.

Figure 7.10: Global system block diagram, encompassing acquisition, interpretation, memory and execution stages.
Chapter 7. Case Study: Underwater Autonomous Manipulation

7.5 Conclusions

After very successful research achievements through previous projects, like TRIDENT or RAUVI, following a semi-autonomous strategy, we are trying to increase now the autonomy levels, under GRASPER project context, by means of learning. This new approach is supported by the ongoing cooperation between UJI (Spain) and ISR (Portugal).

With the aim of increasing the autonomy levels for Underwater Intervention Missions, we have developed a set of computational cognitive skills that allow the system to automatically grasp an object after a learning process. The developed scheme allows the robot to learn by demonstration, memorize, decide and access autonomously the suitable solution to solve a task. The implemented solution (to be publicly available) works as independent ROS nodes that can connect both to the simulator and the real environments, as they share the same interface. The proposed models have been validated in a realistic underwater simulation environment, paving the way for testing them in more complex environments (as previously described in Figure 7.1).

We have also developed a memory, which is organized categorically as a B-Tree, where leaves contain generalized actions for solving a determined task, with generalized symbolic properties. The system is capable of interpreting the environment and its own state and use that information into successfully locate a memory cell for saving the just executed action. In cases where information already exists, a fusion method was developed, which used similarity and memory location (un)certainty, into according to pre-defined rules, defines the weight of the newly acquired knowledge. Results show the robot is capable of integrating new knowledge without degenerative effects on memory, while being able to adapt to short-term conditions.
Chapter 8

Conclusions

In this thesis, we presented methods and developed concepts that enable artificial systems to robustly increase their cognitive capabilities, which is critical for the success of social and/or intelligent robotic systems in the future. Most of relevant applications require those robots to be able to infer different types of information from observing human actions, constantly adapt to new situations and make autonomous decisions while simultaneously dealing with large amounts of noise and uncertainty. Therefore, this thesis had the main purpose of developing novel approaches that allow artificial systems to increase its cognitive capabilities to successfully interpret, learn and act so as to fulfil their tasks.

In mobile social robots or intelligent monitoring systems, an artificial system needs to be able to recognize different types of activities as well as perform analysis over different aspects of human motion. Therefore, we investigated methods which give a system the capability to infer multiple levels of such information. Towards such intent, we have exploited a motion descriptive language, Laban Movement Analysis, which can provide a comprehensive description of motion using basic units. This lexicon has supported a highly flexible and scalable Bayesian-based model. Our models use body part 3-D trajectories as input data. In our experiments, we have demonstrated that we can achieve state-of-the-art classification accuracy, while simultaneously providing different types of symbolic descriptors. Moreover, Laban semantics demonstrated to be generalizable, where its symbolic qualities show to be repeatable for similar actions, performed by different persons, even when those trials have not been used to train the model. In addition, we have applied our framework to 2-D data successfully.

Our approach to motion analysis has been extended towards identifying persons by the way they move. We have developed methods which encode and retain the
expressive characteristics of each person’s motion, using Laban symbolic descriptors. These signatures have been successfully integrated with different Person Recognition methods, also developed in this thesis. Experimental results show that our methods have can in fact identify different persons with high precisions, by observing random activities. Our methods shown to be action invariant, which has been demonstrated in a batch of Leave-One-Out Cross-Validation procedures. Moreover, we have extended our approach to tests using only gait-based actions and also to 2-D data, for which our methods also exhibited similar success.

A relevant property of intelligent video-surveillance systems, apart from its capability to recognize actions, is the ability to recognize different persons. However, tracking specific body parts might not be a trivial issue in real scenarios. Therefore, we have proposed a vision-based motion analysis model, which generalized Laban Movement Analysis description to visual cues. Moreover, the previously developed person recognition methods had to be adapted, so as to cope with such generalization. Results indicate the high potential in our approach, where the proposed methods have shown to be able to simultaneously recognize actions and identify the performer in videos containing motion sequences.

A critical issue in autonomous robots/systems, is their ability to learn by themselves our to analysis or react in a given situation. To that end, we proposed a set of cognitive skills, which allow the robot to infer information from the environment and itself, using such information towards building its own action memory. We have applied Gaussian Mixture Models to generalize actions, so as to have an efficient and robust memory representation and storage. At execution stage, the robot will probe the memory and uses Gaussian Mixture Regression into retrieving a set of functions which can be interpreted by a low-level control module for autonomous acting. We have perform experimental learning and execution tests in a scenario of Autonomous Underwater Intervention, using a realistic environment able to simulate underwater conditions and the Autonomous Underwater Vehicle.

