This copy of the thesis has been supplied on condition that anyone who consults it is understood to recognise that its copyright rests with its author and that no quotation from the thesis and no information derived from it may be published without the author's prior consent.

A DEVELOPMENTAL MODEL OF TRUST IN HUMANOID ROBOTS

by

MASSIMILIANO PATACCHIOLA

A thesis submitted to Plymouth University
in partial fulfilment for the degree of

DOCTOR OF PHILOSOPHY

School of Computing, Electronics and Mathematics

October 2018

Abstract

A developmental model of trust in humanoid robots
Massimiliano Patacchiola

Trust between humans and artificial systems has recently received increased attention due to the widespread use of autonomous systems in our society. In this context trust plays a dual role. On the one hand it is necessary to build robots that are perceived as trustworthy by humans. On the other hand we need to give to those robots the ability to discriminate between reliable and unreliable informants. This thesis focused on the second problem, presenting an interdisciplinary investigation of trust, in particular a computational model based on neuroscientific and psychological assumptions. First of all, the use of Bayesian networks for modelling causal relationships was investigated. This approach follows the well known theory-theory framework of the Theory of Mind (ToM) and an established line of research based on the Bayesian description of mental processes. Next, the role of gaze in human-robot interaction has been investigated. The results of this research were used to design a head pose estimation system based on Convolutional Neural Networks. The system can be used in robotic platforms to facilitate joint attention tasks and enhance trust. Finally, everything was integrated into a structured cognitive architecture. The architecture is based on an actor-critic reinforcement learning framework and an intrinsic motivation feedback given by a Bayesian network. In order to evaluate the model, the architecture was embodied in the iCub humanoid robot and used to replicate a developmental experiment. The model provides a plausible description of children's reasoning that sheds some light on the underlying mechanism involved in trust-based learning. In the last part of the thesis the contribution of human-robot interaction research is discussed, with the aim of understanding the factors that influence the establishment of trust during joint tasks. Overall, this thesis provides a computational model of trust that takes into account the development of cognitive abilities in children, with a particular emphasis on the ToM and the underlying neural dynamics.

Contents

Abstract **i**

Table of Contents **ii**

List of Figures **v**

List of Tables **ix**

Acknowledgements **xi**

Author's declaration **xiii**

1 Introduction **1**
- 1.1 Motivations 1
- 1.2 Expected Contribution to Knowledge 4
- 1.3 Overview 4

2 Background **7**
- 2.1 Introduction 7
- 2.2 Computational Models of Trust 8
- 2.3 The Neuroscience of Trust 9
- 2.4 The Development of Trust 11
- 2.5 Trust in Human-Robot Interaction 13
- 2.6 Contribution 15

3 Methods and Materials **17**
- 3.1 Developmental Robotics 17
  - 3.1.1 Definition 17
  - 3.1.2 The iCub Humanoid Robot 18
  - 3.1.3 The Epigenetic Robotic Architecture 19
- 3.2 Bayesian Networks 22
  - 3.2.1 Definition 22
  - 3.2.2 Inference 23

3.2.3 Learning the Network Parameters 23
3.2.4 Issues related to the use of Bayesian networks 25
3.3 Artificial Neural Networks and Deep Learning 27
3.3.1 Definition 27
3.3.2 Convolutional Neural Networks 28
3.3.3 Adaptive Gradient Methods 30
3.4 Reinforcement Learning 32
3.4.1 Definition 32
3.4.2 Intrinsically Motivated Reinforcement Learning 33

4 A Developmental Bayesian Model of Trust **37**
4.1 Introduction 37
4.2 Model Description 38
4.3 Experiments 40
4.3.1 Methods 41
4.3.2 Results 43
4.4 Discussion 46
4.5 Conclusions 47

5 Head Pose Estimation **49**
5.1 Introduction 49
5.2 Overview 51
5.3 Related Work 52
5.4 Experiments 54
5.4.1 Prima head-pose dataset 57
5.4.2 Annotated Facial Landmarks in the Wild 61
5.4.3 Annotated Face in the Wild 67
5.4.4 Discussion 70
5.5 Conclusions 72

6 A Cognitive Architecture for Trust in Humanoid Robots **75**
6.1 Introduction 75
6.2 Model Description 77
6.2.1 Comparison with neurobiological correlates 77
6.2.2 Functional overview 79
6.2.3 The actor-critic reinforcement learning loop 81
6.2.4 The ERA architecture as function approximator 84
6.2.5 Internal environment and Bayesian networks 85
6.3 Experiments 88

6.3.1 Methods 90
6.3.2 Results 95
6.4 Discussion 98
6.5 Conclusions 99

7 Trust in Human-Robot Interaction 101
7.1 Introduction 101
7.2 Experiment 102
7.2.1 Methods 103
7.2.2 Results 103
7.2.3 Discussion 105
7.3 System architecture 107
7.4 Conclusions 109

8 Conclusions 111
8.1 Overview 111
8.2 Summary of the Contribution to Knowledge 111
8.3 Future Work 113
8.3.1 Adding an episodic memory in the Bayesian network model 113
8.3.2 Social trust in a multi-agent scenario 114
8.3.3 On the predictive power of the model 115

Bibliography 119

List of Figures

3.1 The iCub humanoid robot. The iCub robot has been widely used by the research community, and it is considered one of the best robotics platform for studying cognitive phenomena in a bio-plausible way. 18

3.2 The iCub simulator. The simulator allows testing the algorithms before deploying them in the real platform. 19

3.3 The ERA modules can be arranged in different ways in order to handle complex problems. The SOM can be represented as a vector or a matrix, made of single units (black dots). Multiple SOMs can be connected through Hebbian connections (bidirectional arrows) creating an ERA module. (a) Using a vision and a vocabulary SOM it is possible to have an ERA module representing a value function approximator. (b) Connecting a vision module and a vocabulary SOM to a belief array, it is possible to have an ERA module representing a state-belief function approximator. 20

3.4 An example of Bayesian Network (BN). This BN has two nodes, the first node X is the parent and the node Y is the child. Each node has an associated probability distribution represented with a table in the discrete case. The table of the parent node only has two parameters, whereas the table of the child node has four parameters. 23

3.5 An example of Convolutional Neural Network. A greyscale image of size 64×64 pixels is passed as input to the network and padded with two columns/rows of zeros on the four borders. The first layer performs a convolution with 32 kernels of size 5×5 (stride 1) generating 32 features map of size 64×64. A pooling operation (stride 2) is then performed on the feature maps resulting in a set of 32 features of size 32×32. Dense connections are used to associate the feature maps to standard units (128). Finally, the output layer (16 units) is connected to the previous one by dense connections. 29

3.6 The reinforcement learning cycle represented by the agent-world interaction, through action, reward, and state. 33

3.7 The intrinsic reinforcement learning cycle represented by the agent interaction with the external world passing through an internal environment. . . 34

4.1 The BNs integrated in the agent's cognitive system. The network on the left represents the relation between the child and the unreliable informant (tricker), the network on the right represent the relation between the child and the reliable informant (helper). The subscript U (unreliable) and R (reliable) has been used to distinguish the informants' belief and action, whereas the subscript C represents the child. 41

5.1 Grapichal representation of a Convolutional Neural Network with two convolutional (C1 and C2), two subsampling (P1 and P2) and two fully connected (D1 and D2) layers. The label above each layer specifies number of elements and size (rows × columns) of the future maps. For the dense layers the number of units is reported. The following colour convention has been used to identify the different layers: green for convolution, orange for subsampling, and red for dense layers. 55

5.2 Comparison of the four architectures used in the experiments. The label below each layer represents the number and size (rows × columns) of the future maps. For the dense layers only the number of units is reported. The networks are organised in order of complexity, on the left there is the network with less parameters and on the right the network with more. Network A has 4.3×10^6 parameters, whereas network B has two more layers and a total of 4.6×10^6. Network C has 8.5×10^6 parameters, whereas network D has two more layers for a total of 9.0×10^6 parameters. The following colour convention has been used to identify the different layers: green for convolution, orange for subsampling, and red for dense layers. 56

5.3 This figure represents a collection of images taken from the Prima dataset as they appear after cropping and scaling. 57

5.4 Comparison of different optimisers (trained on network A) for the estimation of the pitch angle on the Prima dataset. 62

5.5 Comparison of different optimisers (trained on network A) for the estimation of the yaw angle on the Prima dataset. 63

5.6 Comparison of the performances of the four networks trained with the Adam optimiser in pitch and yaw estimation of unknown subjects (Prima dataset). The Standard Deviation (STD) has been shrunk by a factor of two for graphical reason. 64

5.7 Representation of the confusion tables for pitch (left) yaw angle (right) of the best architecture (A) on unknown subjects (Prima dataset). Each row of the tables represents the instances in a predicted class while each column represents the instances in an actual class. 64

5.8 This figure represents a collection of images which are comparable to the Annotated Facial Landmarks in the Wild (AFLW) and the Annotated Face in the Wild (AFW) datasets (the licenses do not allow publishing the original images). The AFLW and the AFW datasets contain a large variety of appearances (age, ethnicity, occlusions, expressions, etc), lighting and environmental conditions for both genders. 65

5.9 Comparison of the convergence speed between the six optimisers used to train architecture A for yaw estimation on the AFLW dataset. The loss values are the mean of the five fold. 68

5.10 Comparison in term of Mean Absolute Error (MAE) between four architectures for roll pitch and yaw using the RMSProp optimiser on the AFLW dataset. The STD has been shrunk by a factor of two for graphical reason. 69

5.11 Representation of the confusion tables for roll (left), pitch (centre) and yaw (right) of the best architecture (B) trained with the RMSProp optimiser on the AFLW dataset. Each row of the tables represents the instances in a predicted class while each column represents the instances in an actual class. 69

6.1 Difference between a standard agent (a), an intrinsically motivated agent (b) and the particular case considered here (c). The standard agent (a) acquires the state of the world from the external environment and performs actions in the external environment which returns a reward. The intrinsically motivated agent (b) has both an internal and an external environment, it can convert beliefs to actions and it can evaluate the weight of external rewards based on an internal cost. In the case considered here (c) there is no feedback from the external environment and learning is achieved only through an internal cost measure. 78

6.2 Comparison between the biological model of trust (top), an the computational model presented here (bottom). Both the architectures are based on an actor-critic loop, that is controlled by the dopamine and the temporal-difference error. The projections from the ACC and the medial frontal cortex influence the activity of the critic, they have been modelled as a Bayesian network. The ERA architecture has been used as function approximator in a actor-critic mechanism. 80

6.3 The cognitive architecture implemented in the model. The learning process is represented by 6 steps. In step 1 the information acquired from the environment is used to activate the Self Organizing Maps (SOMs) in actor and critic Epigenetic Robotic Architecture (ERA) units. In step 2 the output of the actor ERAunit is computed based on the activation of vision and vocabulary SOMs. In step 3 a Softmax function is used to squash the raw output into a probability distribution. In step 4 the internal environment estimates the cost using the Bayesian Network (BN) forward step. In step 5 the temporal differencing update rule is used to stabilise the Hebbian connections of the critic unit. In step 6 the error Delta is used to stabilise the actor unit. 81

6.4 The stabilisation, learning and estimation phases in the BN architecture. In the stabilisation phase random inputs are used in order to stabilise the values of a SOM. In the learning phase the connections between active units are strengthened (positive Hebbian learning) or weakened (negative Hebbian learning). In the estimation phase the active units in a subset of the SOMs are used to estimate the activation value of inactive units. 85

6.5 The Bayesian network used in the internal environment. For each informant a reputation distribution is kept in a long term memory storage. The reputation, agreement and confidence distributions are used to estimate a measure of cost. 87

6.6 The Bayesian network with the associated node distributions when a reliable informant suggestion is in not in accordance with the agent belief and the agent is not confident. The star symbol used in the agreement node represent an evidence (agree). The cost sampled from the posterior distribution will be with 15% probability null, with 10% probability negative and 75% positive. 89

6.7 The bayesian network with the associated node distributions when an unreliable informant suggestion is not in accordance with the agent belief and the agent is not confident about the current state. The star symbol used in the agreement node represent an evidence (disagree). The cost sampled from the posterior distribution will be with 45% probability null, with 25% probability negative and 30% positive. 90

6.8 The object learning and familiarisation phases. In the object learning phase the caregiver (grey t-shirt) teaches the names of common objects to the robot. The image shows three of those objects: ball (blue), cup (green) and dog (yellow). In the familiarisation phase both the reliable and unreliable informants show familiar objects to the robot. The reliable informant (blue t-shirt) always gives the correct name to the object. The unreliable informant (red t-shirt) always gives a wrong name. 93

6.9 The explicit judgement phase. the caregiver (grey t-shirt) asks to the robot which informant was not god at answering questions. The robot recalls from the long term memory the reputation distribution returning the name of the unreliable informant. 94

6.10 The endorsement phase. Both the reliable (blue t-shirt) and the unreliable (red t-shirt) informants suggest a name for a new object. After the interaction the caregiver (grey t-shirt) asks for the name of the object. . . . 95

6.11 The results obtained in the explicit judgment task (mean and standard deviation) in two consecutive trials. In this task children and robot have to identify the unreliable informant. Both 3-year-olds and 4-year-olds performed significantly above chance meaning that they could identify the unreliable informant. The robot performed similarly to children and above chance level. 96

6.12 The results obtained in the endorse task (mean and standard deviation) in three consecutive trials. In this task children and robot have to provide the same label given by the reliable informant to an unknown object. Only 4-year-olds performed above chance providing the right label. The robot performed similarly to children in both conditions. 97

7.1 The robots used in the HRI experiments. (a) SCITOS G5. (b) Baxter (right). (c) two NAO robots. (d) Pepper. 102

7.2 The results obtained in the experimental session. 104

7.3 The architecture of the system used in the HRI experiments. 106

7.4 An example of GUI used in the HRI experiments. The interface is essential and is composed of a start button, investment buttons (from 0 to 10), score of both robot and participant, an energy bar representing the investment of the robot, and a central window that guides the user through the different phases of the experiment. In another window the researcher can set the XML file and the data of the subject. 107

List of Tables

4.1 Agent with Mature ToM. The table on the left represents the outputs of the BN after the interactions with the reliable pointer (helper), the table on the right represent the outputs after the interactions with the unreliable pointer (tricker). The rows of the tables represent the posterior probability distributions associated with each node of the networks, for both decision making (DM) and belief estimation (BE) tasks. 44

4.2 Agent with Immature ToM. The table on the left represents the outputs of the BN after the interactions with the reliable pointer (helper), the table on the right represent the outputs after the interactions with the unreliable pointer (tricker). The rows of the tables represent the posterior probability distributions associated with each node of the networks, for both decision making (DM) and belief estimation (BE) tasks. 44

5.1 In this table are compared the results of the proposed method with the result obtained by other authors on the Prima head pose dataset. These results have been obtained training architecture A for 20000 epochs on both unknown (18.75 hours) and know (4.2 hours) test with the Adam optimiser. The results are in term of MAE(accuracy), with MAE expressed in degrees and Accuracy expressed in percentage. The best scores are in bold. 61

5.2 In this table are reported the results obtained on the Prima dataset for leave-one-out (unknown subjects) test using different optimisers. These results have been obtained training the architecture A for 20000 epochs (18.75 hours). The results are in terms of MAE (accuracy) with MAE expressed in degrees and Accuracy expressed in percentage. The best scores are in bold. 61

5.3 In this table are reported the results obtained on the AFLW dataset for the five-fold cross validation test. These results have been obtained training the architecture A for 30000 epochs (14.6 hours). The results are in terms of MAE(accuracy). MAE is expressed in degrees and Accuracy is expressed in percentage. The best scores are in bold. 67

5.4 In this table are reported the results in term of MAE (degrees) and accuracy (percentage) obtained with different methods for the yaw estimation on the AFLW and the AFW datasets. The results have been obtained training the architecture B for 30000 epochs (14.6 hours) with the RMSProp optimiser. MAE is expressed in degrees and Accuracy is expressed in percentage. The best scores are in bold. 72

6.1 Conditional probability table of the cost node in the BN of the internal environment. The first three rows of both upper and lower tables represent the parent node values and the last row (in bold) the value assumed by the cost. The opinion of the unreliable informant is discarded (null cost) when the agent is unconfident, and it is acquired (negative cost) in case of confidence and disagreement. 91

6.2 Conditional probability table of the cost node in the BN of the internal environment in case of an immature ToM. The first three rows of both upper and lower tables represent the parent node values and the last row (in bold) the value assumed by the cost. The opinion of the unreliable informant is taken into account with the same weight of the reliable informant. 91

Acknowledgements

> *"For those who believe in God, most of the big questions are answered. But for those of us who can't readily accept the God formula, the big answers don't remain stone-written. We adjust to new conditions and discoveries. We are pliable. Love need not be a command or faith a dictum. I am my own God."*
>
> – Charles Bukowski, *Life Magazine*, 1988

Author's declaration

At no time during the registration for the degree of Doctor of Philosophy has the author been registered for any other University award without prior agreement of the Doctoral College Quality Sub-Committee.

Work submitted for this research degree at the University of Plymouth has not formed part of any other degree either at the University of Plymouth or at another establishment

This study was financed with the aid of a studentship from the Air Force Office of Scientific Research, Air Force Materiel Command, USAF under Award No. FA9550-15-1-0025 and carried out in collaboration with the Centre for Robotics and Neural Systems (CRNS) at University of Plymouth, under the supervision of Prof. Dr. Angelo Cangelosi, Prof. Dr. Torbjorn Dahal, and Prof. Dr. Giorgio Metta.

A programme of advanced study was undertaken, which included: courses organised within the Researcher Development Programme; PGCAP-A Postgraduate Certificate in Academic Practice; introduction to CUDA and deep learning with DIGITS; HRI Summer School 2015; supervision of Luca Surace during his MSc. project.

Publications:

- **Patacchiola, M.**, & Cangelosi, A. (2017). Head pose estimation in the wild using convolutional neural networks and adaptive gradient methods. Pattern Recognition, 71, 132-143. DOI: `https://doi.org/10.1016/j.patcog.2017.06.009`
- Vinanzi, S., **Patacchiola, M.**, Chella, A., & Cangelosi, A. (2019). Would a robot trust you? Developmental robotics model of trust and theory of mind with episodic memory. Philosophical Transactions B. Biological Sciences. Royal Society of London. PEARL: `http://hdl.handle.net/10026.1/12413`

Presentations at conferences:

- **Patacchiola, M.**, & Cangelosi, A. (2016). A developmental Bayesian model of trust in artificial cognitive systems. In Development and Learning and Epigenetic Robotics (ICDL-EpiRob), 2016 Joint IEEE International Conference on (pp. 117-123). IEEE. DOI: `https://doi.org/10.1109/DEVLRN.2016.7846801`
- Zanatto, D., **Patacchiola, M.**, Goslin, J., & Cangelosi, A. (2016). Priming Anthropomorphism: Can the credibility of humanlike robots be transferred to non-humanlike robots?. In The Eleventh ACM/IEEE International Conference on Human Robot Interaction (pp. 543-544). IEEE Press. DOI: `https://doi.org/10.1109/HRI.2016.7451847`

Word count for the main body of this thesis: 45337

Signed Massimiliano Patacchiola

Date 23/10/2018

Chapter 1

Introduction

1.1 Motivations

The motivation behind this thesis is to investigate trust in artificial systems and to define a computational model that can be embodied in robots. In particular, the main objective is the computational modelling of the cognitive abilities underlying trust relationships in humans, with a particular focus on the development of trust in children. The long term goal is to provide a computational model that can be integrated in artificial cognitive systems, and to join it with recent findings in humanoid robots credibility (Goetz et al., 2003; Zanatto et al., 2016), moving toward an improved cooperation between humans and robots.

The research followed the guidelines defined by the developmental robotics approach (Cangelosi & Schlesinger, 2015), describe in Chapter 3. However, given the breadth of this work it has been necessary to adopt an interdisciplinary view and to consider hints and methodologies from different fields. In this thesis, machine learning, developmental psychology and neuroscience are tightly connected and converge into a holistic representation of trust.

Before describing the role of trust in robotics it is important to provide a formal definition. In literature trust has been described from different perspectives. From a generic point of view, trust is the reliance on or confidence in the dependability of someone or something (VandenBos, 2015). From a psychodynamic perspective, trust is a basic sense of faith in the self and the world (Erikson, 1950). Moreover, trust can be defined from a sociological point of view as integral to the organisation, survival, and efficiency of society, and societies’ local, national, and international relations (Rotter, 1967). Other theorists have delineated multiple forms of trust. For example, Earl (1987) identified three types of trust: trust in others, self-efficacy, and self-trust. A particular form of trust called *epistemic trust* was introduced by Shafto et al. (2012). Epistemic trust relates to the knowledge of the trustee in a particular context or situation. For instance, in some cases perceiving an informant as knowledgeable may increase the level of trust. However generally this is not enough, because it is also important to understand her intent (helpful or deceptive). This consideration connects epistemic trust to the Theory of Mind (ToM), a link that will be further discussed later in this thesis. An exaustive definition of trust is given by Falcone & Castelfranchi (2001), who link it to the cognitive abilities of the agent and in particular to its belief and goals. In the article the authors define trust via multiple assertions, the first two having a particular relevance in this thesis: (i) only an agent endowed with goals and belief can trust another agent; (ii) trust is a mental state, a complex attitude of an agent x towards another agent y about the behaviour/action relevant for the goal g. The first assertion links trust to the terms *goals* and *belief*, pointing out two component that must be included in a computational model of trust. The second assertion defines a convention that allows discriminating the agents: the agent x is the relying agent, who feels trust, and the agent y is the trustee. This definition has the advantage of being compact and immediate, and it will be used extensively in this work.

Trust is an important topic in robotics. When robots interact with users in unstructured environments, a certain level of cooperation is required. In this sense, the reliability of a robotic platform is fundamental in order to encourage the adoption of those platforms. An example is provided by military robots, that are used every day by soldiers for bomb disposal. In this context it has been observed that perceived trust is a very important factor in order for these systems to be used by the soldiers (Hancock et al., 2011). More broadly, an effective social robot must have the ability to communicate, interact, and empathise with us (Breazeal, 2004). However the precursor of such a complex set of skills is trust, because if the user does not trust the robot then none of those skills can be triggered.

Trust has another role in robotics: the robot must discriminate between different informants and acquire new knowledge only from the reliable ones. An example of the importance of this point has been given by a recent accident involving an artificial agent. Microsoft released an experimental chat bot called Tay, with the purpose of engaging users in conversations. The idea was to allow Tay to learn from interaction with users through social networks. The experiment went wrong and the chat bot was shut down after one day because of its obscene tweets (Neff & Nagy, 2016). What happened with Tay is an example of acquisition of knowledge from unreliable sources and it is one of the risks of a wrong trust evaluation.

In humans the ability to discriminate between reliable and unreliable informants is not an innate skill. As shown in the psychological literature (Koenig & Harris, 2005; Vanderbilt et al., 2011), there are different developmental stages that influence its acquisition. In particular, there is consistent literature confirming that by the age of 4, children are able to track the reliability of informants, and prefer the most accurate source (Fusaro et al., 2011; Jaswal & Neely, 2006). A lack of cognitive skills may interfere with the process of source selection, generating some errors in inferring. In the context of trust, one of the most important skills is the ability of reading others' belief, known as Theory of Mind (ToM) (Premack & Woodruff, 1978). Several studies focused on the link between trust and ToM finding that immature ToM abilities may cause errors to be made in estimating the informant's reliability (Koenig & Harris, 2005; Moses & Baldwin, 2005; Vanderbilt et al., 2011).

In this thesis a model that integrates trust and ToM into an unified scheme is described. The model confirms that under particular conditions children use a cause-effect strategy to learn and predict future events. The work does not focus on explaining how ToM is acquired, but instead aims to incorporate aspects of ToM inside a model of trust to explain how it affects decision making and belief estimation. The model predicts the behaviour of children with a mature ToM, and explains some of the mistakes made by children with an immature ToM and is discussed in Chapter 4. The second stage of this thesis consisted in integrating the model into a wider framework, taking into account the embodiment provided by a robot. The framework includes a head pose estimator, with the aim of estimating the focus of attention of a user who is engaging in a joint task with the robot. Finally, the aforementioned work was integrated into a cognitive architecture, based on psychological and biological observations, reaching the final objective.

It is worth mentioning that this thesis is part of a larger project called Trust in Human Robot Interaction Via Embodiment (THRIVE). The THRIVE project aimed to investigate the role of the embodiment in the development of trust between human and robots, in particular focusing on joint decision tasks. The project was organised into two research objectives: (i) to define a developmental cognitive architecture for autonomous robots to support and sustain trust during interactions with humans; (ii) to carry out Human-Robot Interaction (HRI) experiments validating the role of embodiment in the development of trust. The role of the author was within the first objective. However, the he gave a minor contribution to the second part that is shortly discussed in Chapter 7.

1.2 Expected Contribution to Knowledge

It is helpful to summarise in this section the objectives of the thesis and the expected contribution to knowledge. A complementary discussion is presented in the final chapter and it will revisit what has been discussed here. As stated at the end of the previous section, the objective of this thesis was to create a computational model of trust using a developmental robotics approach. To achieve this goal, three sub-tasks have been defined, these are summarised below:

1. Implementation of a flexible theoretical framework to link trust and developmental psychology, that takes into account how the cognitive processes of children evolve with time.
2. Definition of how trust is connected with embodiment and how the embodiment can be exploited. Developing a flexible machine learning model to sustain social robots in building trustworthy relationships with users.
3. Development of a cognitive architecture that merges previous theoretical work into a coherent framework, taking into account developmental psychology, neuroscience, and the embodiment provided by a robotic platform.

These three tasks have been designed together, but tackled in three separate phases. In the next section these phases are related to the structure of the thesis providing an overview of the work that has been done.

1.3 Overview

This thesis consists of 8 chapters organised similarly to the workflow followed by the author during his PhD. Most of the content has already been published in peer-reviewed conferences and journals. In this section the content of each chapter is shortly described, moreover it is explained how those chapters are linked together.

Chapter 2 provides a literature review on trust with a particular focus on previous computational models, neuroscientific literature, psychological theories and HRI experiments. This chapter is important because it provides an overview on the complex field of trust. The last part discusses the expected contribution to knowledge.

Chapter 3 provides an overview on the methods used in the research and aims to familiarise the reader with specific machine learning techniques.

Chapter 4 describes the first formalisation of a Bayesian network for modelling the developmental process underlying trust in 3 and 4 year old children. This work is connected with developmental theories of how the ToM develops in children and how it impacts on the establishment of trust.

Chapter 5 describes a state-of-the-art method for head pose estimation based on Convolutional Neural Networks (CNNs) that has been used as component of the cognitive architecture. Head pose estimation has a main role in trust. It has been shown that estimating the focus of attention of others is a fundamental ability for establishing trust, and it is particularly helpful in joint tasks.

The Chapter 6 presents the integration of the work that has been done previously. The resulting cognitive architecture is introduced and described. In order to validate the architecture a developmental experiment has been reproduced and the results compared with the original work.

In Chapter 7 the contribution given to the second branch of the THRIVE project is briefly described. This concerns the design and implementation of a control architecture for HRI experiments.

Chapter 8 concludes the thesis with a summary of the main contributions, future work, and take-away lessons.

Chapter 2

Background

2.1 Introduction

Relevant literature and the theoretical background of the research are introduced here, however some aspects that are related to specific experiments are explained in the methodological section of each chapter in order to add clarity and context. Trust is a broad topic that has been investigated from different perspectives. For this reason is not easy to present a concise literature review on the argument. To simplify this chapter it is helpful to follow the Marr's levels of analysis (Marr & Vision, 1982), describing trust from the computational, representational, and biological point of view. However, the reader should bear in mind that this thesis advocates a holistic explanation of trust. Therefore the discussion has been divided into different sections only for the sake of clarity. This chapter is organised as follows:

- In Section 2.2, an overview of the different computational models of trust available in the literature is presented.
- In Section 2.3 recent advances in the understanding of the neurobiological correlates of trust are discussed.
- In Section 2.4 the emergence of trust during child development is discussed.
- In Section 2.5 a brief overview of trust in HRI is provided.
- Finally, in Section 2.6 the relations between this thesis, previous work and some of the most relevant differences are discussed.

2.2 Computational Models of Trust

The scientific research in the area of computational trust has increased in the last two decades thanks to virtual societies and e-commerce. There are numerous computational models of trust and reputation for economical exchange and e-business. These models have been used in the context of multi-agent systems, those systems composed of autonomous agents that need to interact with each other in order to achieve their goal. Since the intentions of the agents are unknown it is necessary to protect good agents from the fraudulent ones. A recent review (Pinyol & Sabater-Mir, 2013) classified those models in three ways: the security approach, the institutional approach, and the social approach. In the security approach a low level encryption of the information is used in order to secure the transit of information. In the institutional approach a central authority controls the agents actions, punishing the fraudulent entities. Finally in the social approach, agents themselves are able to identify undesired behaviours and exclude some partners from the network.

A conceptual classification of computational trust has been given by Sabater & Sierra (2005), where trust was characterised in a cognitive and game-theoretical way. In the cognitive model of trust the mental states of the others and the mental consequences of relying on another agent play an essential role. In the game-theoretical approach trust is defined as a subjective probability by which an agent expects another agent to perform an action on which its own welfare depends. A line of research connected with the game-theoretical approach is based on game theory. In Schillo et al. (2000) a graph model of trust has been used to predict the choices of multiple agents in a Prisoner's dilemma set of games. The interaction has been represented as a binary impression (good or bad) on the honesty of the partner. Another line of research in the game-theoretical setting is based on investment games. There are different types of economic games that have been used by psychologists and economists to probe social computations underlying trust. The standard version of the investment game has been proposed in Berg et al. (1995). An investor is given money to keep or invest with a trustee. The money will triple automatically when received by the trustee. The trustee then decides how much money to return. In this two-person game the players must consider their own actions and those of the opponent in order to maximise the income. A symbolic cognitive architecture that explains transfer of learning across strategic interactive games, has been proposed in Juvina et al. (2015).

The architecture was developed using the ACT-R (Adaptive Control of Thought-Rational), a framework that is used to create computational models of various tasks Anderson (1996). The model included a sort of trust accumulator, represented as a variable that increases when the opponent makes a cooperative move and decreases when the opponent makes a competitive move. Recently another kind of games, called Markov games, have been used in order to study trust in multi-agent systems. In Rishwaraj et al. (2017) the authors proposed a trust estimation model built on top of a reinforcement learning framework, that used an adaption of the concepts of Markov games and heuristic in sample experience evaluation. Using temporal difference learning the model can empirically estimate the trust of an agent in a group. A similar ideas has been adopted in Leibo et al. (2017) under the name of sequential social dilemmas. The authors analysed the dynamics of policies learned by multiple self-interested independent learning agents on two Markov games, showing how the learned behaviour evolved over time. In particular it has been observed that cooperation or conflict are related to the availability of shared resources in the environment. The same scheme has also been applied to model a common-pool resource appropriation (Pérolat et al., 2017), where if an agent obtains an individual benefit, the remaining amount available for appropriation by others is diminished. In this setting cooperation and trust may play an important role, and the results showed that cooperation only happened in the first phases of the training, with a catastrophic depletion happening when agents learned to harvest the resources as quick as possible.

2.3 The Neuroscience of Trust

The cognitive architecture presented in Chapter 6 is strictly connected to the biological correlates of trust. For this reason this section is particularly important, since it provides a review of the main results achieved so far by the neuroscience community and provides a theoretical framework that justifies the proposed architecture.

In the mammalian brain the interaction between ventral and dorsal striatum, modulated by dopamine, is involved in trial-and-error learning. It is often referred to the ventral striatum as the critic and to the dorsal striatum as the actor (Takahashi et al., 2007). These two modules are tightly connected. On the one hand the dopamine released from the substantia nigra pars compacta in the critic is responsible for the activity of the actor. On the other hand the activation of a sub-region of the dorsal striatum, the caudate nucleus, diminishes as over time cues begin to predict correct actions and outcomes (Delgado et al., 2005b). This result seems to suggest that the caudate may have the role of guiding future actions through a feedback loop. A similar activity pattern has been observed in trust games, in which participants had to learn whether the other players were reliable. A modified version of the standard trust game proposed in Berg et al. (1995) has been used in a large-scale fMRI study where neural responses from both partners have been recorded during a multiround interaction (King-Casas et al., 2005). The signal in the striatum of the player predicted the likelihood to reciprocate investment. The magnitude of the signal correlated with the intention to trust in the next round of the game. The peak of the signal shifted in time as the reputation developed. These results confirm the role of the striatum in trust acquisition and the presence of a reward prediction error that resembles the temporal difference in reinforcement learning. A review on the subject has been done in Montague et al. (2006). The authors hypothesised that abstractions like the intention to trust can act themselves as primary rewards. Humans can build a model of what to expect from other agents and generate a signal in the striatum and the midbrain, mimicking a reward prediction. This hypothesis, and in particular the role of prior social and moral information, has been further investigated in Delgado et al. (2005a). The authors find out that the participants of a trust game were more likely to make risky choices with reliable partners, meaning that prior perception can diminish the reliance on the feedback mechanism part of the reward-based learning circuitry. Moreover it shows that the perception of a "good" profile creates a prior belief that is used to modulate the reward. It is still not clear which brain area is responsible for the aggregation of prior belief into the backward signal to the striatum. A possible candidate is the Anterior Cingulate Cortex (ACC), a distributed neural system implicated in the representation and updating of decision values (Samejima et al., 2005). Some fMRI experiments based on the free-choice paradigm have been done in order to confirm the role of this area in trial-and-error learning. The observations confirm that projections from the ACC to ventral striatum modulate the current prediction error with respect to the next value estimate (Behrens et al., 2007).

There is also a place for the ToM in the biological picture of trust. A dorsomedial prefrontal region has been found to be active during ToM games played against other individuals but not played against computers (Saxe, 2006). The activity of the same area has been linked to a long series of ToM related tasks such as: watching animation of objects that elicit attributions of mental states, cooperative games in which success requires reading the others' belief, competitive games that require interpreting the intentions and goals of the others. In all these cases an activation of the medial frontal cortex (Amodio & Frith, 2006) has been observed. Although the involvement of this area seems evident, it is still not clear how it can influence the striatum. It is difficult to test with precision the function of all these regions, because of the complex set of mental processes recruited in ToM-related tasks (Behrens et al., 2009).

Taking those studies into account a reinforcement learning framework based on the temporal differencing method proposed in Sutton (1984) has been implemented. The temporal differencing algorithm has been included into an Actor-Critic (AC) module that resembles the ventral and dorsal subdivision of the striatum observed in the mammalian brain (Takahashi et al., 2007). The role of previous information about the trading partners has been modelled as a Bayesian network. The network has been included into an internal environment that represents the contribution of brain areas involved in planning, such as the ACC and the dorsomedial prefrontal region. Similarly to the ACC projection to the dorsal striatum the Bayesian network has been wired to the critic in the AC module. A wider description of these components is provided in Chapter 6.

2.4 The Development of Trust

Our point of view on trust is based on a developmental perspective, and our aim is to find a deeper explanation of trust studying how it evolves at different stages. Going back to the definition given in Falcone & Castelfranchi (2001) and introduced in Section 6.1, we defined the agent x as the relying agent, and the agent y as the trustee. We are interested in the particular case of x being a child and y an adult. The shift from adults to children introduces new dynamics, the most important being the ability of reading others' belief through a mature ToM. The ToM has a significant impact on trust and there are several studies that focused on understanding the link between these two factors. In particular it seems that a lack of ToM may cause error in estimating the informant's reliability (Koenig & Harris, 2005; Moses & Baldwin, 2005; Vanderbilt et al., 2011; Heyman et al., 2013).

The ToM is a well known concept in psychology. Nowadays there are four main approaches that are investigating the ToM: the modularity theory (Leslie, 1994), the theory-theory (Gopnik & Meltzoff, 1997), the rationality theory (Dennet, 1987), and the simulation theory (Gordon, 1986). Among those approaches the theory-theory (Gopnik & Meltzoff, 1997) seems to be the most promising because it can be easily formalised in a computational framework. The theory-theory is based on the assumption that children generate theories about physical phenomena, belief, and desire. In particular it states that children make transitions from simple theories to more complex ones following linear developmental stages. The advantage of using such an approach is that, given certain assumptions and given enough information about the correlations among the events, a well defined causal system can be identified. This makes easier building models that explain children reasoning in a causal way. In fact, the theory-theory framework has been extensively used to develop causal probabilistic models (Gopnik et al., 2001, 2004; Gopnik & Schulz, 2004; Goodman et al., 2006; Sobel et al., 2004). In a similar way, BNs have been used in this thesis to model children's mental processes in accordance with the theory-theory framework. Previous work used BNs to model epistemic trust (Shafto et al., 2012). However, in that work the variables of knowledge, helpfulness, and beliefs were all properties of the informant, and the values associated to the conditional probability tables were estimated by best-fit on developmental experiments. Here we are interested in the mental processes of the children, and we want to know how a label provided by an informant is used to build a reputation model of that informant, and how the reputation is used to acquire new knowledge and learn the name of an object.

Just a few models of ToM have been implemented in robotic platforms. For instance, in Scassellati (2002) the authors defined guidelines for systems driven by a developmental model and tested a ToM module on the humanoid robot Cog. In Butterfield et al. (2009) a Markov random field was used in order to show how a probabilistic approach can model the ToM in a multi-agent setting. This work is particularly relevant because it reproduced two developmental experiment and it showed some interesting applications in a multi-robot scenario and in a gaze following task. In the first experiment a starting location and a target have been provided to a swarm of ground robots. Using a message passing of beliefs the robots were able to locate the target and communicate the information to other members of the team. In the work presented in Chapter 4 this Bayesian model of ToM has been expanded to include a measure of trust, showing how a bias in the conditional probability tables of dependent random variables could explain the differences observed between 3 and 4-year-olds in multiple developmental experiments.

2.5 Trust in Human-Robot Interaction

Since a robotic platform has been used for testing the model, it is necessary to provide a comparison with the human-robot interaction literature. However, before focusing our attention on HRI it is important to briefly summarize how trust is shaped in human-human interaction, and in particular how trust is built upon group interaction. This point is particularly relevant in the context of modern societies, where complex dynamics (e.g. shared goals, social roles, institutions, cultural differences, etc) extensively permeate everyday life. It is not easy to study how trust evolves in such a large setting, however hints from psychological studies can shed some light on these dynamics. There are four factors that contribute to the acquisition of cultural and group behaviour: joint attention, joint action, imitation and group assimilation (Carpenter, 2009). Of particular interest for the developmental component of this thesis are joint attention and joint action. Joint attention develops very early in infancy. Babies participate in interactions with their caregivers, using sights to indicate objects and capture their attention (Trevarthen, 1998). The same does not apply to joint action, that is considered as a process depending on more complex abilities. Even though infants may engage in primitive form of joint action using skills such as social referencing and intentions-in-action (Tollefsen, 2005), a mature ToM is generally considered a basic prerequisite to allow an understanding of others' goals, intentions, and common knowledge (Carpenter, 2009). These considerations relates joint action to the ToM, and therefore to the ability of discriminating reliable and unreliable agents as discussed in Section 2.4.

Other two factors mentioned by Carpenter (2009), imitation and group assimilation, are relevant from an HRI perspective. Those factors are tightly connected with the literature concerning cultural differences and social trust in HRI.

Different studies investigated the effects of communication styles and culture on people's accepting recommendations from robots. Rau et al. (2009) looked for differences between Chinese and German users who had to deal with a robot and decide to accept or not its recommendations. Chinese participants preferred an implicit communication style, and evaluated the robots as being trustworthy accepting the implicit recommendations. The German participants evaluated the robots as being less trustworthy and credible, and were less inclined to accept implicit recommendations. Moreover, cross-cultural studies have investigated the effects of cultural background on how people accept the choices and recommendation of a robot (Evers et al., 2008), confirming that people from different national cultures respond differently to robots. The effect of comunication style was studied by Bickmore & Cassell (2001). The authors described a model of social dialogue, and its implementation in an embodied conversation agent. They investigated how a variety of relational strategies used to establish trusting relationships between humans are used to communicate with artificial agents. The agent was displayed on a large projection screen, in front of which the user stood. The agent simultaneously processed the organization of conversation and its content using small talk. Within initial interactions between professionals and their clients, small talk is often used to build trust and solidarity. To evaluate whether social dialogue can actually build trust and solidarity with users using small talk, the authors conducted an empirical study in which subjects were interviewed by the agent about their housing needs. The results showed that there was more engaging when the agent used small talking.

Another line of research in HRI is more concerned in evaluating the properties that robots must have in order to be accepted by users. In this context one of the most important factors is the performance.

The effect of the robot performance on trust were studied by Freedy et al. (2007). In their work the authors described a decision-analytical measure of trust and the results of two experiments designed to examine trust in a tactical human-robot collaborative tasks. The operator controlled an unmanned ground vehicle (UGV) and the main mission was to ensure that a route through an area was safe by eliminating all enemies. The UGV could be set to one of three levels of competency. Operators were trained to recognize the distinct symptoms associated with competency failures. The results showed that the operator learned to compensate partially for low competence of the UGV, but there was little or no adjustment for medium competence. Moreover the amount of manual control was less if the operator first experienced a UGV of high or medium competence rather than one of low competence. Hancock et al. (2011) evaluated and quantified the effects of human, robot, and environmental factors on perceived trust in human-robot interaction. Meta-analytic methods were applied to the available literature on trust and HRI, collecting a total of 29 empirical studies. They examined three main categories of factors that may have a role in building trust between humans and robots: human-related factors (ability-based and characteristics), robot-related (performance-based and attributes-based), environmental (team collaboration and tasking). The results showed small effects for the human dimensions and also the environmental characteristics. The robot performance factors were more strongly associated with trust development and maintenance, whereas robot attributes had only a relatively small associated role.