All techniques present in this work have been thoroughly tested. Experimental set-ups have been carried with 3-D data and extended to 2-D. We have also recorded and made publicly available a motion database, which encompasses 11 different activities acquired with different sensor technologies. Our techniques have been demonstrated to give artificial systems a set of cognitive skills, allowing them to operate under uncertainty and noise, while robustly been able to fulfil their tasks. They also support our claim that Laban Movement Analysis has the properties, which make it a serious candidate to a grounding motion symbolic descriptor for human to robot communica-
This thesis’ contributions have been presented as solutions to several problems in the context of action recognition and analysis as well as cognitive skills in autonomous systems. Some of these techniques have been integrated in the Robot Operating System (ROS) and a comprehensive motion database has been released to the community in this research area. The proposed methods allow a robot to autonomously solve the following challenges:

1. How can a robot infer different types of information from human motion, past learning a limited set of actions/properties?

2. How can a system recognize a person by its motion, without requiring a cooperative behaviour, observed in most biometric approaches?

3. How can a robot autonomously learn new actions or adapt its knowledge to new unknown conditions of the environment?

4. How can a system autonomously search a solution to a task and reproduce it in a new situation?

The answers to this questions have been addressed in this thesis, enabling artificial systems to have a more robust and comprehensive perception of the environment and provide appropriate, adaptive solutions. We expect that the proposed methods and solutions can be relevant to increase robots cognitive intelligence to assist us in our lives in a near future.

8.1 Future Work

Even considering that the results presented in this thesis are promising, it is possible to identify open research question that may be investigated in the future. As an example, we think expanding the analysis beyond expressive motion properties might increase robustness even more, up to a nearly perfect action recognition classifier. Moreover, it would be of interest to study explicit interactions in order to improve the robot’s ability to react more accordingly. A first step could be teaching the robot how autonomously interpret two person interactions by passive observation, and then replace one of those actors by the robot itself. Also it could make sense to integrate a set of social rules, which would increase its ability to categorize actions. Moreover, developing a method
allowing the robot to measure the outcome of its own actions would greatly increase the cognitive capability to autonomously learn unknown actions, environments and reactions.

In our work we address hierarchical analysis of human motion. It would be of interest if the robot could perform unsupervised learning of new unknown actions. That is, if a new observed action exhibited a high uncertainty in the searchable classification space, the robot would autonomously learn it as being a new unknown action. Posteriorly, it could replicate it for a human expert, which would teach the robot the correct symbol for characterizing it. The capability to deal with unknown actions is currently not solved in our approach.

We have proposed an intelligent video-surveillance system, which can recognize both action and performing actor, given that its motion profile is known. Adapting Laban symbolic description to the whole body proved to withdraw some of its discriminant capabilities. We suggest exploiting different visual cues, such as optical flow, silhouette segmentation, skeleton reconstruction, with the purpose of characterizing different parts of the body and thus increasing the vision-based Laban signature discriminant capabilities. Also, it would be interesting to apply the concepts of our cognitive skills, to give the system the ability to learn unknown person and/or incrementally refine signature profiles for existing identities.

With respect to action memory, it would be of interesting to evaluate how much it would grow to a comprehensive knowledge database and how it could be used to disseminate information to other robots, in a paradigm of “cloud knowledge”. Also, given a set of basic actions, how would the memory be incrementally filled with new actions, or solve tasks under unknown environment conditions. Moreover, in case of a highly complex searchable memory, how could we use context information to narrow the search space to a specific memory cluster.

A ongoing research project is currently using or extending our approach to autonomous learning and execution of actions. GRASPER addresses the problem of the “Autonomous Manipulation” of Underwater Vehicles, which involves locating objects, grasping and manipulating them into fulfilling a given task. It is desirable that the task is also identified autonomously by the robot, from the sensed environment information. The GRASPER represents only a sub-project inside a Spanish Coordinated Project, entitled: TRITON. It is a marine robotics research project focused on the development of intervention technologies really close to the real needs of the final user.

---

*Multisensory Based Underwater Intervention through Cooperative Marine Robots (TRITON).
and, as such, it can facilitate the potential technological transfer of its results.