2.6 Contribution

Given the literature review provided in the previous sections, it is now possible to better describe how the contribution of this thesis relates to it. As a starting point it is necessary to explicitly state how this work differs from other computational models.

First of all, this work significantly differentiate itself from the one based on the game-theoretical approach and in particular on the investment game paradigm, since we are interested in the particular case where no external rewards are given. In this case only internal factors and prior beliefs influence trust. Although these internal factors may have been influenced by external ones in previous phases, at learning time they do not receive any more the external feedback, leaving to the system the responsibility to autonomously decide which information is relevant and which information must be discarded. Previous knowledge, belief, and a record of past interactions are still available, and they can be used to drive learning.

Secondly, it may seem natural to include the model in the cognitive branch as defined by the conceptual taxonomy presented in Sabater & Sierra (2005). However, this work takes into account developmental factors and biological hints. This is a key point and the main difference from other architectures.

Despite the rich literature on e-trust and investment games discussed in Section 2.2 and 2.4 there are just a few developmental models of trust. For instance, a computational model of epistemic trust was introduced in Shafto et al. (2012). The authors built a probabilistic modelling framework where learning is based on data and formalised as a Bayesian inference network. Although all these models are valuable from a representational point of view, they do not take into account a wider picture, and they fail in recognising valuable aspects such as the biological substrate and the embodiment.

Trust has been examined in HRI, however a perspective shift is introduced here. Most of the work in human-robot interaction is based on the human point of view. Using again the definition of trust given in Falcone & Castelfranchi (2001) this thesis mainly focuses on the case where x is the human and y the machine. Trust is important in classical HRI context because it affects the willingness of people to accept robot-produced information. See Hancock et al. (2011) for an extensive review. However, here we are interested in the other side of the coin. We want to build a system that is able to discriminate between reliable human informants taking into account their past performance. In this case x is the robot and y the human.

Given all these considerations, the current thesis provides a valuable contribution to the computational modelling of trust expanding the literature with an original cognitive architecture. In the following chapter the methodology adopted to build up the model is introduced and described.

Chapter 3

Methods and Materials

3.1 Developmental Robotics

3.1.1 Definition

Developmental robotics is the study of the development of cognitive processes with the aim to create methods and algorithms for the autonomous acquisition of high-level cognitive skills. A formal definition has been given by Cangelosi & Schlesinger (2015):

"Developmental robotics is the interdisciplinary approach to the autonomous design of behavioral and cognitive capabilities in artificial agents (robots) that takes direct inspiration from the developmental principles and mechanisms observed in the natural cognitive systems."

The interdisciplinary nature of developmental robotics is based on the tight connection between empirical sciences such as psychology, sociology, linguistics, neuroscience, and computational disciplines such as robotics, artificial intelligence and machine learning.

In this thesis the developmental robotics paradigm has been used in order to study the development of trust in 3 and 4 years old children. Following an interdisciplinary approach multiple lines of research have been taken into account. For instance, psychological observations played a vital role in this investigation and they were widely used to set-up the architecture. Moreover, neurobiological studies gave a solid theoretical foundation based on the observation of the neural correlates of trust.

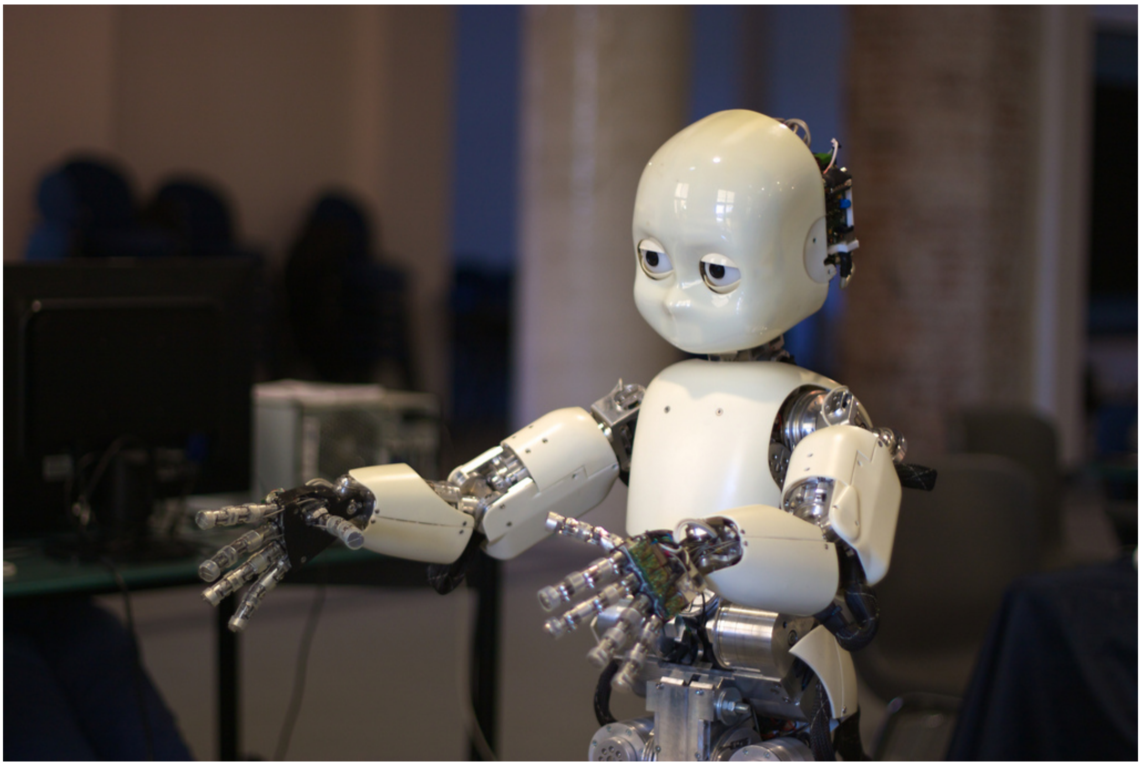


Figure 3.1: The iCub humanoid robot. The iCub robot has been widely used by the research community, and it is considered one of the best robotics platform for studying cognitive phenomena in a bio-plausible way.

3.1.2 The iCub Humanoid Robot

The central idea in the developmental robotics paradigm is to embody an algorithm in a robot. The embodiment allows the researcher to take into account additional factors that may play an important role in biological systems. In this work, an open-source robotic platform widely adopted by the research community has been used, namely, the iCub humanoid robot (Metta et al., 2008) (Figure 3.1). The iCub has 53 degrees of freedom, the head and the eyes can move independently in a bio-plausible way and can track moving targets in the environment.

The iCub software suite includes an open source simulator (Tikhanoff et al., 2008) that can be used to test algorithms in a safe controlled environment. These algorithms can then be adapted to the physical robotic platform. The simulator environment has a screen were a video or a live webcam stream can be projected. Moreover, it allows simple geometrical objects such as spheres, cubes and pyramids to be created and moved (Figure 3.2).

The low-level motor activity of the robot is controlled by an internal CPU, whereas additional programmes can be deployed on an external computer or a cluster. The

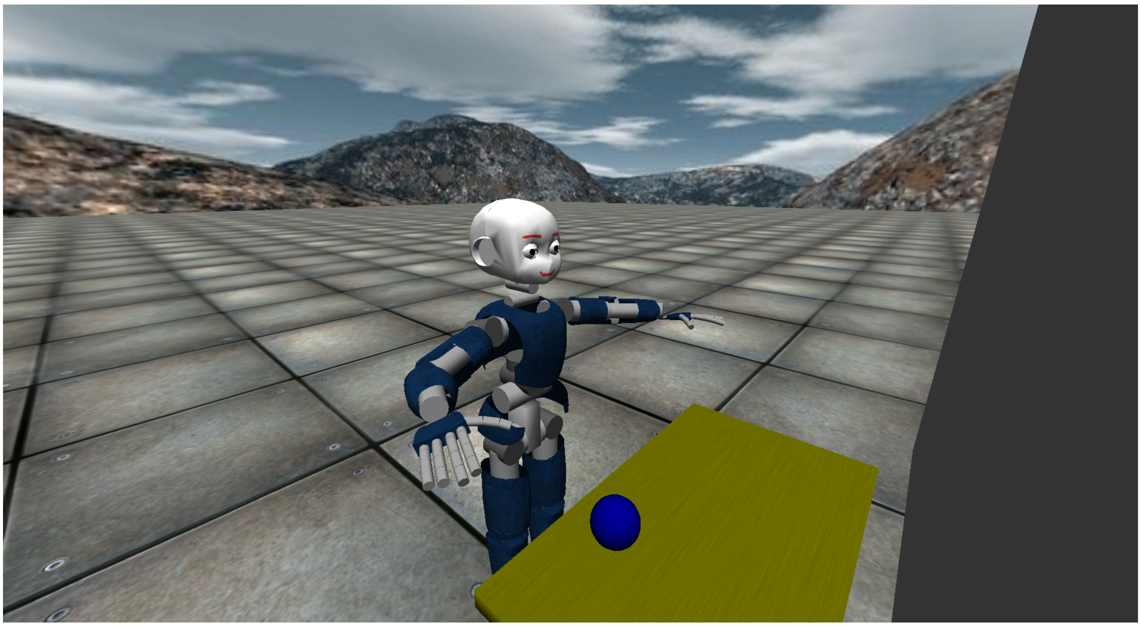


Figure 3.2: The iCub simulator. The simulator allows testing the algorithms before deploying them in the real platform.

communication between multiple nodes is made easy due to the use of a software package called Yet Another Robot Platform (YARP) (Metta et al., 2006), an open source middleware that allows different nodes to communicate in a peer-to-peer way. A natural interaction between the user and the robot is extremely important. In this sense the use of a keyboard has been discarded in favour of a more natural voice-based interface. The vocal commands to the robot were interpreted using Pocketsphinx (Huggins-Daines et al., 2006) an open-source real time language recognition system.

3.1.3 The Epigenetic Robotic Architecture

The developmental robotics approach requires another component in order to be effective. This component is an architectural framework flexible enough to facilitate the deployment of higher level algorithms. One of the best available options is the Epigenetic Robotics Architecture (ERA) of Morse et al. (2010). The ERA is a simple but extremely powerful hierarchical framework, consistent with constructivist sensorimotor theories, that can be easily extended and scaled up in different scenarios. ERA is formed by multiple Self-Organising Maps (SOMs). Each SOM receives a subset of the inputs and organises those inputs in clusters. For instance, a SOM can be trained on visual stimuli and then used to classify new features. The SOM reduces the complexity of a large state space projecting the input features in a smaller sub-space. The SOMs in the ERA unit are connected through

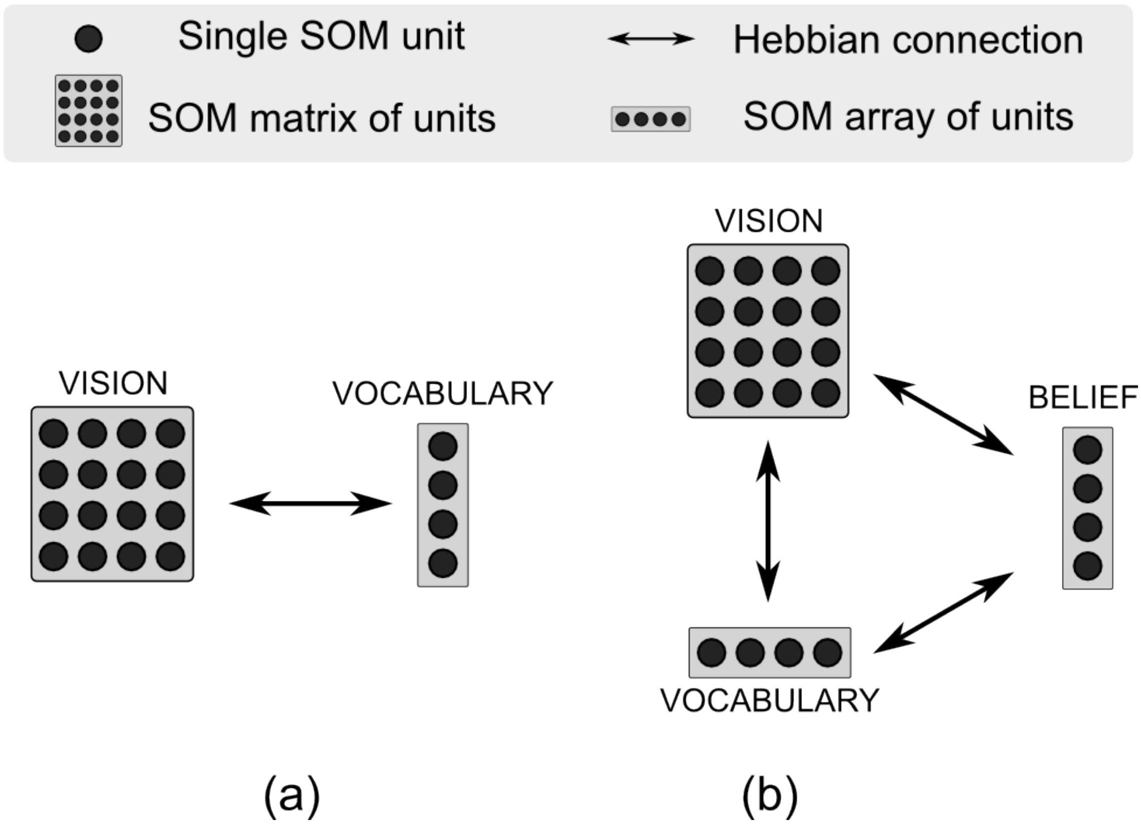


Figure 3.3: The ERA modules can be arranged in different ways in order to handle complex problems. The SOM can be represented as a vector or a matrix, made of single units (black dots). Multiple SOMs can be connected through Hebbian connections (bidirectional arrows) creating an ERA module. (a) Using a vision and a vocabulary SOM it is possible to have an ERA module representing a value function approximator. (b) Connecting a vision module and a vocabulary SOM to a belief array, it is possible to have an ERA module representing a state-belief function approximator.

Hebbian synapses.

The flexibility of ERA permits to arrange the components of a unit in many possible ways. For instance, in our case we are dealing with a state space of image-label pairs that we can represent as a unit composed of a vision and a vocabulary module as illustrated in Figure 3.3 (a). At the same time we introduce a belief module and represent a state-belief function as shown in Figure 3.3 (b).

In the following sub-sections the main phases involved in the use of ERA are briefly described: stabilisation, belief estimation, and learning.

Stabilisation

The SOMs in the ERA unit must be stabilised presenting randomly generated inputs which cover a wide range of values (Morse et al., 2010). The stabilisation ensures that the resulting maps fully cover the range of possible inputs. Here we briefly describe this stabilisation process. Given a D dimensional space, we define the input vector as $\mathbf{x} = \{x_i : i = 1,...,D\}$. Every time an input vector $\mathbf{x}$ is passed to the SOM a discriminant function is called. We define our discriminant function to be the Euclidean distance between the input vector $\mathbf{x}$ and the weight vector $\boldsymbol{w}$ for each internal unit j:

$$d(\mathbf{x}, \boldsymbol{w}_j) = \sqrt{\sum_{i=1}^{D} (x_i - w_{ji})^2} \tag{3.1}$$

The unit j whose weight vector $\boldsymbol{w}$ is most similar to the input $\mathbf{x}$ is called Best Matching Unit (BMU). The vector $\boldsymbol{w}$ of the BMU is updated at the next time step $t+1$ using the following rule:

$$\boldsymbol{w}(t+1) = \boldsymbol{w}(t) + \eta(t)(\mathbf{x}(t) - \boldsymbol{w}(t)) \tag{3.2}$$

The value η is called learning rate and it is a t dependent value which is updated at each iteration based on the equation

$$\eta(t) = \eta_0 \exp\left(-\frac{t}{\lambda}\right) \tag{3.3}$$

The value of η decreases exponentially over time in order to reduce the update magnitude on the vector $\boldsymbol{w}$.

Learning

The learning phase in ERA consists of strengthening or weakening the connections between two BMUs. Here, the connections among SOMs are represented as a parameter vector $\boldsymbol{\theta}$. When two BMUs i and j are active at the same time the connection between them is modified using positive Hebbian learning:

$$\boldsymbol{\theta}_{i,j}(t+1) = \boldsymbol{\theta}_{i,j}(t) + \beta x_i x_j \tag{3.4}$$

Where β is a learning rate value in the range [0,1], x_i is the activity in the winning node in one map and x_j is the activity of the winning node in the second map. Often the connection between units that are not active at the same time is weaken using a negative factor (negative Hebbian learning).

Estimation

Once the SOMs have been stabilised and the learning phase has adjusted the connections, it is possible to do a forward pass and estimate the values of the output units. For instance given a specific visual input and a label it is possible to find the BMU in the belief module. This is obtained through the weighted sum of the BMUs in the visual and vocabulary module. The forward pass can be applied in any other direction, from the vocabulary to the vision unit and viceversa.

3.2 Bayesian Networks

3.2.1 Definition

BNs are probabilistic graphical models that represent the conditional dependencies between a set of random variables. They are represented as a directed graph where the nodes are random variables and the edges are causal relationships.

Our analysis will be limited to the case of finite and countable values. In this case the random variables are called discrete, and the associated probability distribution is called probability mass function. We used capital letters to indicate random variables and small letters to indicate the possible state or event of the variables. For example, the random variable X is discrete and it can assume four states $\{a, b, c, d\}$. When the outcome of a random variable depends on the outcome of another one, we have a conditional dependence. If the random variable Y depends on X we can calculate the probability of Y given a particular outcome of X which is denoted by $P(Y|X)$. When the two random variables are conditionally independent $P(Y|X) = P(Y)$ are also indicated with $Y \perp X$.

In a BN the conditional dependence incorporates a principle of causality. If X causes Y we have a BN with a direct edge from X to Y as represented in Figure 3.4.

The probabilities associated with the conditional dependence between X and Y in a BN are represented with a conditional distribution, and in the case of discrete random variables can be described with a table, called conditional probability table. In Figure 3.4 the

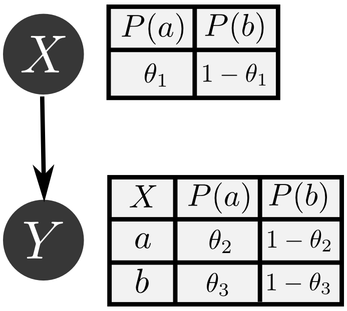


Figure 3.4: An example of Bayesian Network (BN). This BN has two nodes, the first node X is the parent and the node Y is the child. Each node has an associated probability distribution represented with a table in the discrete case. The table of the parent node only has two parameters, whereas the table of the child node has four parameters.

conditional probability tables associated with X and Y are reported near each node. Each row of a conditional probability table sums to one, and contains the conditional probability of each value for a conditional case. A conditional case is a combination of values for the parent nodes. The first column of the table associated to Y in Figure 3.4 represents the conditional values a and b for the parent X.

3.2.2 Inference

The main advantage of using BN is the possibility to infer the posterior probability of the nodes given some observations. In particular we used inference for estimating the belief given an action, and vice versa. For example, if the belief is $X = b$ we can calculate the posterior probability of the state $Y = a$ given $X = b$. In the simplified case of Figure 3.4 the posterior value is contained in the probability table associated with the node Y. The inference process can be applied also from the effect to the cause. For example, if we observe the action $Y = b$ we can find the posterior probability of $X = a$ by means of the Bayes' Theorem:

$$P(X = a|Y = b) = \frac{P(Y = b|X = a)P(X = a)}{P(Y = b)} \tag{3.5}$$

3.2.3 Learning the Network Parameters

BNs are commonly used for creating expert systems, software that permits emulation of decision-making ability at the level of human experts on a tightly delineated problem (Spiegelhalter et al., 1993). In this context the network is built by hand with the help of domain experts. When the amount of knowledge required is huge it is possible to use data instead of the experts for setting the network parameters, an approach known in literature as parameter estimation or parameter learning (Koeller & Friedman, 2009).

Here, Θ is the set of all possible parameters $\boldsymbol{\theta}_i$, where $\boldsymbol{\theta}$ is the vector containing all the parameters of the estimator. The goal is to find the subset of parameters that better explains the observed set of data $\mathcal{D} = \{d_1, ..., d_N\}$. This goal can be achieved via Maximum Likelihood Estimation (MLE). As showed in Equation 3.5, the main factors in Bayesian learning are the hypothesis prior $P(\boldsymbol{\theta})$ and the likelihood of the data given the configuration of the parameters $P(\mathcal{D}|\boldsymbol{\theta})$. However, assuming a uniform prior of the parameters it is possible to use a frequentist inference approach and exploit the information provided by the likelihood for estimating the parameters of the model, given the observations. This is MLE in a nutshell.

Given the likelihood function $\mathcal{L}(\boldsymbol{\theta}, \mathcal{D})$ we want to find a set of parameters that maximise this function. It is important to recall that the likelihood function is not a probability density function. The likelihood is estimated under the assumption that the observations are *indipendent and identically distributed (i.i.d)* so that:

$$P(\mathcal{D}, \boldsymbol{\theta}) = \prod_{i=1}^{N} P(d_i|\boldsymbol{\theta}) \tag{3.6}$$

This equation has a flaw, being the product of multiple values that may lead to round-off errors. To change this product into a sum the natural logarithm of the likelihood, also known as the *log-likelihood*, is used. If $\boldsymbol{\theta}$ belongs to a Bernoulli estimator then it is no more a vector but a single parameter that can be expressed via a non-bold notation θ. This parameter represents the probability of a as $P(a) = \theta$, and the probability of b as $P(b) = 1 - \theta$. The resulting single variable log-likelihood can be expressed in the following form:

$$\begin{aligned} \ln \mathcal{L}(\theta, \mathcal{D}) &= \ln P(\mathcal{D}, \theta) \\ &= \sum_{i=1}^{N} \ln P(d_i|\theta) \\ &= \ln \left[\theta^{N_a}(1-\theta)^{N_b}\right] \end{aligned} \tag{3.7}$$

To keep the notation uncluttered, in the last passage the iterative sum over d_i has been replaced with the total number of times the state a has been observed N_a, and with the total number of time the state b has been observed N_b.

To find the maximum likelihood it is necessary to differentiate the log-likelihood with respect to the parameter θ and set this derivative to zero. The derivative can be estimated as follows:

$$\begin{aligned}\frac{d \ln \mathcal{L}(\theta, \mathcal{D})}{d\theta} &= \frac{d \ln \left[\theta^{N_a}(1-\theta)^{N_b}\right]}{d\theta} \\ &= \frac{d\, N_a \ln \theta + N_b \ln (1-\theta)}{d\theta} \\ &= \frac{N_a}{\theta} - \frac{N_b}{1-\theta}\end{aligned} \tag{3.8}$$

Setting the derivative to zero and isolating the parameter, we get:

$$\begin{aligned}\frac{N_a}{\theta} - \frac{N_b}{1-\theta} &= 0 \\ \theta(N_a + N_b) &= N_a \\ \theta &= \frac{N_a}{N_a + N_b}\end{aligned} \tag{3.9}$$

In conclusion, to obtain the parameter of a Bernoulli distribution is just necessary to divide N_a (the total number of times the state a has been observed) by $N = N_a + N_b$ (the total number of observations). This is coherent with the frequentist approach and a uniform prior.

3.2.4 Issues related to the use of Bayesian networks

Although BNs provide a strong causal framework for modelling cognitive functions, they do not come without problems. In particular, there are two main issues related to the use of BNs that will be discussed here shortly. The first problem is the space and time complexity required in order to estimate the posterior probability. To understand this point it is necessary to consider something more complex than the toy network represented in Figure 3.4. We can suppose that there are N random variables X_i as roots and that they are connected to a single random variable Y. Each variable X_i, and the variable Y, can assume two states a and b. In this new case, the conditional table in Y must take into account all the possible configurations of the N root nodes, meaning that it requires a table of size 2^N. Even worse, if the number of possible states for the root nodes is $M \gg 2$, then the size of the table is M^N. Having a large table creates problems in both storage and access, and since there is an exponential factor involved, the time and space required to manipulate the table can easily become intractable.

The second problem is concerned with the structure of the network itself. Computing the posterior distributions in complex networks presents some difficulties. When the network has only one root and each node has only one parent, the network is called a tree. In this case the Pearl's message passing algorithm (Pearl, 1982) computes the exact posterior distribution for each node. In case of multiple roots, the network is called a poly-tree and another version of the Pearl's message passing algorithm must be used (Kim & Pearl, 1983). Networks with complex connections that cannot be classified as tree or poly-tree are called multi-connected. Exact inference is almost impossible to obtain with these networks, and approximation methods are generally used.

The focus of the thesis is not on these problems since they are active research areas in machine learning and can be considered here as collateral issues being weakly connected with the main goal of the research. The BNs used in this work are poly-trees and exact inference methods, such as the Pearl's message passing algorithm (Pearl, 1982), can be used without problems. In complex settings this can be an oversimplification. In fact, there are many domains (e.g. acquisition of syntactic, mathematical, or musical knowledge) that do not seem to involve the recovery of causal structure, and causal learning might not apply (Gopnik et al., 2001). In the particular case considered here, acyclic graphs are not simply used to mitigate technical issues, they are related to unconscious activity used to draw conclusions about patterns of contingency (Gopnik et al., 2004). This point is consistent with previous work relating acyclic BNs to children's reasoning (Gopnik et al., 2001, 2004; Gopnik & Schulz, 2004; Sobel et al., 2004), that has showed how BNs can provide an accurate description of experimental data.

3.3 Artificial Neural Networks and Deep Learning

3.3.1 Definition

The research on artificial neural networks started in the 1940s when McCulloch & Pitts (1943) described a logical calculus of the nervous system based on propositional logic. Given the technical limitations of the time the work remained a theoretical contribution without the possibility of computational applications. Only 15 years later the work of Rosenblatt (1958) allowed the community to understand the intrinsic power of neural networks. The Rosenblat's perceptron was a single layer neural network, with a threshold activation function and an update rule based on Hebbian learning (Hebb et al., 1949). In the following years there were different proposals for improving the perceptron, one of the most remarkable was ADALINE (Widrow, 1960), a neural network based on a new update rule called the *Delta Rule* a particular case of error backpropagation. Although the perceptron was able to discriminate between different unseen inputs, its capabilities were questioned when the work of Minsky & Papert (1969) was published a few years later. Minsky & Papert (1969) mathemathically proved that the perceptron was a linear classifier, meaning that it was only able to discriminate linearly separable input spaces. This flaw was an important limitation and rapidly discouraged the research community to pursue the approach. The resurgence of interest in neural networks happened in the 1980s, when a new learning rule called *backpropagation* (developed in the 1960s and 1970s) was applied to the training of multi-layer perceptrons. In particular, was an article of 1986 that contributed to popularise the use of backpropagation in neural networks (Rumelhart et al., 1986b).

In its standard form a neural network is made of different units (often called neurons) associated in layers, and connected via weights (often called parameters). The numerical input vector $\mathbf{x}$ is multiplied with the weight matrix $\boldsymbol{W}$, and a bias vector $\mathbf{b}$ is added. The result $\boldsymbol{z}$ is passed through a differentiable non-linear activation function $h(\boldsymbol{z})$ leading to the final activation of the layer. This process is iteratively repeated with the output of a layer given as input to the next. When the output of the final layer is computed, the output $\mathbf{o}$ is compared with the desired target $\mathbf{t}$ and passed through a differentiable error function $E(\mathbf{t} - \mathbf{o})$. It is on the error function that the backpropagation is performed. The gradients of the weight matrices in each layer are computed via chain rule and the weights adjusted in the direction of the negative gradient vector, so to minimise the error.

3.3.2 Convolutional Neural Networks

The first use of CNNs for image classification dates back to the early nineties. One of the main contribution was the one of LeCun et al. (1998) that used a CNN to recognise hand-written digits. In recent years deep convolutional networks have showed their strength in numerous pattern recognition contests. Some remarkable achievements have been recently obtained in object detection (Krizhevsky et al., 2012), facial expression recognition (Lopes et al., 2016) and scene classification (Nogueira et al., 2016). This technology is increasingly used in commercial applications such as content filtering in social networks, recommendation systems in e-commerce websites or image classifiers in web-search engines. The deep learning revolution has been driven by the diffusion of cheap Graphical Processing Units (GPUs), originally used for video games. The use of GPUs speeds up matrix and vector multiplications, which are the core operations in neural network training. Because the general introduction of CNNs is well established by now, we will not extend this section any further. Instead we refer the reader to the review article of Schmidhuber (2015).

A CNN is defined as an ensemble of many single units grouped in three-dimensional layers. The units inside a layer are connected to a small region of the layer before it with connections called kernels (or filters, weights, parameters). The input to a CNN is a matrix X of dimension $m \times m \times r$, where m is the height and width of the matrix and r is the number of channels. The convolutional layer has k kernels of size $n \times n \times q$, where $n < m$ and $q \leqslant r$. The convolution multiplies each element of X with its local neighbours, weighted by the kernel W, generating k feature maps of size $m - n + 1$. The convolutional layer is often followed by a mean or max pooling layer which permits subsampling the maps over a $p \times p$ local region, with $2 \leqslant p \leqslant 5$. A graphical representation of a CNN is reported in Figure 3.5.

During the training phase the kernels are adjusted following a well-known algorithm called backpropagation (Rumelhart et al., 1986a). The backpropagation minimises a loss (or error) function $J(w)$ with an iterative process of gradient descent that updates w at

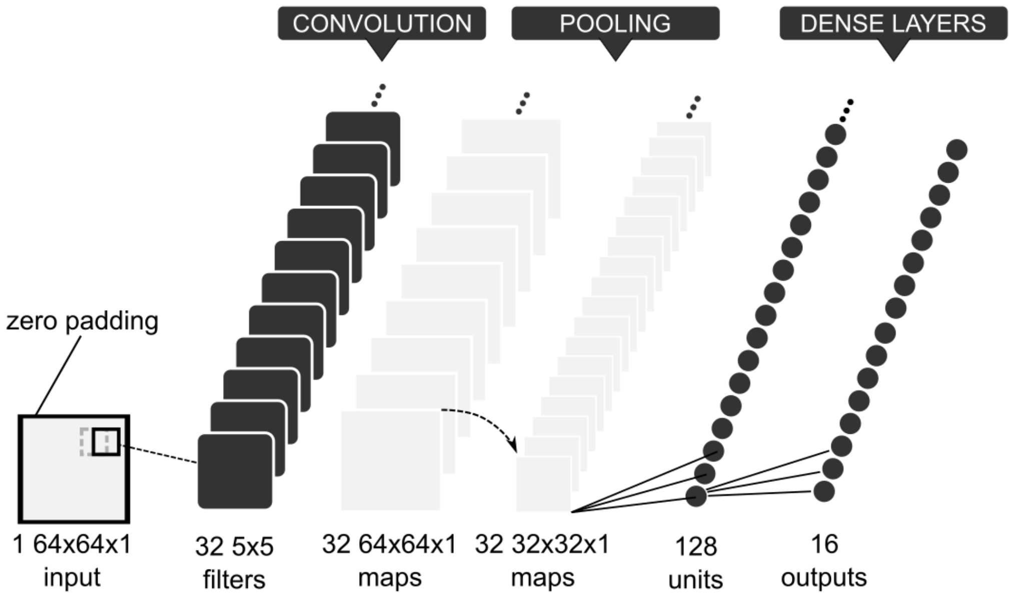


Figure 3.5: An example of Convolutional Neural Network. A greyscale image of size 64×64 pixels is passed as input to the network and padded with two columns/rows of zeros on the four borders. The first layer performs a convolution with 32 kernels of size 5×5 (stride 1) generating 32 features map of size 64×64. A pooling operation (stride 2) is then performed on the feature maps resulting in a set of 32 features of size 32×32. Dense connections are used to associate the feature maps to standard units (128). Finally, the output layer (16 units) is connected to the previous one by dense connections.

$t+1$ using the gradient information at t, as expressed in the following equation:

$$w_{t+1} = w_t - \alpha \nabla E|J(w_t)| + \mu v_t \tag{3.10}$$

The expectation is approximated with the cost and gradient over the full training set. The value α is called learning rate and corresponds to the step taken by the algorithm in the direction of the gradient. The value $\mu \in [0,1]$ is called momentum and is a technique for accumulating a velocity vector v in the direction of persistent reduction of the loss function. A variation of the standard gradient descent is called Stochastic Gradient Descent (SGD). The SGD computes the gradient using a few training examples or mini-batches instead of the whole training set. Using the stochastic approach the variance is reduced leading to a more stable convergence.

There are different kinds of loss functions. In this thesis the sum of squares of the differences between the target value y and the estimated value $\hat{y}$ has been used:

$$J(w) = \sum_{n=1}^{N} (y_n - \hat{y}_n)^2 + \lambda \sum_{l=1}^{L} w_l^2 \tag{3.11}$$

The additive factor λ is an L2 regularisation term, used in each hidden layer l to prevent a very large growth of the parameters during the minimisation process.

3.3.3 Adaptive Gradient Methods

Neural networks are not off-the-shelf algorithms, and generally the research of the best hyperparameters can be extremely time consuming. One of the most important parameters is the learning rate. When the learning rate is too high the optimisation can diverge, on the contrary if it is too low the optimisation can be slow. To solve these problems some adaptive gradient methods have been proposed recently. Adaptive gradient methods use first order information to approximate second order information and then find an optimal step size.

One of the best adaptive methods introduced recently is Adagrad (Duchi et al., 2011). This optimiser incorporates information about the features to control the gradient step. The procedure associates a low learning rate with frequently occurring features and high learning rate with infrequent features. Therefore, the adaptation facilitates identifying the most predictive features and it is well suited for sparse data. The authors tested the algorithm on different image and text databases, showing how it outperforms the non-adaptive counterpart. The updating rule for the weights w following the Adagrad algorithm can be expressed with the following equation:

$$w_{t+1} = w_t - \frac{\alpha}{\sqrt{G_t + \epsilon}} \odot g_t \tag{3.12}$$

The matrix G is a diagonal matrix where each entry in the main diagonal is the sum of the squares of the previous gradients up to time t. The value ϵ is a small value used to avoid division by zero. The symbol $\odot$ represent a matrix-vector multiplication. The main problem with Adagrad is the accumulation of squared gradients in G that leads to an infinitesimally small learning rate α and then to a loss in knowledge accumulation.

To solve the problem related with Adagrad an extension called Adadelta (Zeiler, 2012) has been proposed. Adadelta does not accumulate all the past gradients but it constrains the window to a fixed interval. The denominator of the Equation 3.12 is then replaced by a moving average $E[g^2]$ which represents all the past squared gradients. The advantage of this solution is that the moving average depends only on the previous average and the current gradient, as follows:

$$w_{t+1} = w_t - \frac{\alpha}{\sqrt{E[g^2]_t + \epsilon}} g_t \tag{3.13}$$

As experimentally shown in Zeiler (2012) Adadelta is particularly robust and it guarantees convergence with different learning rate values. Another effective method to solve the Adagrad issue is an unpublished algorithm called RMSProp (Tieleman & Hinton, 2012). RMSProp has an updating rule which is similar to Equation 3.13 with the only difference being that it introduces a decaying value γ which specifies how long the old gradients in $E[g^2]$ are kept. The value $E[g^2]$ at time t is then updated in this way:

$$E[g^2]_t = \gamma E[g^2]_{t-1} + g_t^2 \tag{3.14}$$

The authors suggest some default values for the learning rate and the decaying value which should be closer to $\alpha = 0.001$ and $\gamma = 0.9$. In our experiments we used this configuration.

Finally we want to introduce another method called Adaptive Moment Estimation (Adam) (Kingma & Ba, 2014). Like Adadelta and RMSProp, Adam stores a decaying average of past gradients and squared gradients. The two decaying averages are then used to estimate m_1 and m_2 which are the first and second moment (mean and variance) of the gradient. The updating rule for Adam can be expressed as follows:

$$w_{t+1} = w_t - \frac{\alpha}{\sqrt{m_2} + \epsilon} m_1 \tag{3.15}$$

The two moments m_1 and m_2 are taken at time t and before the weights update they are corrected to limit a bias toward zero during the first steps. The moments are regulated by two decaying factors β_1 and β_2. The authors suggest to initialise these parameters to standard values $\beta_1 = 0.9$ and $\beta_2 = 0.999$. We used these values in our experiments.

3.4 Reinforcement Learning

3.4.1 Definition

Reinforcement learning is learning how to map situations to actions in order to maximise a numerical reward signal (Sutton & Barto, 1998). The learner must discover the optimal set of actions that maximises the long term reward. This goal is different from *supervised learning*, where a labeled training set is provided, and the objective is to generalise to unseen data. Reinforcement learning is also different from *unsupervised learning*, where the objective is to find regularities in a set of unlabeled data, and group those sets in different clusters. Since the main objective in reinforcement learning is to maximise a reward signal, we cannot consider it as part of the aforementioned classes. We can instead think of it as a separate class made of the following components:

1. *world*: the main environment, it is defined by a set of rules that can be accessible (model-based) or not accessible (model-free) from the agent.
2. *state*: in case of discrete worlds the state is the configuration at a given time step.
3. *action*: this is produced by the policy and allows the agent to move in the world changing the current state (transition).
4. *episode*: an episode is the series of transitions that leads from a starting state to a final state.
5. *policy*: this is the mapping from states to actions represented by a function or look-up table that the agent can access.
6. *reward*: numerical feedback received from the world at every transition.

The typical reinforcement learning cycle starts from an initial state s_t. This is a configuration of the world (including the agent). At each time step the agent performs an action a accessing its internal policy $\pi(s)$, and the transition leads to the next state s_{t+1}. A numerical reward is given at each step based on the initial state, the action, and the next state through the function $R(s_t, a, s_{t+1})$. The reinforcement learning cycle is graphically represented in Figure 3.6.

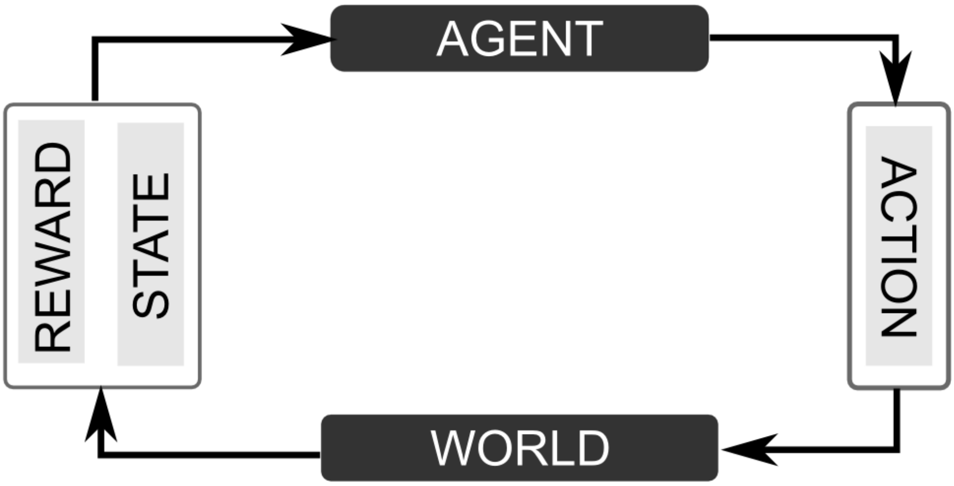


Figure 3.6: The reinforcement learning cycle represented by the agent-world interaction, through action, reward, and state.

3.4.2 Intrinsically Motivated Reinforcement Learning

The concept of intrinsic motivation was initially proposed and developed within psychology literature to overcome the difficulties of the behaviourist theory in explaining why humans and animals spontaneously engage in puzzles and games (Baldassarre, 2011). An intrinsically motivated behaviour is engaged by an internal motivation that is inherently enjoyable, meaning that the agent is not looking for any external rewarding outcome. The formalisation of intrinsic reinforcement learning is based on the concept of an intrinsic environment where the internal decisions of the agent (beliefs) are redirected before the actions are actually performed in the external world (Singh et al., 2004). Also the rewards and states received by the agent are decoupled from the external world through the intrinsic filter. A graphical representation is presented in Figure 3.7.

The intrinsic environment is often thought of as a *critic* whose feedback is forwarded to the agent policy controller. As pointed out by Sutton & Barto (1998), one should be able to identify the critic through processes within the brain of the animal that monitor both internal and external states. The role of the critic can be better understood with an example. The physiological reward associated with taking drugs is high in the short term, however, an efficient critic will discourage this because it has detrimental consequences in the long term. In this case the critic evaluates the external state (drug) and the reward associated to it (high), and applies a cost (very high) that significantly penalises the next state (taking drug).