In terms of applicability, we expect our methods could be exploited with significant economic impact in the following key areas:

- Elderly Care Robots
- Physiotherapy and Sports Assistant Diagnosis Tools
- Intelligence Surveillance Systems
- Autonomous Intervention Robots

To conclude, we feel intelligent robots will have its place in future society. In this thesis we have presented several approaches to unsolved challenges and/or limited solutions that are recurrent in action and person recognition systems. We hope that our work is able to increase robot’s autonomy and intelligence so as to given them the ability to co-exist and assist humans in their daily lives, so contributing to the development of truly autonomous social and/or service robots.
Appendices
Appendix A

Adaptive Sliding Window

Within the scope of this thesis, alternative solutions to classification have been investigated. We have applied a sliding window approach, as a mean to aggregate data into the computation of alternative representations. Classical sliding window approaches are defined by two, usually pre-defined, key parameters, window length and time shift, which defines the step between consecutive windows. Applying sliding window approaches, may present slow convergence to accurate classes or present low confidence, borderline decisions. This happens because, generally, sliding window approaches use fixed values for two key parameters: time shift and window size. In our research, we hypothesize that adapting these parameters during the classification process could present some benefits, allowing better and faster decisions from the model. The proposed solution is posed as an entropy minimization problem, using it as a feedback parameters for adapting sliding window parameters. The proposed solution were tested within the scope of this thesis’ author and Khoshhal’s joint work [SK13], where results show an improved classification framework, either in terms of classification speed and model confidence, allowing to reduce the delay between ground truth annotation and classified states, as well as reducing the number of borderline decisions, where despite selecting the correct state, the entropy was still high. The following subsections describe the developed adaptive sliding window approach, based on entropy feedback.

A.1 Definitions

The classification inference algorithms usually apply fixed parameter sliding windows. However, selecting optimal parameters is not easy. In fact, what can be a good parameter selection for a sequence, might fail to show correct segmentation when using
a different performer. Contrary to this classic sliding window approaches, we propose a method which continuously adapts the window parameters. Let us assume the following definitions:

- $h$ = Entropy value.
- $H$ = Entropy time series.
- $w$ = Window size.
- $w_d$ = Default window size.
- $W$ = Window size time series.

Consider that for a distribution $p = \{x_1, \cdots, x_n\}$, the maximum value for $\max(h) = \log(n)$. Bear in mind, entropy is a normalized value, upon the $\max(h)$, such that $h \in [0, 1]$.

### A.2 Window Size

#### A.2.1 Rationale

The rationale behind our approach is summarized in Table A.1. Let us use an example to further enlightenment. Assume the case where the entropy value $h_{t-1} < h_t$, i.e. from instant $t - 1$ to $t$ the model has become more uncertain. We analyse this phenomenon in light of the immediate past window sizes $w_{t-1}$. Whichever scale direction is observed (from the first order backward difference), it lead to a decreasing model confidence, therefore the window size needs to be corrected in the opposite direction. In cases where the scaling direction leads to increased model certainty, the window length should maintain scaling direction.

<table>
<thead>
<tr>
<th>$dH$</th>
<th>$h$</th>
<th>$dW$</th>
<th>$w$</th>
<th>$d^2H$</th>
<th>$h$</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>Worst</td>
<td>+</td>
<td>Increasing</td>
<td>N/R</td>
<td>N/R</td>
</tr>
<tr>
<td>0</td>
<td>Stable</td>
<td>+</td>
<td>Increasing</td>
<td>Increasing Tendency</td>
<td>N/R</td>
</tr>
<tr>
<td>0</td>
<td>Stable</td>
<td>+</td>
<td>Increasing</td>
<td>Decreasing Tendency</td>
<td>(+) Smaller Growth</td>
</tr>
<tr>
<td>-</td>
<td>Good</td>
<td>+</td>
<td>Increasing</td>
<td>N/R</td>
<td>N/R</td>
</tr>
<tr>
<td>+</td>
<td>Worst</td>
<td>-</td>
<td>Decreasing</td>
<td>N/R</td>
<td>N/R</td>
</tr>
<tr>
<td>0</td>
<td>Stable</td>
<td>-</td>
<td>Decreasing</td>
<td>Increasing Tendency</td>
<td>(+) Smaller Growth</td>
</tr>
<tr>
<td>0</td>
<td>Stable</td>
<td>-</td>
<td>Decreasing</td>
<td>Decreasing Tendency</td>
<td>(-) Smaller Shrinkage</td>
</tr>
<tr>
<td>-</td>
<td>Good</td>
<td>-</td>
<td>Decreasing</td>
<td>N/R</td>
<td>N/R</td>
</tr>
</tbody>
</table>

Table A.1: Summary of implicit signal rules. N/R = Not relevant.
There are however cases where consecutive instants have equal values for \( h \), i.e. \( h_{t-1} = h_t \), for which the backward difference is zero. When such event occurs, we replace the first order backward difference by its second order counterpart, which represents the growth tendency. Equivalent to analysing the second derivative for a continuous time series, we assume that upwards concavity represents tendency to increase and vice-versa. Bear in mind that by analysing a tendency, the scaling factor needs to be constrained when compared to using the first order difference.