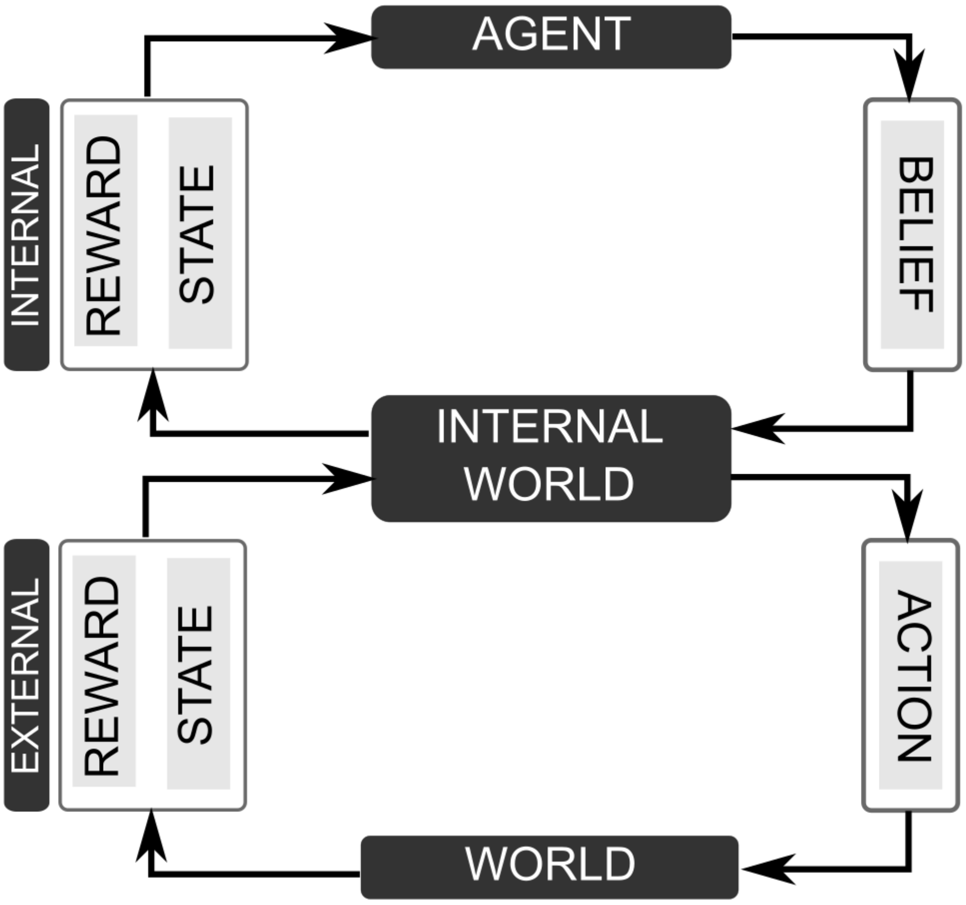

AGENT
INTERNAL
REWARD
STATE
BELIEF
INTERNAL WORLD
EXTERNAL
REWARD
STATE
ACTION
WORLD

Figure 3.7: The intrinsic reinforcement learning cycle represented by the agent interaction with the external world passing through an internal environment.

We can also think of behaviour that is based on intrinsic motivation only. This has often been called *curiosity driven exploration* and it has been the subject of extensive research in the artificial intelligence community (Schmidhuber, 1991, 2006; Ngo et al., 2013; Frank et al., 2014). A curiosity driven exploration can enable continual learning in the absence of external rewards. This kind of learning has been observed in children, adults, and animals (Piaget, 1955; Berlyne, 1966), and can manifest so strongly that urgent needs are overridden. For instance, an hungry rat may spend time exploring a new feature in the environment instead of eating (Majorana, 1950). Intrinsic curiosity is often triggered by a collection of stimuli that is perceived at different times or all at once. According to Berlyne (1966) the properties of the stimuli that elicit curiosity are: incongruity, complexity, variability, puzzlingness, novelty, and surprisingness. An important brain region involved in novelty detection is the hippocampus which plays a key role in responding to new events and activating projection to different cortical areas (Kumaran & Maguire, 2007). When a novel stimulus is perceived the hippocampus activate a recurrent dopaminergic pathway between itself and the ventral tegmental area, that reinforces the memory trace.

In this thesis intrinsic motivation has an important role since trust-based learning has been defined as a process based on internal representations of the informants that are not linked to any external reward.

Chapter 4

A Developmental Bayesian Model of Trust

4.1 Introduction

The model presented in this chapter is a probabilistic causal model which has strong connections with the theory-theory framework discussed in Chapter 2. The proposed method is based on BNs. These are causal probabilistic models that can describe the mental processes of the child (Gopnik et al., 2001). This statement is based on consistent research which is part of the theory-theory approach to the ToM (Gopnik & Meltzoff, 1997). The theory-theory uses causal probabilistic models to formalise the child's reasoning in a mathematical way. Because the model must be flexible and applicable to different conditions it has been necessary to keep it constrained to a tested developmental framework. This choice makes the model more robust and links it to a long series of high quality experiments (Gopnik et al., 2001, 2004; Gopnik & Schulz, 2004; Goodman et al., 2006; Sobel et al., 2004).

It is worth pointing out that a previous model of ToM based on a probabilistic graphical model has been developed by Butterfield et al. (2009) and it has been briefly discussed in Chapter 2. In Butterfield et al. (2009) a Random Markov Field (RMF) was used in order to show how a probabilistic approach can model some aspects of the ToM. To show the potential of the Bayesian approach we applied a modified version of Butterfield et al. (2009) to trust estimation. To achieve this goal it was necessary to reshape the model of Butterfield et al. (2009). RMFs are undirected graphs, meaning that they do not incorporate the cause-effect principle, since instead of using conditional probabilities, they use joint probabilities to define relations between nodes. The major issue we had to overcome in order to use the approach discussed in Butterfield et al. (2009) is due to an assumption made by the authors. The model is explicitly based on the hypothesis that agents do not deceive each other. As a consequence there is no distinction between the inferred mental state and the actual mental state. Such an assumption is reasonable when the agents act cooperatively, but it is not plausible when it is necessary to estimate trust. The concept of trust includes the possibility that the agents deceive and that their mental states are different from their actions. The authors assert that people are unable to access a human's actual mental state. This is only partially true, because given a record of past actions we can infer others' mental states. For example, if a person was unreliable in past situations we can use this record to infer her behaviour in a similar situation. In the proposed model the mental state of the others has been taken into account and an inference step for estimating these mental states has been introduced. This inference step was ignored in Butterfield et al. (2009) and was in fact impossible due to the limitations in the model design.

Another novelty introduced in this work is the use of the MLE for adjusting the network parameters. The MLE can be compared to a familiarisation phase, during which the child acquires information about the informants. As showed in the next section, this method can be used to collect data and it can use this data to update the BN parameters. The theory behind the model has been introduced in Chapter 3. The reader is invited to review that chapter in order to better understand the following sections.

4.2 Model Description

In the Bayesian model introduced in this section X represents beliefs and Y represents actions. Furthermore X_C denotes the belief associated to the child where X_R and X_U denote the belief associated to the reliable and unreliable informant. The same subscripts have been used for naming the variables Y. All the random variables are discrete, in particular boolean random variables that assumed two states a and b are used. The states can be considered possible labels for an object, like in Koenig & Harris (2005), or locations, like in Vanderbilt et al. (2011).

The core concept is the distinction between belief and actions whereas X represents beliefs and Y represents actions. An edge from X to Y indicates that the action Y is a direct consequence of her belief X like in Figure 4.1. This differentiation is very powerful because it can describe reliable and unreliable informants. When an informant is deceiving there is a difference between belief and action, and therefore the posterior distribution of X is significantly different from the one of Y.

To estimate the network parameters the MLE has been used. The MLE can take a dataset to adjust the BN parameters. In our case the dataset represents statistical information collected by the child during the interaction with the informants. The implicit hypothesis in this approach is that children can collect statistical information for tracking the reliability of others, and that this process is similar to how MLE sets the parameters of a BN. This is in line with recent research (Kushnir et al., 2010) that showed how young children can use statistical information, particularly a violation of random sampling, to infer the preferences of an adult for certain type of toys. This result was observed also in 20-month-old infants, confirming that it is an early developmental mechanism.

The MLE has been discussed in Section 3.2.3, here the insight behind its adoption is explained through a practical situation: an interaction between a child and a caregiver. In this example the child is observing a caregiver choosing one of two boxes a and b. The child wants to predict the behaviour of the caregiver in a future trial. Observing the caregiver for a certain period of time the child can collect a dataset, specifically a set of outcomes where each element can be a or b. We denote N_a and N_b as the number of times the caregiver chooses a and b. To describe this toy example in the BN language we need only one discrete node Y representing the action of the caregiver. Defining a parameter θ we can formalise the probabilities that the caregiver chooses the two boxes this way:

$$P_Y(a) = \theta$$
$$P_Y(b) = 1 - \theta$$

Knowing the parameter θ allows predicting the behaviour of the caregiver in a future trial. Evaluating different hypotheses and choosing the one that better predicts the data, the MLE can find an estimation of the parameter θ using the equation:

$$\hat{\theta} = \frac{N_a}{N_a + N_b} \tag{4.1}$$

This equation has been derived in Section 3.2.3 via equations number 3.7, 3.8, and 3.9. During the learning phase the MLE adds to an internal counter a value for each observed event, eventually reaching N_a and N_b. When the dataset is small enough that some events can have not yet been observed a problem arises: the MLE assigns zero probability to those events, negatively affecting inference. To solve this issue it is often recommended to initialise the count of each event to one instead of zero (Russell & Norvig, 2010). This solution does not perturb the posterior distributions in any significant way, and it was used in the simulation given the small size of the dataset. All the parameters of the networks have been set during the learning phase without any external intervention.

The MLE can also be used in more complex scenarios. Let's suppose that the child wants to predict both actions and beliefs of the caregiver. To describe this new example in term of BNs we need the node X (representing beliefs) in addition to the node Y. Because the node Y has a parent, its conditional probability table has four entries. We call θ_2 and θ_3 the parameters associated to Y, and θ_1 the parameter associated to X. In this scenario the child has to collect a new set of information to effectively predict the belief of the caregiver. When the caregiver performs an action, the child has to estimate the related belief and incorporate it for future prediction. The MLE can estimate $\theta_1, \theta_2, \theta_3$ counting the number of events and using Equation 4.1. The posterior distributions are then estimated through Bayesian inference as described in Section 3.2.2.

4.3 Experiments

To test the model a developmental experiment (Vanderbilt et al., 2011) has been replicated in a simulated environment. The approach in Vanderbilt et al. (2011) is particularly relevant because it directly investigated the correlation between trust and ToM. The experiment concerned children with mature and immature ToM who had to deal with two kinds of informant: helpers and trickers. The children assisted at two different scenes in which an adult indicated to a protagonist the location of a sticker hidden inside one of two boxes. The helper always revealed the correct location of the sticker, whereas the tricker always gave wrong advice. After observing the scene it was the child's turn to be the finder. The helper and the tricker gave conflicting advice to the child who had to guess where the sticker was. The *Theory of Mind Scale* (Wellman & Liu, 2004) was used in order to estimate the maturity of the child's ToM. The Theory of Mind Scale is a five-item scale containing tasks that measure the developmental progression of children's mental state understanding. The tasks ask children to reason about situations in which a protagonist

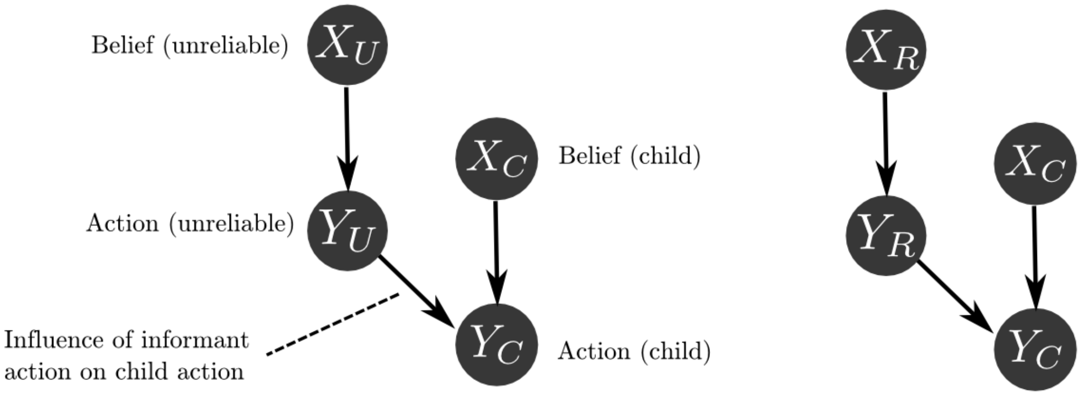


Figure 4.1: The BNs integrated in the agent's cognitive system. The network on the left represents the relation between the child and the unreliable informant (tricker), the network on the right represent the relation between the child and the reliable informant (helper). The subscript U (unreliable) and R (reliable) has been used to distinguish the informants' belief and action, whereas the subscript C represents the child.

has different preferences than their own. Five factors are investigated: diverse-desire, diverse-beliefs, knowledge-ignorance, false belief, false emotions. In order to investigate the relation between trust and ToM further, the authors used also metacognitive questions. The children observed a new pointer helping two finders and another pointer tricking two finders. After observing each new pointer, the children answered four forced-choice questions each one investigating a different factor: intention judgment, same-context prediction, trait judgment, different-context prediction. For more details about the Tehory of Mind Scale and the metacognitive questions see Wellman & Liu (2004) and Vanderbilt et al. (2011).

4.3.1 Methods

To reproduce Vanderbilt et al. (2011) a simulated environment with two different artificial agents has been developed. The first agent represented children with mature ToM and the second agent represented children with immature ToM. Because in the original experiment helpers and trickers act in separate contexts two separate BNs have been included into the agents' cognitive systems. The first BN modelled the interaction between the agent and the helper, whereas the second BN modelled the interaction between the agent and the tricker. As said before this is consistent with the original experiment, in which the same child never received suggestions from both informants at the same time. A graphical illustration of the two BNs can be observed in Figure 4.1. The two nodes X_C and Y_C are beliefs and actions of the agent. The posterior distribution of the node Y_C allows the agent to choose one action among all the possible outcomes. In our case there are only two possible actions: choose box a and choose box b. For example, if $P_{Y_C}(a) = 0.8$ and $P_{Y_C}(b) = 0.2$ the agent will choose the box a because the associated action has a higher probability. The connections between Y_U and Y_C, and between Y_R and Y_C represent the influence that the opinions of the informants have on the agent's action. The action of the agent is then a consequence of its own belief X_C and the informant action Y_R or Y_U. Because there were two categories of agents (Mature ToM VS Immature ToM) and two kinds of pointer (Helper VS Tricker) there was a total of four BNs with four related datasets: Mature ToM and Reliable Pointer, Mature ToM and Unreliable Pointer, Immature ToM and Reliable Pointer, Immature ToM and Unreliable Pointer. Following the procedure in the original experiment the simulation is split into three parts: familiarisation, decision making, and belief estimation.

Familiarisation

The familiarisation consisted of learning the parameters of each BN using the MLE and the associated dataset. The Bayesian approach makes it possible to make inferences with limited data. Each dataset consisted of only six trials, where each trial represented an interaction between the child and the informant. The number of trials is exactly the same as in the original experiment. In Vanderbilt et al. (2011) there was a session where the child watched the helper interacting with two finders for a total of six trials. In a second session the child watched the tricker interacting with two different finders, also in this case for six trials. As for the original experiment the box selection has been counterbalanced, 50% of the time the helper suggested the box a and 50% the box b. The suggestions were always correct revealing the correct position of the stickers. In the other session a tricker always gave wrong suggestions. The dataset associated with the tricker contained six entries, with each box recommended half of the time. The recommendations were always incorrect. During the familiarisation phase the information acquired by the agent with immature ToM differed substantially from the information acquired by the agent with mature ToM. The agent with immature ToM associated the action Y_U to the wrong belief X_U, whereas the agent with mature ToM identified the deception and associated to Y_U the real belief X_U. Because of this deficit in reading the informants' intention the agent with immature ToM collected wrong statistical data and the inference was then distorted in subsequent phases. This bias is what divides a mature ToM from an immature ToM. It is well documented in the literature but the exact mechanism behind it is still not well understood. The bias has been integrated in the model as part of the familiarisation phase but it is beyond the scope of this thesis to explain its origin, see Wellman et al. (2001) for an exhaustive meta-analysis of the phenomenon.

Decision Making

In the decision making phase the agent had to choose one of two boxes given the informant's opinion. To test this condition the informants' action node $P_{Y_R}(a) = 1.0$ and $P_{Y_U}(a) = 1.0$ have been set as evidence. Such a configuration corresponds to an informant indicating box a as the one containing the sticker. In a similar way it was possible to set the network to model a situation in which the informant indicates box b but for the sake of clarity that result has been omitted in the tables. After this preliminary phase the message passing algorithm (Kim & Pearl, 1983) was used to compute the posterior distributions for each node.

Belief Estimation

The last phase of the simulation was the belief estimation. In Vanderbilt et al. (2011) the children were asked some metacognitive questions in order to investigate their perception of the two informants. These questions were used to determine whether the children could correctly identify the tricker as unreliable and the helper as reliable. To test the network on this task the action and belief nodes were set as evidence: $P_{Y_C}(a) = 1.0$ and $P_{X_C}(a) = 1.0$. Such a configuration represents the agent inferring actions and beliefs of the informants given evidence that the sticker is in the box a. Like in the previous phase after this preliminary step the message passing algorithm was used to compute the posterior distributions of the network.

4.3.2 Results

The final results in Vanderbilt et al. (2011) showed that only children with mature ToM distinguished between helpers and trickers, trusting helpers significantly more often than trickers $t(29) = 2.340$, $p = .026$. Moreover, it was observed a significant decrease in trust displayed between the ages of 3 and 4 years, $F(1,58) = 18.647$, $p < .001$, but not between

Table 4.1: Agent with Mature ToM. The table on the left represents the outputs of the BN after the interactions with the reliable pointer (helper), the table on the right represent the outputs after the interactions with the unreliable pointer (tricker). The rows of the tables represent the posterior probability distributions associated with each node of the networks, for both decision making (DM) and belief estimation (BE) tasks.

	DM		BE	
	a	b	a	b
X_C	0.5	0.5	1.0	0.0
Y_C	0.65	0.35	1.0	0.0
X_R	0.8	0.2	0.57	0.43
Y_R	1.0	0.0	0.62	0.38

Helper

	DM		BE	
	a	b	a	b
X_C	0.5	0.5	1.0	0.0
Y_C	0.35	0.65	1.0	0.0
X_U	0.2	0.8	0.57	0.43
Y_U	1.0	0.0	0.38	0.62

Tricker

Table 4.2: Agent with Immature ToM. The table on the left represents the outputs of the BN after the interactions with the reliable pointer (helper), the table on the right represent the outputs after the interactions with the unreliable pointer (tricker). The rows of the tables represent the posterior probability distributions associated with each node of the networks, for both decision making (DM) and belief estimation (BE) tasks.

	DM		BE	
	a	b	a	b
X_C	0.5	0.5	1.0	0.0
Y_C	0.65	0.35	1.0	0.0
X_R	0.8	0.2	0.57	0.43
Y_R	1.0	0.0	0.62	0.38

Helper

	DM		BE	
	a	b	a	b
X_C	0.5	0.5	1.0	0.0
Y_C	0.65	0.35	1.0	0.0
X_U	0.8	0.2	0.57	0.43
Y_U	1.0	0.0	0.62	0.38

Tricker

the ages of 4 and 5 years. In particular, the 3-year-olds followed the suggestions on almost all trials, even when pointers had previously tricked others. The children's score on the finding task was positively correlated with their performance on the Theory of Mind Scale, $r(87) = .339$, $p = .001$. This result seems to confirm the fact that children's reasoning about whom to trust is strongly correlated with their understanding of mental life. Other research is consistent with this conclusion (Lee et al., 2002; DiYanni & Kelemen, 2008). The simulations were coherent with these results. The agent representing children with mature ToM recognized helpers and trickers. When the informant was a helper the agent accepted the suggestion, when the informant was a tricker the agent rejected the suggestion. The agent representing children with immature ToM did not predict the behaviour of the tricker, as observed in the experiment.

To understand the results we can directly examine the output of each node in the BNs after the inference phase. Four tables that illustrate the posterior distributions of each BN have been reported. In the following sections the results reported in these tables are discussed. The use of a probabilistic approach made it possible to have a closer look at the internal mechanics of the model, and it is possible to take advantage of this in the discussion of the results. By studying the posterior distributions we get a clear understanding of the decision taken by the agents in the simulation.

Mature ToM

Table 4.1 illustrates the output of the BNs for the agent with mature ToM. In the decision making task the reliable pointer indicated the box a when the sticker was in that box. The suggestion was accepted by the agent as demonstrated by the inequality $P_{Y_C}(a) > P_{Y_C}(b)$. When the informant was unreliable the same suggestion was rejected. The network output showed the rejection in the form $P_{Y_C}(a) < P_{Y_C}(b)$, which means that the agent selected the box b. To test the belief estimation it has been assumed the agent knew which box the sticker was in and it had to guess the informants' belief. In the query to the BN the stickers were inside the box a. After the computation of the posterior distributions it is possible to observe the inequalities $P_{Y_R}(a) > P_{Y_R}(b)$ and $P_{X_R}(a) > P_{X_R}(b)$, which shows that the model successfully anticipated the actions and the belief of the reliable pointer. A query similar to the previous one was sent to the BN that modelled the interaction with the unreliable informant. The computed posterior distribution was $P_{Y_U}(a) < P_{Y_U}(b)$

and $P_{X_u}(a) > P_{X_u}(b)$. These inequalities imply that the agent identified a discrepancy between belief and action, and predicted the malevolent intentions of the tricker.

Immature ToM

Table 4.2 illustrates the output of the BNs for the agent with an immature ToM. In the decision making task the reliable pointer indicated the box a when the sticker was in the box a. The suggestion was accepted as demonstrated by the inequality $P_{Y_c}(a) > P_{Y_c}(b)$. The same suggestion was given by the unreliable informant but the sticker in that case was in the box b. Because of the lack of ToM the agent could not recognise the deception and accepted the advice. It is possible to observe the result in the posterior distribution $P_{Y_c}(a) > P_{Y_c}(b)$. In the belief estimation task the model produced the right posterior distributions for the helper but the wrong distributions for the tricker: $P_{Y_u}(a) > P_{Y_u}(b)$ and $P_{X_u}(a) > P_{X_u}(b)$. These inequalities show that the agent cannot predict the malevolent intent of the tricker returning a wrong distribution for the action node Y_u.

4.4 Discussion

In this chapter a module for estimating trust in HRI has been introduced. Starting from an existing probabilistic model of ToM (Butterfield et al., 2009), trust has been integrated as a main component of the model. Following a developmental approach BNs have been used as a probabilistic approach for integrating trust and ToM into a common scheme. The MLE was used to set the network parameters. It has been illustrated how the MLE can be considered a mathematical extension of the process that allows the children to collect their own statistical information from the others'. To verify the reliability of the model a developmental experiment (Vanderbilt et al., 2011) has been replicated. In this experiment children with mature and immature ToM selected the advice of reliable and unreliable informants regarding the location of some stickers. The results showed that children with mature ToM could identify the unreliable informant, whereas children with immature ToM could not. In the simulated environment the same results have been observed.

4.5 Conclusions

It is important to point out that the proposed model can be extended to more complex situations. For example, similarly to Butterfield et al. (2009) it can take into account the contemporary influence of two informants, and consider the weigh of each opinion based on past reliability. Thanks to the flexibility of BNs it is straightforward to reorganise nodes and edges for representing this new scenario. When two informants give advice at the same time, a single network is sufficient to integrate them in the decisional process of the child. In this network the action nodes of the two informants Y_U and Y_R have a direct connection to the child node Y_C. The posterior distribution of Y_C is influenced by the internal beliefs and the two external actions. Using the same approach it is possible to integrate more than two sources. Moreover the model is not limited to situations where the tricker cheats every time, it can integrate partially reliable and unreliable users. The posterior distribution $P_1(a)$ of an informant who always told the truth, will be higher than the distribution $P_2(a)$ of a second informant that sometimes cheated. In this case we have that $P_1(a) > P_2(a)$ with $P_1(a) > P_1(b)$ and $P_2(a) > P_2(b)$, meaning that the first informant is more reliable than the second and that both are indicating the right box.

Finally, given the growing presence of autonomous systems in our society it is necessary to implement a module that permits the estimate of reliability. The model presented in this work can be considered a good candidate for such a module because it allows an artificial cognitive system to estimate others' belief. The simulation presented here must be considered as a premise. The next step will be to integrate the module in a humanoid robot, in order to estimate the reliability of different users in online activity.

Chapter 5

Head Pose Estimation

5.1 Introduction

One of the components of effective communication is body language. Body language is particularly important in young children whose ability to communicate verbally is still not mature enough. Children use pointing, gestures and touch to compensate for those deficiencies (Wood, 1976). Among all the non-verbal cues gaze and head orientation are the most important. In the animal realm, head pose has a vital role, especially when a single individual identifies a predator and the information is rapidly spread to the herd. Even primates use gaze and attention to convey social information. For example, rhesus monkeys follow gaze and use attentional cues of other monkeys to orient their own attention to objects (Emery et al., 1997). In multiparty mediated communication among humans integrating gaze solves two problems: knowing who is talking to whom, and who is talking about what (Vertegaal, 1999). Tomasello et al. (1995) pointed out that joint attention is strictly connected with head orientation and gaze in children. The child may look to the adult in order to estimate her focus of attention, then mutual synchronisation is established when the two parties look to each other (or to the same object) and both of them demonstrate knowledge of the other's focus of attention. In development, this joint attention is pivotal for language and ToM development, and it has been positively associated with language gains and social and communication skills (Charman, 2003).

In the context of trust the establishment of joint attention via mutual gaze and head pose estimation is very important. Individuals who make eye contact are perceived as more trustworthy and more attractive than individuals who do not make eye contact (Mason et al., 2005). Gaze is essential in social interactions and it can be used to deceive. In fact the gaze shift is widely used in sports competitions for deceiving the opponent. A practical example is given by penalty shootouts, with the player looking in the wrong direction so to misdirect the attention of the goalkeeper. Even in non-human primates the gaze can be effectively used for deception, an interesting review is provided by Emery (2000). Gaze activity and more specifically gaze aversion has been linked with deception and lying. Vrij & Mann (2001) examined videotaped interviews with a convicted murderer using behavioural analysis. The researchers quantified gaze aversion as the number of seconds the murderer looked away from the interviewer. The results showed more gaze aversion, longer pauses, and slower and more frequent speech disturbances than truth-telling.

Most of the psychological research investigating the relationship between gaze cues and trust employs the gaze cueing paradigm (Driver IV et al., 1999). The paradigm identifies two roles in a bidirectional communication between two parties. The first is following another's gaze, which is labelled as *receiving*. The second is gaze-cueing of the other party, that is labelled as *sending*. An interesting work based on this paradigm is the one of Bayliss & Tipper (2006), who investigated how gaze shift is correlated with the perceived trustworthiness of the others. The authors manipulated the predictability of gaze cuing. The subject had to locate the presence of an object on the left or right of the screen. The object was anticipated by a face with directed gaze (left or right). Some faces were completely unpredictive, since half the time they looked toward the subsequent location of a target and half the time they looked to the opposite side. Other faces were predictive, and they looked toward the object all the time. At the end of the experiment two faces (one unpredictive and the other predictive) appeared on the screen and the subject had to decide which subject was more trustworthy. The results of the study showed that the participants perceived the faces that predicted the presence of the object as more trustworthy.

The gaze cueing paradigm has been used to identify other relationships between gaze and trust. King et al. (2011) observed that objects cued by a trustworthy gaze sender were preferred to objects ignored by this sender or those appearing alongside an untrustworthy gaze sender. Süßenbach & Schönbrodt (2014) observed something similar, participants had a greater tendency to follow a trustworthy person's gaze rather than an untrustworthy person's gaze.

Taken all together these results show that gaze and head pose are very important not only in joint attention tasks but also in the perceived reliability of the others. However, while gaze and head pose are related, the gaze can still have a fixation point that is different from the orientation of the head. This difference has an important role in HRI has showed by Kennedy et al. (2015). The researchers used data collected from real experiments to verify if an head pose estimator based on RGB-D was accurate enough to provide a reliable measure of gaze during an interaction between people and robots. The result of this analysis showed that the mean agreement between the RGB-D data and human hand-coded video data was a Cohen's Kappa value of 0.19 that indicates a slight agreement. Even though this result seems to suggest an inconsistency between head pose and gaze, it does not mean that head pose data should be completely dismissed. Head pose can be used as a means of generating basic reciprocal responsive behaviours and be a valuable information when it is difficult to see eyes (e.g. because of occlusions, distance, and low resolution images). For all the aforementioned reasons it is important to build a robust estimator of the focus of attention that can work in real time, in unstructured contexts, and in situations where the gaze information is unavailable. In the next section it is provided a technical overview of the current methods available, and afterwards it is proposed a solution based on a state-of-the-art deep neural network system.

5.2 Overview

In the last few years major advancements in robotics, augmented reality and driving assistance have highlighted the need for robust methods to estimate the head pose in real-world scenarios. For instance, robots are gradually leaving factories and becoming part of our lives as companions and as assistants. It has been shown that in human-robot interaction a coarse pose estimation of the head is a fundamental prerequisite for building trust with users during joint-attention tasks (Zanatto et al., 2016). In the context of autonomous cars a driving assistance system could take advantage of head pose estimation for decelerating the car when pedestrians do not notice the presence of the vehicle (Geronimo et al., 2010). Moreover a similar system can be installed inside the vehicle and used to monitor the driver's awareness. The need for a robust head pose estimation is not limited to these domains. There have been significant applications in surveillance and anomaly detection, human-computer interaction and crowd behavioural dynamics analysis (Baxter et al., 2015). All of these unconstrained scenarios need an estimator which is resistant to variable environmental conditions, and which can evaluate the focus of attention in absence of more accurate information such as the gaze. Here it is necessary to specify what is considered to be a *wild environment*. It is defined as taken in a wild environment those face images exhibiting a large variety in appearance (pose, expression, ethnicity, age, gender, etc), environmental conditions (artificial light, shadows, etc), and containing relevant occlusions (sunglasses, masks, scarves, etc). In this chapter it is showed how CNNs can be considered one of the best algorithms for robust head pose estimation in a wild environment.

The main contributions of this chapter can be summarised in three points:

- This is one of the few thesis that investigated the use of CNNs in head pose estimation. The main contribution is a rigorous evaluation of multiple CNN models and factors. The results are compared with other algorithms, and show how an approach based on CNNs, dropout and adaptive gradient methods represents the state of the art in head pose estimation.
- Deep learning is a rapidly growing field, which is bringing new techniques that can significantly improve the performance of CNNs. Because these techniques have been released in the last few years, there is still a validation process for establishing their cross-domain usefulness. The role of adaptive gradient methods is explored and a valuable contribution to their ongoing validation is provided.
- The results obtained in this work have been used to implement a Python library called Deepgaze. The library includes pre-trained CNNs based on Tensorflow (Abadi et al., 2015) which can run in real-time on GPUs and mobile devices. Deepgaze is released under an open-source license and is available for both academic and commercial purposes. The library is available at the following address: `https://github.com/mpatacchiola/deepgaze`.

5.3 Related Work

The head pose estimation problem has been investigated from different points of view and with different techniques. Devices such as laser pointers, camera arrays, stereo-cameras, magnetic and inertial sensors, have been used to get a stable estimation in controlled situations (Murphy-Chutorian & Trivedi, 2009). More recently some good results have been obtained with commercial depth cameras (Fanelli et al., 2011). However the use of these devices is not always feasible due to space constraints and to technical problems when operating outdoors. The current work made use of RGB images taken from monocular cameras which permits the greatest portability in real-world applications, given the proliferation of such cameras in mobile phones and laptops. This introduction is thus limited to this kind of approach. A complete description of all the methods available is out of the scope of this work, see Murphy-Chutorian & Trivedi (2009) for a recent survey.

The main branches of a functional taxonomy in head pose estimation can be considered to be appearance-based methods, model-based methods and nonlinear regression methods. Appearance-based methods compare the view of a person's head with discrete models which represent pose labels (Wang & Song, 2014; Wu & Trivedi, 2008). With different kinds of template matching techniques it is possible to evaluate the similarity of the input features with the exemplar set. Appearance-based methods are quite simple to implement but suffer from some serious limitations. They cannot estimate discrete pose locations without using interpolation methods.

Model-based methods use geometric information, non-rigid facial models or landmark locations to estimate the head pose (Lia & Pedrycz, 2014). The accuracy of these models is correlated with the quality and quantity of the geometric cues extrapolated from the image. Occlusions and low resolution images can negatively affect the prediction.

Nonlinear regression methods use a labelled training set to create a nonlinear mapping from images to poses. CNNs can be considered as part of these methods. Nonlinear methods have demonstrated their representational ability in tolerating systematic errors in the training set data. For instance, an approach based on a multilayer perceptron (Stiefelhagen, 2004) had for long time the lowest reported mean angular error on the Prima head pose dataset (Murphy-Chutorian & Trivedi, 2009). The use of these methods for head pose estimation has been sporadic. In particular, the use of CNNs for head pose estimation is limited to very few articles. An approach based on CNNs and energy-based models has been proposed for simultaneous face detection and pose estimation (Osadchy et al., 2007). The authors trained the model on a dataset containing images taken in laboratory conditions, and validated it on datasets containing frontal faces with in-plane rotations and faces in profile. The prediction of the network was limited to two degrees of freedom.

This work is valuable but it is ten years old and at that time advanced techniques for training deep networks were not available. Moreover the results relative to the head pose estimation accuracy were not included. A more recent work investigated the use of CNNs with low-resolution images from monocular cameras (Ahn et al., 2014), obtaining the best reported result on the Biwi Kinect Head Pose dataset (Fanelli et al., 2012). The authors did not use any of the most recent techniques available such as dropout or adaptive methods, and their results have been validated on a single dataset that does not contain in-the-wild images. However the results obtained confirm the validity of CNNs in head pose estimation. In Malagavi et al. (2014) CNNs have been used for monitoring the driver alertness. The authors used a six-layer CNN to classify five discrete head poses: left, right, up, down and frontal. Because of the limits implicit in a discrete classification with only five poses it is not possible to compare this work with the others presented here. In Mukherjee & Robertson (2015) the authors used deep convolutional networks to estimate the head pose in multimodal RGB-D videos. Depth information is not always available due to space constraints and to problems operating outdoors. The authors did not test the effect of different numbers of layers or parameters, moreover they did not use any kind of adaptive gradient method or regularisation to improve the performance of the network.

In the next sections it is explained how CNNs can be applied to the head pose estimation problem and how dropout and adaptive gradient methods can boost their performance. The next section is not intended to be an exhaustive description of the mechanics behind CNNs because they have been already introduced in Chapter 3.

5.4 Experiments

In this section are reported the results obtained using CNNs, dropout and adaptive gradient methods on three public datasets: the Prima head-pose dataset (Gourier et al., 2004), the AFLW dataset (Koestinger et al., 2011), and the AFW dataset (Zhu & Ramanan, 2012). The former is a well-known dataset which has been around for more than ten years, and it is considered a classic benchmark for head pose algorithms. The second is a recently released in-the-wild dataset, and it has the largest number of annotated poses currently available. The third is a small in-the-wild dataset used mainly for benchmarks. Other details about these datasets are available in Sec. 5.4.1, 5.4.2 and 5.4.3.

In these experiments a total of four networks has been used: A, B, C, D. The architecture A

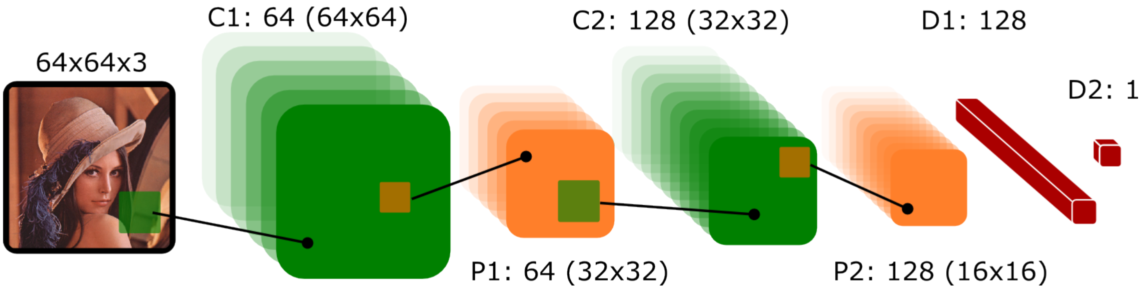


Figure 5.1: Grapichal representation of a Convolutional Neural Network with two convolutional (C1 and C2), two subsampling (P1 and P2) and two fully connected (D1 and D2) layers. The label above each layer specifies number of elements and size (rows × columns) of the future maps. For the dense layers the number of units is reported. The following colour convention has been used to identify the different layers: green for convolution, orange for subsampling, and red for dense layers.

is a standard LeNet-5 (LeCun et al., 1998) with six layers and 4.3×10^6 parameters. The graphical representation of this architecture is presented in Fig. 5.1. The architecture B has one more convolutional layer and one more pooling layer. The number of parameters is slightly higher (4.6×10^6). The idea is to keep the architectures as similar as possible to isolate the effect of the additional layers. It is hard to define a priori how many parameters lead to a good performance, for this reason the third and fourth networks have more weights. The architecture C has a high number of parameters (8.5×10^6). The fourth architecture (D) is similar to a standard AlexNet (Krizhevsky et al., 2012) and it has a total of 9.0×10^6 parameters. A graphical comparison of the four architectures is reported in Fig. 5.2.

The methodology used here is based on a divide-and-conquer strategy, where one CNN for each degree of freedom (roll, pitch, and yaw) has been used. This kind of strategy has the advantage of splitting the main problem into different sub-problems which are easier to manage. Having a specialised network for roll, pitch and yaw, permits fine tuning the network for a specific degree of freedom without losing the predictive power obtained on another one. The methodology consisted of two parts. First, evaluating which optimiser was better suited to a specific dataset. Second, testing CNNs with a variable number of layers and parameters to understand the impact of deeper architectures. As discussed in Murphy-Chutorian & Trivedi (2009) the Mean Absolute Error (MAE) and the Standard Deviation (STD), in degrees, are the best metric of accuracy in head pose datasets with discrete or continuous labels. Both MAE and STD are reported and then used for comparing results. In all of the experiments the same hardware configuration has

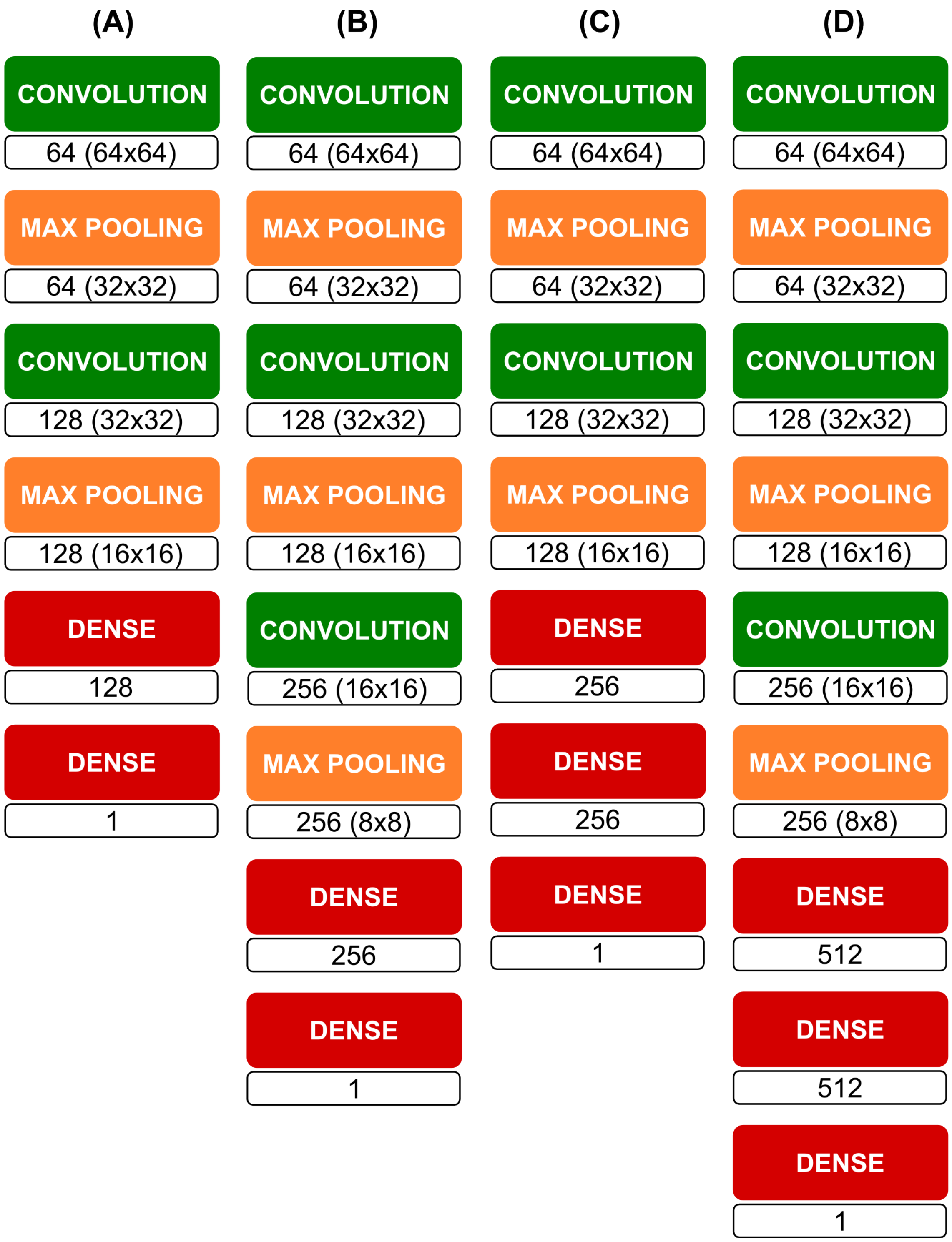

(A)
(B)
(C)
(D)
CONVOLUTION
64 (64x64)
MAX POOLING
64 (32x32)
CONVOLUTION
128 (32x32)
MAX POOLING
128 (16x16)
DENSE
128
DENSE
1
CONVOLUTION
256 (16x16)
MAX POOLING
256 (8x8)
DENSE
256
DENSE
512

Figure 5.2: Comparison of the four architectures used in the experiments. The label below each layer represents the number and size (rows × columns) of the future maps. For the dense layers only the number of units is reported. The networks are organised in order of complexity, on the left there is the network with less parameters and on the right the network with more. Network A has 4.3×10^6 parameters, whereas network B has two more layers and a total of 4.6×10^6. Network C has 8.5×10^6 parameters, whereas network D has two more layers for a total of 9.0×10^6 parameters. The following colour convention has been used to identify the different layers: green for convolution, orange for subsampling, and red for dense layers.