### A.2.2 Formulation

In light of the presented rationale, the basic definition for the window length obeys the following equation:

\[
wt = (1 + \alpha)w_{t-1}
\]  \hspace{1cm} (A.1)

where \( wt \) is the window length at instant \( t \), and the variable \( \alpha = [\alpha_{min}, \alpha_{max}] \) a scaling factor such that:

\[
\left(1 + \alpha_{min}\right)w_{min} \leq wt \leq \left(1 + \alpha_{max}\right)w_{max}
\]  \hspace{1cm} (A.2)

The scaling direction according to the aforementioned rationale, is formulated mathematically as:

\[
- \frac{dH}{dt} \frac{dW}{dt}
\]  \hspace{1cm} (A.3)

For the special cases where \( \frac{dH}{dt} = 0 \), this argument is replaced by the second order backward difference \( \frac{d^2H}{dt^2} \).

\[
- \frac{d^2H}{dt^2} \frac{dW}{dt}
\]  \hspace{1cm} (A.4)

However, when \( \frac{dH}{dt} = 0 \), the second order difference is considered a weak indicator. Therefore, we propose two constraints \( a \) and \( b \), such that \( \frac{dH}{dt} \geq \frac{d^2H}{dt^2} \). From equations A.3 and A.4, we obtain:

\[
- \frac{dW}{dt} \left( a \frac{dH}{dt} + b \frac{d^2H}{dt^2} \right)
\]  \hspace{1cm} (A.5)

We must also consider the specific case where \( \frac{dW}{dt} = 0 \), which leads to \( \vec{h} = 0 \). Our solution is making \( \vec{\alpha} \) converge to the default window size, for which equation A.5 is rewritten as:

\[
(w_d - w) \left| a \frac{dH}{dt} + b \frac{d^2H}{dt^2} \right|
\]  \hspace{1cm} (A.6)
Figure A.1: Envelope function for the growth percentage. When $x \to \infty$ then $y \to 100\%$ 

where the derivatives no longer control the scaling direction, but rather relative window size with respect to the default value. The scaling direction $\vec{h}$ can then be summarized as:

$$\vec{\alpha} = \begin{cases} -\frac{dW}{dt} \left( a \frac{dH}{dt} + b \frac{d^2H}{dt^2} \right), & \text{if } \frac{dW}{dt} \neq 0 \\ (w_d-w) \left| a \frac{dH}{dt} + b \frac{d^2H}{dt^2} \right|, & \text{if } \frac{dW}{dt} = 0 \end{cases} \quad (A.7)$$

This latter formulation addresses direction, whereas the issue of scale, i.e. how much should the window grow or shrink needs to be address. We aim at obtaining a simple constant factor in the form of a percentage value. This factor should be proportional to the margins between the current and maximum/minimum values for window size. In addition, the selected function should be symmetric to the origin, meaning that the factor $\alpha$ should share the same signal as $\vec{\alpha}$. The function in equation (A.8) encompasses both of these properties.

$$\alpha = \frac{1}{k} \sqrt{(1 + 4\vec{\alpha}^2)} - 1 \quad (A.8)$$

where $k$ is a inverse proportional factor which may limit growth (default $k = 1$). Figure A.1 illustrates equation (A.8) for a clearer visualization. One should note that the window size must not scale beyond the limits defined in equation (A.2). Hence, the following formulation is proposed:

$$w_t = \begin{cases} w_{t-1} + \alpha |w_{\text{max}} - w_{t-1}| & \text{if } \vec{\alpha} > 0 \\ w_{t-1} + \alpha |w_{\text{min}} - w_{t-1}| & \text{if } \vec{\alpha} < 0 \end{cases} \quad (A.9)$$

which means that we are growing only a percentage of what is left within the window limits, assuring the window will never grow beyond them.
A.3 Time Shift

The time shift is a relevant parameter in sliding window approaches, as it defines two relevant properties: segment overlap and the time between each classification. Selecting an appropriate value might present itself as an easier task than with the size parameter. However, as previously stated, we hypothesize that adjusting the time shift can optimize the segmentation process, speeding up the classifier and reducing the redundancy and adjusting segment overlap accordingly. Let us consider the time shift $\Delta$ limits as defined in equation (A.10), which is a function of the acquisition frequency $f$.

$$\frac{1}{f} < \Delta < \frac{f}{\Delta_{\text{min}}} \quad \Delta_{\text{max}} \quad (A.10)$$

We will explore three different approaches, which are tested separately and are again based on the values of the entropy:

1. **Adapt1-$\Delta$**: When entropy is high, we want to apply short time shifts. This approach aims at an exhaustive exploration of the data, by augmenting the number of analysed samples per second. Although we recognize that increasing the number of samples in degenerate data samples will naturally increase the number of miss-classified samples, we expect true positive results to be in greater number, resulting in a better overall accuracy ratio. The proposed formulation for this first approach, is as follows:

$$\Delta_{t+1} = \frac{w_t - (h_t \ast w_t)}{f} \quad (A.11)$$

2. **Adapt2-$\Delta$**: During action class state transitions, entropy values tend to be higher. In this case, we hypothesize that forwarding the window to a time period where the new action is already well defined can reduce the number of false positive results. Hence, we want to extend the time shift to its maximum value, thus yielding a minimum successive window overlap. Therefore, we propose the following formulation, which reflects our idea:

$$\Delta_{t+1} = \frac{w_t - ((1-h_t) \ast w_t)}{f} \quad (A.12)$$

where $f$ stands for sampling frequency, $h_t$ the entropy at instant $t$ and $w_t$ the current window size measured in samples.
3. **Adapt3-$\Delta$**: We also consider interesting to study another approach when in the presence of action transitions, but addressing entropy when it becomes a volatile signal, i.e. it experiences big differences in consecutive computed values, which is reflected in its first derivative. Hence, to overcome this volatility effect, we consider the formulation in equation (A.12), integrating the $1^{st}$ order backward difference for the entropy signal, which results in:

$$\Delta_{t+1} = \begin{cases} 
  w_t - \left(1 - \nabla H \right) w_t, & \nabla H \geq \text{thr} \\
  w_t - \left(1 - h_t \right) w_t, & \nabla H < \text{thr}
\end{cases} 
$$

(A.13)

where $\nabla H = h_t - h_{t-1}$ corresponds to the $1^{st}$ order backward difference, and $\text{thr}$ a pre-defined numerical threshold.
Appendix B

UC-3D Motion Database

Action analysis and recognition is a very active research topic. The research group of the Mobile Robotics Laboratory from the Institute of Systems and Robotics, University of Coimbra releases the University of Coimbra 3-D Motion Database (UC-3D)\textsuperscript{*} so to promote this research. The database was developed in the context of an Action-based Person Recognition research work, for which it is also found suitable for research on action-based biometrics. Available data types encompass high resolution Motion Capture, acquired with MVN Suit from Xsens\textsuperscript{†} and Microsoft Kinect RGB and depth images. This diversity intents to provide the community with an homogeneous dataset for fair comparison between methods using different data types.

In the UC-3D Motion Database researchers will currently find 11 different activities, performed by 13 different actors. 6 of these actions are interactive, containing 2 different persons, for which the MVN data is only recorded for one of them. Each person performs 3 trials for each different action. Typical action duration is 3 seconds.

B.1 Data Types

The database encompasses 3 different data types:

- MVN Suit data
- RGB image sequences
- Depth image sequences

\textsuperscript{*}http://mrl.isr.uc.pt/experimentaldata/public/uc-3d/
\textsuperscript{†}http://www.xsens.com/
Appendix B. UC-3D Motion Database

Figure B.1: Different data types in the UC-3D Database.

Within the MVN Suit data, a user will find the linear and angular velocities and accelerations of each Inertial Measuring Unit (IMU), as well as relative Cartesian 3D pose. Figure B.1 shows an example of each acquired data type.

### B.2 Actions

The UC-3D database encompasses actions divided in two different categories: individual and interactive actions. Interactive actions are performed by two actors, for which the MVN data is only recorded for the dominant one. The following Table B.1 summarizes the available actions, while Figure B.2 illustrated an example frame for each one of them.
B.2. Actions

(a) Bend.  (b) Conversation.  (c) Handshake.