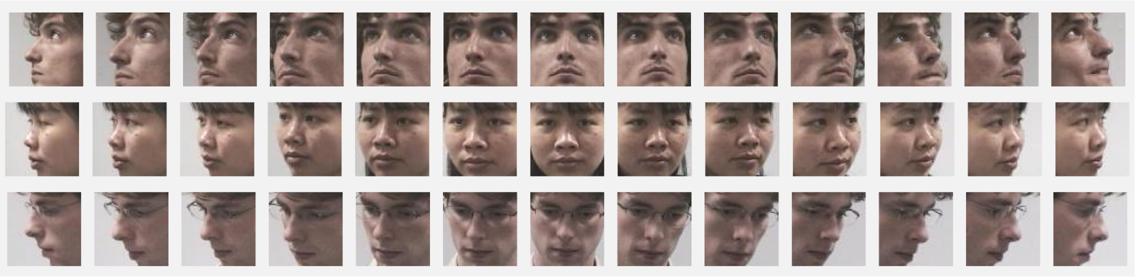


Figure 5.3: This figure represents a collection of images taken from the Prima dataset as they appear after cropping and scaling.

been used: a multi-core workstation with 32 GB of RAM and a GPU NVIDIA Tesla K-40. The experiments have been implemented in Python using the TensorFlow library (Abadi et al., 2015).

5.4.1 Prima head-pose dataset

This dataset consists of 2790 monocular face images of 15 subjects. The subjects range in age from 20 to 40 years old, five possessing facial hair and seven wearing glasses. Pitch and yaw angles are in the range $[-90°, 90°]$ for a total of 93 discrete poses for each person (Fig. 5.3). Two series of pictures with different lighting conditions are provided increasing the total number of available pictures to 186 per person. The head pose is estimated using a procedure called directional evaluation, that consists in defining landmarks with a specific angular displacement with respect to the subject position. The subject has to turn her head toward each one of the landmarks while an image is captured by a frontal camera. This procedure is inaccurate and as a result the dataset is extremely challenging because the predictor must deal with substantial errors and poor uniformity between subjects.

Many different methods have been tested on the Prima dataset. In Tu et al. (2007) a performance comparison is made between high-order Singular Value Decomposition (SVD), Principal Component Analysis (PCA) and locally embedded analysis. Neural networks-based methods have been tested in Voit et al. (2007) and Stiefelhagen (2004). This dataset is the only one in which the human performance has been measured (Gourier et al., 2004).

Methods

This experiment investigated the influence of three factors (network type, optimiser, dropout) and it consisted of two phases:

1. Optimiser selection. In the first phase a standard network has been used to select the best optimiser for this dataset. The standard network used is LeNet-5 (LeCun et al., 1998) shown in Figure 5.1. The optimisers used were the four described in Chapter 3 plus SGD and SGD with momentum.
2. Network selection. In the second phase the best optimiser selected previously has been used for finding the best architecture for each degree of freedom.

In this experiment the sigmoid activation function has been used. This function produces a continuous output in the range $[0,1]$. As a loss function the sum of squares of the differences between the target value y and the estimated value $\hat{y}$ has been used (reported in Eq. 3.11) with learning rate $\lambda = 5 \times 10^{-4}$. The networks have been trained for 20000 epochs, using mini-batches of size 64. The network weights were sampled from a Gaussian distribution ($\mu = 0$, $\sigma = 0.1$), and any values that had a magnitude more than two standard deviations from the mean were dropped and re-sampled. The weights were updated using Eq. 3.10. For each optimiser the learning rate value recommended by the authors has been used (see Chapter 3). When the authors did not suggest any standard value or when the recommended values did not lead to convergence a grid-search procedure has been adopted to find those parameters. Starting from $\alpha = 0.1$ the loss function in the first 1000 epochs has been observed. In case of divergence the learning rate was divided by 2 and the procedure repeated. The value that permitted the fastest convergence was then selected and used for the training. Since a dropout regularisation has been used, the value of the momentum was set to 0.95 as recommended in Srivastava et al. (2014). The faces were isolated using the bounding boxes included in the dataset and then resized to 64×64 pixels.

Two kinds of cross-validation tests were applied to this dataset, in accordance with the procedures reported in Gourier et al. (2006). Testing on known faces was done by dividing the images per series into two separate folds. The two folds contained images of the same subjects taken under different lighting conditions. Testing on unknown faces is done using the Jack-Knife (leave-one-out) procedure, which consists of training the algorithm on all the subjects but one, which is used for testing. The procedure was repeated 15 times, leaving out each subject. The mean value of all the measurements is considered to be the final score. The leave-one-out procedure has been used to select the best optimiser and the best network. In a second moment the best configuration was tested with the known-subjects procedure. During the training any kind of early stopping technique has been adopted. The use of the early stopping requires to monitor the error on a validation set taken from the test set, but the standard procedure reported in Gourier et al. (2006) does not take it into account. Considering that the other methods used the standard procedure, it has been decided to keep the test set unchanged in order to have a fair comparison.

Training the architecture A for 20000 epochs on the two-fold dataset took 4.2 hours, while training it on the fifteen-fold dataset took 18.75 hours.

Results

The results in term of MAE for the first phase (optimiser selection) are reported in Table 5.2. The convergence speed has been analysed, observing the loss value during each of the 20000 epochs. For each optimiser has been considered the loss values obtained from the mean of the 15 subject in the leave-one-out test, and the resulting graphs for both pitch (Fig. 5.4) and yaw (Fig. 5.5) are here reported. The Adam optimiser had the lowest MAEs for both pitch and yaw ($10.71 \pm 11.04, 7.74 \pm 8.03$). A similar performance has been obtained with RMSProp ($10.75 \pm 10.51, 8.3 \pm 8.17$).

In the second phase (network selection) the Adam optimiser has been used to train the four architectures shown in Figure 5.2. The best performances for both pitch and yaw are obtained with the architecture A ($10.71 \pm 11.04, 7.74 \pm 8.03$). A comparison of the four architectures is shown in Figure 5.6. Mapping the continuous output in 9 discrete categories for pitch and 13 discrete categories for yaw, it was possible to interpret the results in terms of classification accuracy. The best architecture (A) has an accuracy of 60.93% (pitch) and 62.33% (yaw) on unknown subjects. The accuracy is even higher on known subjects where it reaches 69.26% (pitch) and 67.61% (yaw). The normalised confusion tables for the best architecture are reported in form of heatmaps in Figure 5.7.

The best architecture has been used also for the know-subjects test and it obtained a MAE of 8.06 ± 8.88 for the pitch estimation (accuracy 73.91%), and a MAE of 6.93 ± 7.32 for the yaw estimation (accuracy 66.6%).

Additional experiments have been done in order to check if data augmentation of the images (horizontal flip) had an impact on the final score. No significant improvement has been noticed.

The best results on the Prima dataset have been obtained with the smallest network (A). However it is possible that networks with less parameters could perform better than the best architecture. This conclusion is reasonable since the Prima datasets contains a few thousands images and overfitting problems could deteriorate the performances of larger networks. To further investigate this hypothesis the Adam optimiser has been used to train two networks. The networks were similar to LeNet-5 (LeCun et al., 1998) with six layers organised as in Figure 5.1. The first network had 2.1×10^6 parameters and the following architecture: C1=32(64×64), P1=32(32×32), C2=64(32×32), P2=64(16×16), D1=128, D2=1. The second network had 0.5×10^6 parameters and the following architecture: C1=16(64×64), P1=16(32×32), C2=32(32×32), P2=32(16×16), D1=64, D2=1. The results did not show any major improvement except for the first architecture that got a slightly lower MAE (10.57 ± 10.78) and an higher accuracy (61.4%) for the pitch estimation on unknown subjects. The same network had a worse performance in yaw estimation with an MAE of 8.36 ± 8.42 and accuracy of 57.67%. The network with 0.5×10^6 parameters reported a score of 11.14 ± 11.24 (accuracy 58.02%) for pitch and 8.38 ± 8.57 (accuracy 57.53%) for yaw, which is significantly below the score obtained with architecture A. Also in the known-subjects test it was not observed any major improvement. The network with 2.1×10^6 parameters obtained an MAE of 8.24 ± 9.81 (accuracy 71.15%) for pitch and 7.23 ± 7.43 (accuracy 64.51%) for yaw. The network with 0.5×10^6 parameters obtained an MAE of 8.9 ± 10.35 (accuracy 70.82%) for pitch and 7.23 ± 7.42 (accuracy 64.19%) for yaw. It is possible to hypothesise that the use of dropout and L2 regularization helped to prevent overfitting in larger architectures leading to stabler solutions.

The comparison between the proposed approach and other methods is reported in Tab. 5.1. CNNs perform better than any other methods on the two-fold test for known subjects and in the leave-one-out test for unknown subjects. The comparison in terms of MAE between human and the proposed method shows how CNNs outperform naive humans in both pitch and yaw, and pre-trained humans only for the yaw estimation. The comparison in terms of accuracy (Tab. 5.1) shows that the proposed method has the best reported scores for tests on known and unknown subjects. In this case the performance of pre-trained humans for pitch estimation on unknown subjects (59%) is lower than the network score

Table 5.1: In this table are compared the results of the proposed method with the result obtained by other authors on the Prima head pose dataset. These results have been obtained training architecture A for 20000 epochs on both unknown (18.75 hours) and know (4.2 hours) test with the Adam optimiser. The results are in term of MAE(accuracy), with MAE expressed in degrees and Accuracy expressed in percentage. The best scores are in bold.

	Unknown Subjects		**Known Subjects**	
Method	**Pitch**	**Yaw**	**Pitch**	**Yaw**
Human Performance (Gourier et al., 2006)	12.6(48)	11.9(42.4)	–	–
Human Performance (training) (Gourier et al., 2006)	9.4(59)	11.8(40.7)	–	–
Locally Embedded Analysis (Tu et al., 2007)	–	–	17.44(50.61)	15.88(45.16)
High-order SVD (Tu et al., 2007)	–	–	17.97(54.84)	12.9(49.25)
PCA (Tu et al., 2007)	–	–	14.98(57.99)	14.11(55.2)
Neural Network (Voit et al., 2007)	–	–	12.77(52.1)	12.3(41.8)
Large Margin Likelihoods (Ricci & Odobez, 2009)	–	–	10.5	9.1
Associative Memories (Gourier et al., 2006)	15.9(43.9)	10.3(50.04)	10.1(61.7)	8.5(60.8)
Steerable Filters (Alioua et al., 2013)	13.8	11.0	12.4	9.6
Steerable Filters (manual) (Alioua et al., 2013)	11.4	9.97	10.1	8.7
Convolutional Neural Networks (mine)	**10.57**(**61.4**)	**7.74**(**62.33**)	**8.06**(**73.91**)	**6.93**(**66.6**)

Table 5.2: In this table are reported the results obtained on the Prima dataset for leave-one-out (unknown subjects) test using different optimisers. These results have been obtained training the architecture A for 20000 epochs (18.75 hours). The results are in terms of MAE (accuracy) with MAE expressed in degrees and Accuracy expressed in percentage. The best scores are in bold.

Optimiser	**Pitch**	**Yaw**
Adadelta (Zeiler, 2012)	20.71(35.3)	13.7(35.23)
Adagrad (Duchi et al., 2011)	12.57(53.69)	9.23(54.73)
Adam (Kingma & Ba, 2014)	**10.71**(**60.93**)	**7.74**(**62.33**)
RMSProp (Tieleman & Hinton, 2012)	10.75(57.67)	8.30(58.92)
SGD	12.87(52.11)	9.06(56.06)
SGD (Momentum)	13.26(49.35)	8.66(56.81)

(60.93%).

5.4.2 Annotated Facial Landmarks in the Wild

The AFLW dataset (Koestinger et al., 2011) provides a collection of 21997 annotated images gathered from an image hosting website. The dataset contains a large variety of appearances (age, ethnicity, occlusions, expressions, etc), lighting and environmental conditions for both genders (56% female, 44% male). Images comparable to the datasets are shown in Figure 5.8. As reported by the authors the ratio of non-frontal faces (66%) is higher than in other databases. The head pose is estimated from 21 manually annotated landmarks using the POSIT algorithm (Dementhon & Davis, 1995).

Different methods have been tested on this dataset (Aghajanian & Prince, 2009; Hegde et al., 2012; Sundararajan & Woodard, 2015; Torki & Elgammal, 2011; Zhu & Ramanan, 2012). To compare the proposed method with the others has been used the data reported in Sundararajan & Woodard (2015), where the authors describe the performance in terms

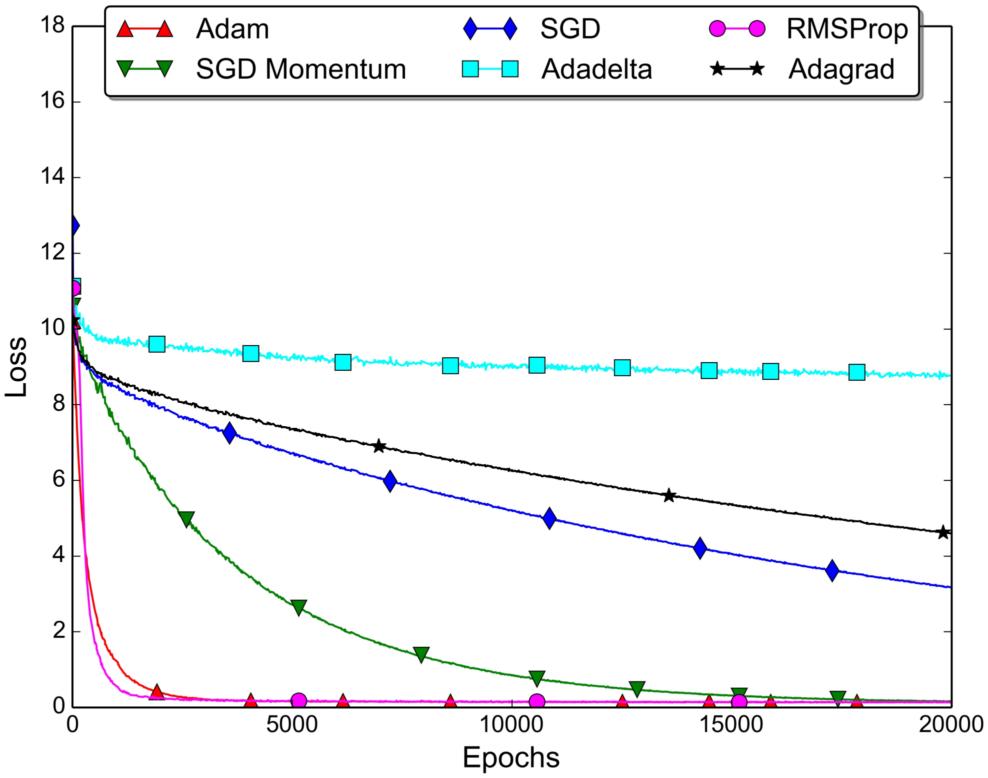


Figure 5.4: Comparison of different optimisers (trained on network A) for the estimation of the pitch angle on the Prima dataset.

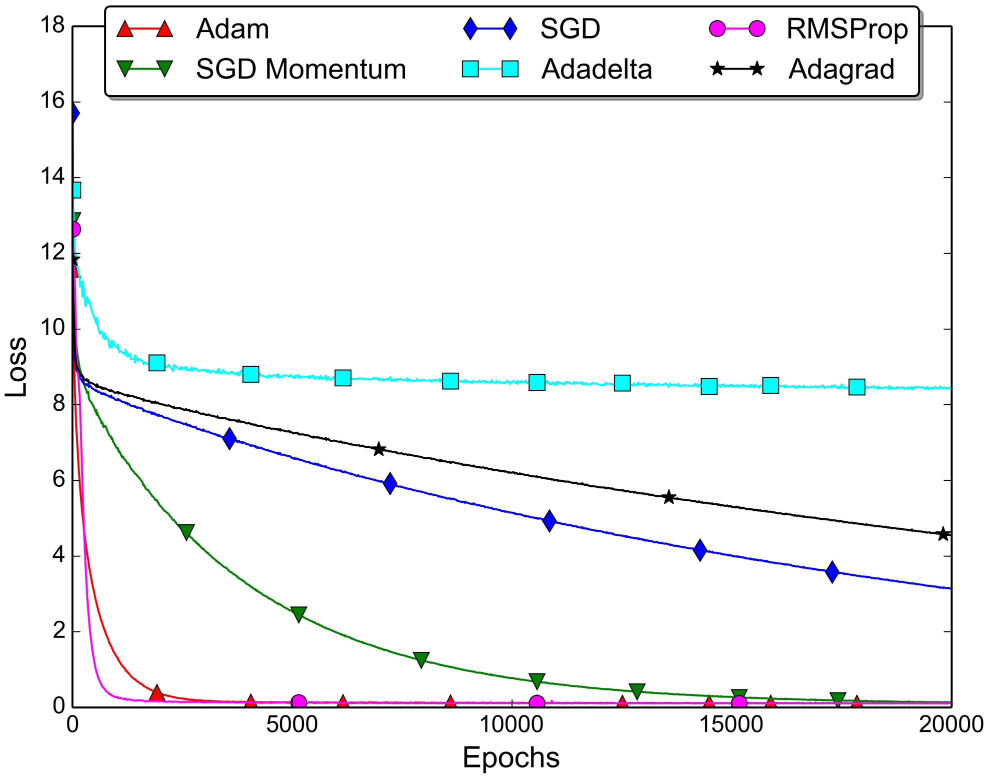


Figure 5.5: Comparison of different optimisers (trained on network A) for the estimation of the yaw angle on the Prima dataset.

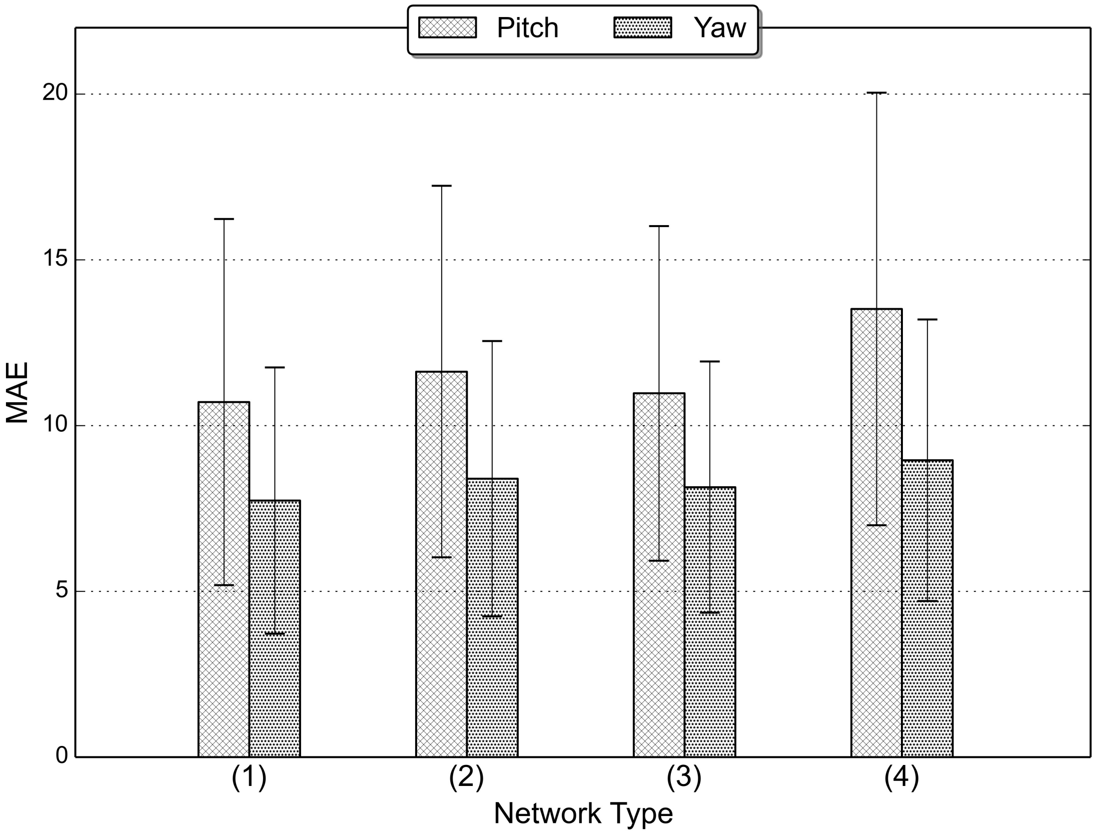


Figure 5.6: Comparison of the performances of the four networks trained with the Adam optimiser in pitch and yaw estimation of unknown subjects (Prima dataset). The STD has been shrunk by a factor of two for graphical reason.