(d) Hugging.  (e) Jumping.  (f) Punching.

(g) Punched.  (h) Pushing.  (i) Running.

(j) Walking.  (k) Sit/Stand.

Figure B.2: Different actions in the UC-3D Database.
Table B.1: Actions in the UC-3D Database, example and description.

<table>
<thead>
<tr>
<th>Category</th>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Individual</td>
<td>Bend</td>
<td>An actor starts in a standing position and bends to pick an object from the floor.</td>
</tr>
<tr>
<td></td>
<td>Jump</td>
<td>An actor starts in a standing position and performs a cycle of three jumps in the air.</td>
</tr>
<tr>
<td></td>
<td>Run</td>
<td>An actor runs for about 3 metres.</td>
</tr>
<tr>
<td></td>
<td>Walk</td>
<td>An actor walks for about 3 metres.</td>
</tr>
<tr>
<td></td>
<td>Sit/Stand</td>
<td>An actor starts in a standing position, sits on a chair, rests for a couple of seconds and stands up.</td>
</tr>
<tr>
<td>Interactive</td>
<td>Conversation</td>
<td>Two actors stand in front of each other, making conversation while freely gesturing.</td>
</tr>
<tr>
<td></td>
<td>Handshake</td>
<td>Two actors approach each other and shake hands.</td>
</tr>
<tr>
<td></td>
<td>Hugging</td>
<td>Two actors approach each other and hug each other.</td>
</tr>
<tr>
<td></td>
<td>Punching</td>
<td>The dominant actor, steps forwards while punching the secondary actor, which tries to deviate.</td>
</tr>
<tr>
<td></td>
<td>Punched</td>
<td>The dominant actor tries to evade a punch, performed by the secondary actor.</td>
</tr>
<tr>
<td></td>
<td>Pushing</td>
<td>The dominant actor steps forward to push the secondary actor in the chest.</td>
</tr>
</tbody>
</table>
Bibliography


[PA] Daniel Gatica Perez and Oya Aran. Automatic analysis of group conversations via visual cues in nonverbal communication (novicom). EU FP7 Marie Curie - Intra-European Fellowship (IEF) project.


[SD] Luis Santos and Jorge Dias. Laban-based multilayer model for activity recognition and annotation. under peer review.


[Sen] Reactive tactile sensor test.


[Tay00] C. J. Taylor. Reconstruction of articulated objects from point correspon-
dences in a single uncalibrated image. In IEEE International Conference

[TCLZ12] YingLi Tian, Liangliang Cao, Zicheng Liu, and Zhengyou Zhang. Hier-
archical filtered motion for action recognition in crowded videos. IEEE
Transactions on Systems, Man and Cybernetics - PART C: APPLICA-

[TK09] Sergios Theodoridis and Konstantinos Koutroumbas. Pattern Recognition

[TLWM07] Dacheng Tao, Xuelong Li, Xindong Wu, and Stephen Maybank. General
tensor discriminant analysis and gabor features for gait recognition. IEEE
Trans. on Pattern Analysis and Machine Intelligence, 29(10):1700–1715,
2007.

[TSA03] C. Tomasi, S.Petrov, and A.Sastry. 3d tracking = classification + inter-

[Tuc66] Ledyard R. Tucker. Some mathematical notes on three-mode factor anal-

[UF04] Raquel Urtasun and Pascal Fua. 3d tracking for gait characterization and
recognition. In 6th IEEE International Conference on Automatic Face and

[Vas02] M. Alex 0. Vasilescu. Human motion signatures: Analysis, synthesis,

models. Pattern Analysis and Machine Intelligence, IEEE Transactions

[WRR03] Michael Wall, Andreas Rechtsteiner, and Luis Rocha. Practical Approach
to Microarray Data Analysis, chapter Singular value decomposition and

[WSNK10] Jin Wang, Mary She, Saeid Nahavandi, and Abbas Kouzani. A review of
vision-based gait recognition methods for human identification. In IEEE
International Conference on Digital Image Computing: Techniques and

[YLW09] Junsong Yuan, Zicheng Liu, and Ying Wu. Discriminative subvolume
search for efficient action detection. In IEEE Computer Vision Pattern
Recognition (CVPR09), 2009.

[YNC04] ChewYean Yam, Mark Nixon, and John Carter. Automated person recogni-
tion by walking and running via model-based approaches. Pattern Recogni-