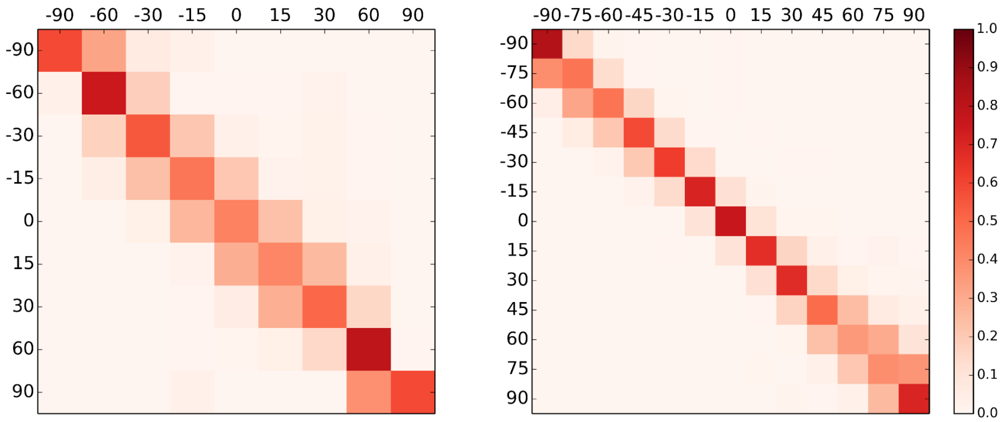


Figure 5.7: Representation of the confusion tables for pitch (left) yaw angle (right) of the best architecture (A) on unknown subjects (Prima dataset). Each row of the tables represents the instances in a predicted class while each column represents the instances in an actual class.

Figure 5.8: This figure represents a collection of images which are comparable to the AFLW and the AFW datasets (the licenses do not allow publishing the original images). The AFLW and the AFW datasets contain a large variety of appearances (age, ethnicity, occlusions, expressions, etc), lighting and environmental conditions for both genders.

of MAE for all of these methods, although limited to the yaw angle. The accuracy has been measured by dividing the range $[-90°, 90°]$ into steps of 15°, and it is intended as the percentage of images within ±15 degrees of error. In the measurement of the yaw accuracy has been used the same metric to enable a comparison with these works.

Methods

In this experiment two factors have been manipulated: network type and optimiser. The experiment was divided into two phases:

1. Optimiser selection. In the first phase a standard network has been used to select the best optimiser for this dataset. The LeNet-5 (LeCun et al., 1998) has been trained using the four optimisers described in Chapter 3 plus SGD and SGD with momentum.
2. Network selection. In the second phase the optimiser selected previously has been used for finding the best architecture among the four previously described (Fig. 5.2).

The AFLW dataset is extremely challenging because the distribution of poses is not uniform. Roll has a mean and standard deviation of 1.07 ± 14.04 degrees, pitch -8.1 ± 13.4 degrees, and yaw 1.91 ± 41.8 degrees. Because of this asymmetrical distribution it is difficult to uniformly map the angles in a continuous interval without losing precision. For this reason it has been decided to take only images above and below 2.698 standard deviations (1.5 the interquartile range of the lower and higher quartile). The final distribution contained poses in the following ranges: roll=[−25°,25°], pitch=[−45°,45°], yaw=[−100°,100°]. To further compensate for the high variability in the angle distribution, it has been decided to use the hyperbolic tangent activation function. Using the hyperbolic function the output of the networks was constrained to within the range $[-1, 1]$ instead of $[0, 1]$. Only faces with a bounding box greater or equal to 64×64 pixels have been considered. In total very few images were discarded (less than 5%) because the mean and standard deviation of the bounding boxes was 300 ± 343 pixels, with 84% of the samples distributed between 110 and 647 pixels. In the first phase the networks have been trained for 20000 epochs and in the second phase for 30000 (mini-batches of size 64). In both phases dropout whit $p = 0.5$ has been used. In the second phase it has been decided to extend the number of epochs to 30000 because the training was much faster compared to the Prima dataset (e.g. 8.0 vs 18.75 hours for network A). Training the network for more epochs generally leads to stabler solutions and it is recommended when using dropout without time constraint (Krizhevsky et al., 2012).

A five-fold cross-validation procedure was used to test the networks on the dataset. After randomly dividing the dataset into five folds, the models were trained on four of the five folds and tested on the remaining one. The procedure was repeated five times, independently testing the model on each one of the five folds. The mean of the five tests was considered to be the final score. The total number of images included in each one of the five folds was 16696, whereas the number of images left for the test was 4173. To test the classification accuracy the yaw poses in the range $[-100°, 100°]$ have been split into different categories using steps of 15°. The accuracy is intended as the percentage of images with ±15 degrees error. This measure is in line with the one used in Sundararajan & Woodard (2015) and makes it possible to compare the results obtained with the different methods reported in that article. To have more reliable results the mean of the five folds has been considered.

The hyper-parameters selection followed the same methodology described in Section 5.4.1. The training of roll, pitch and yaw for a single experimental condition (20000 epochs) took 8.0 hours on the smallest architecture (A) and 9.8 hours on the largest architecture (D).

Table 5.3: In this table are reported the results obtained on the AFLW dataset for the five-fold cross validation test. These results have been obtained training the architecture A for 30000 epochs (14.6 hours). The results are in terms of MAE(accuracy). MAE is expressed in degrees and Accuracy is expressed in percentage. The best scores are in bold.

Optimiser	Roll	Pitch	Yaw
Adadelta (Zeiler, 2012)	6.26(22.2)	9.98(42.67)	21.87(22.19)
Adagrad (Duchi et al., 2011)	5.27(69.23)	8.24(51.29)	17.83(28.8)
Adam (Kingma & Ba, 2014)	4.81(72.41)	7.78(53.49)	13.5(38.56)
RMSProp (Tieleman & Hinton, 2012)	**4.4**(**75.14**)	**7.15**(**55.98**)	**11.04**(**44.54**)
SGD	7.0(58.64)	9.89(42.15)	22.98(22.23)
SGD with momentum	6.17(63.19)	10.3(42.37)	21.31(22.52)

Results

The results for the first phase (optimiser selection) are reported in Table 5.3. The results show that the RMSProp had the lowest reported MAE for roll, pitch and yaw (4.4 ± 4.35, 7.15 ± 6.0, 11.04 ± 10.86). As it is possible to see in Figure 5.9 the RMSProp had the fastest convergence rate and it reached the lowest loss values.

The results for the second phase (network selection) are reported in Figure 5.10. The architecture B had the lowest MAE (4.15 ± 3.87, 6.8 ± 5.64, 9.51 ± 9.21). Mapping the continuous output in three discrete categories for roll, nine for pitch and 13 for yaw, it was able to interpret the results in terms of classification. The accuracy for roll, pitch and yaw was originally 76.55%,57.68% and 48.55%. Considering an error of $\pm 15°$ the accuracy drastically increases to 99.66%,97.24% and 92.47%. To better visualise the accuracy is reported in Figure 5.11 the confusion table of the best network (second architecture, RMSProp optimiser) for roll, pitch and yaw classification.

The comparison with other methods (Tab. 5.4) shows that CNNs perform better than any other algorithm in terms of MAE and accuracy.

5.4.3 Annotated Face in the Wild

The AFW dataset is a benchmark proposed in Zhu & Ramanan (2012) to test the performance of face detection and head pose estimation methods. Similarly to the AFLW dataset, the AFW is composed of images sampled from social networks. It contains 205 images with 468 faces. Most of the images contain cluttered backgrounds with large variations in both face viewpoint and appearance (age, occlusion, expression, etc). Each face is labeled with a bounding box, 13 discrete viewpoints (range $[-90°, 90°]$) along pitch and

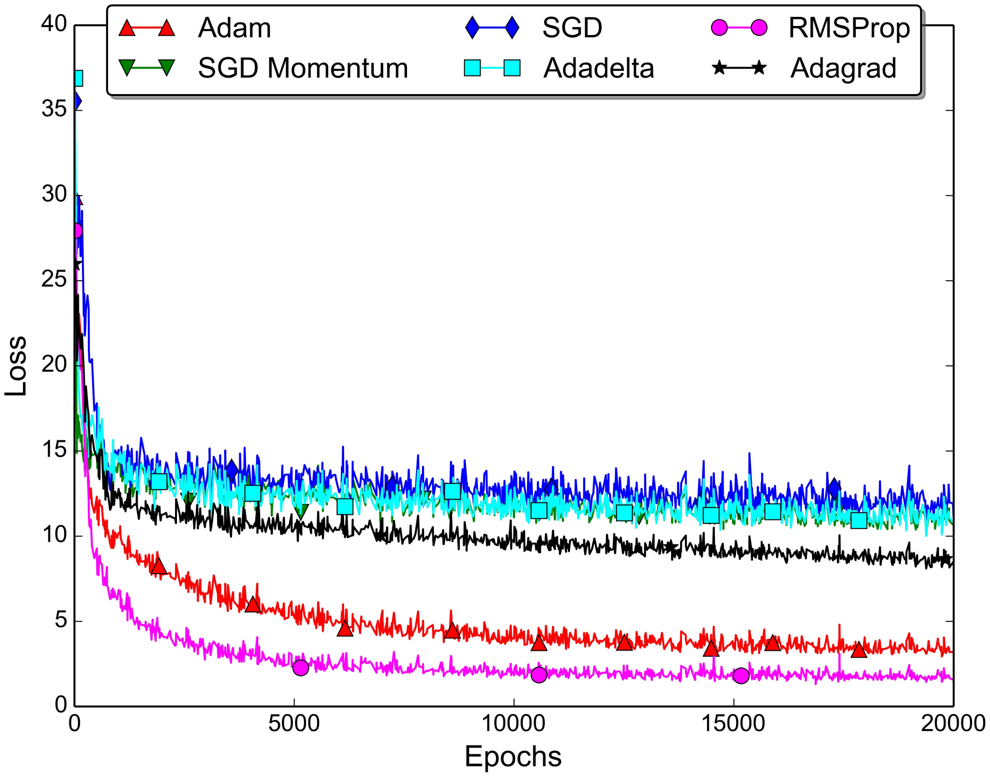


Figure 5.9: Comparison of the convergence speed between the six optimisers used to train architecture A for yaw estimation on the AFLW dataset. The loss values are the mean of the five fold.

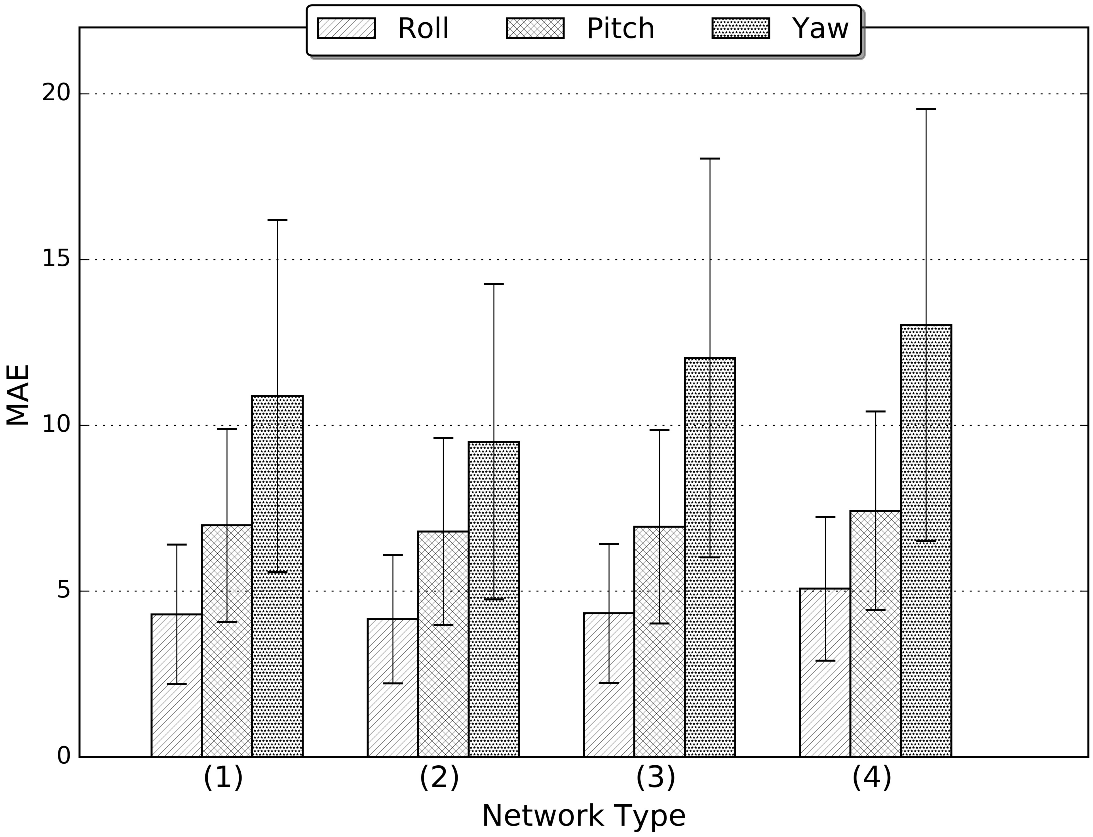


Figure 5.10: Comparison in term of MAE between four architectures for roll pitch and yaw using the RMSProp optimiser on the AFLW dataset. The STD has been shrunk by a factor of two for graphical reason.

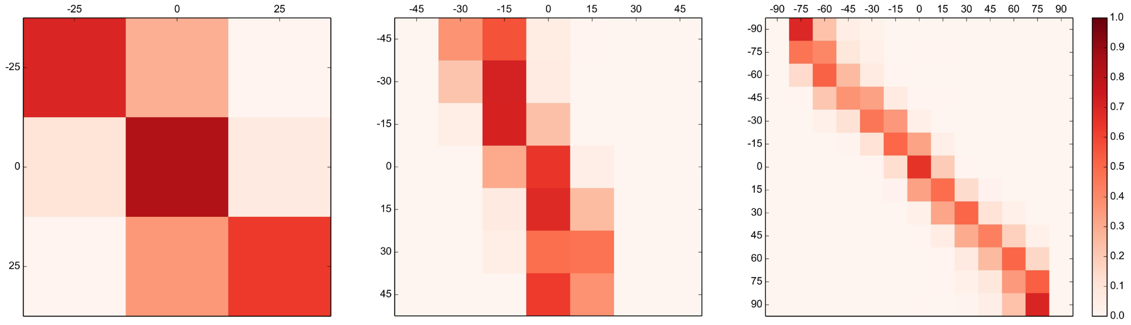


Figure 5.11: Representation of the confusion tables for roll (left), pitch (centre) and yaw (right) of the best architecture (B) trained with the RMSProp optimiser on the AFLW dataset. Each row of the tables represents the instances in a predicted class while each column represents the instances in an actual class.

yaw directions, and three discrete viewpoints along the roll direction (left, centre, right). This dataset differs from similar collections in its annotation of multiple, non-frontal faces in a single image. Pictures compatible with this dataset are shown in Figure 5.8. Given the low number of images this dataset is generally used only for testing (Sundararajan & Woodard, 2015; Zhu & Ramanan, 2012). In this test it has been used the AFW dataset to test the best architectures trained on the AFLW dataset. In this way it was possible to have one more in-the-wild benchmark for comparing the proposed method with other approaches.

Methods

In this experiment the whole AFLW dataset (20869 faces) has been used to train the architecture B. The network was trained for 30000 epochs. It was adopted the RMSProp optimiser with the same hyper-parameters as used in the previous experiment: $\alpha = 0.001$ and $\gamma = 0.9$. Each face presented in the images was cropped and resized to 64×64 pixels. Smaller images were removed (only five in total). The trained network was tested on the AFW dataset. To measure the network performance the average of five learning cycles has been used. The CNN was initialised five times with random weights, trained on the AFLW dataset and tested on the AFW dataset. The average of the five MAEs has been used as the final score.

Results

The results show an MAE and STD of 16.73 ± 17.17 and an accuracy of 32.96% which reaches 75.29% when considering an error of ±15°. This is the best score ever reported on this dataset. The comparison of CNNs with other methods is reported in Table 5.4. Some additional tests have been done in order to augment the data in the AFLW dataset by horizontal flipping. These experiments did not lead to any major improvement in the final performance.

5.4.4 Discussion

The results achieved are here detailed. The first phase of each experiment consisted of the optimiser selection. The results on the Prima dataset show the superiority of Adam on the other methods. The use of adaptive methods has been extremely important also for in-the-wild datasets where they significantly outperform SGD. In the AFLW dataset the RMSProp had the lowest reported MAE and the highest accuracy. It must be pointed out that in the last epochs the SGD with momentum reaches similar loss values of Adam and RMSProp, however the results in terms of MAE are higher. It is possible that the flexibility of Adam and RMSProp allowed exploring the space better than SGD with momentum, especially because of the high variability of the input features. Another advantage of the adaptive methods is the convergence speed. As it is possible to observe from Figures 5.4, 5.5 and 5.9, Adam and RMSProp converge to a very low loss value during the first epochs.

The second phase of the experiments consisted of the architecture selection. In the Prima dataset, comparing architecture A and its deeper counterpart (B) we see that the second architecture had an highest MAE for pitch ($+0.92°$) and yaw ($+0.66°$) estimation. The same effect has been observed between network C and D ($+2.54°, +0.81°$). We can conclude that adding another convolutional layer or adding more parameters did not lead to any improvement. The reasons could be the small size of the Prima dataset and the controlled environment where the pictures have been taken. In the AFLW dataset, comparing architecture A with architecture B we can see that the second architecture had lowest MAEs for roll, pitch and yaw ($-0.15°, -0.18°, -1.38°$). Augmenting the number of parameters did not lead to any improvement. In this case having an additional convolutional layer made a significant difference, and we can conclude that estimating the head pose in unconstrained datasets requires more complex architectures.

The introduction stated that the use of nonlinear regression methods can tolerate systematic errors in the training set. The experimental results confirm this statement. The accuracy of CNNs is higher than any other method meaning that the networks can grab the general rules and are not heavily affected by mislabelling. To visualise the accuracy is reported the confusion tables as heatmaps for both Prima (Fig. 5.7) and AFLW (Fig. 5.11) datasets. The darker cells are disposed along the main diagonal, meaning that the prediction has a high accuracy with false positives constrained to within the proximity of the correct category. The main difference between Prima and AFLW is in the prediction of values which are close to $\pm 90°$. The reason for this difference is the AFLW dataset does not have a uniform distribution and values distant from the mean are very limited.

Table 5.4: In this table are reported the results in term of MAE (degrees) and accuracy (percentage) obtained with different methods for the yaw estimation on the AFLW and the AFW datasets. The results have been obtained training the architecture B for 30000 epochs (14.6 hours) with the RMSProp optimiser. MAE is expressed in degrees and Accuracy is expressed in percentage. The best scores are in bold.

Method	**AFLW**	**AFW**
Mixture of trees (Zhu & Ramanan, 2012)	46.54(15.72)	40.17(26.07)
Patch Based (Aghajanian & Prince, 2009)	38.39(23.87)	41.67(21.36)
Feature Embedding (Torki & Elgammal, 2011)	33.01(32.82)	28.15(40.38)
Learning Manifold (Hegde et al., 2012)	16.31(63.13)	18.26(58.33)
Approximate View (Sundararajan & Woodard, 2015)	17.48(58.05)	17.2(58.33)
Convolutional Neural Networks (mine)	**9.51**(**92.47**)	**16.73**(**75.29**)

5.5 Conclusions

In this chapter the use of CNNs for head pose estimation has been introduced. This approach is significantly different from previous research (Ahn et al., 2014; Malagavi et al., 2014; Mukherjee & Robertson, 2015; Osadchy et al., 2007), and shows how using the most recent deep learning techniques leads to the state-of-the-art in constrained and unconstrained datasets. The method should be considered as part of a broader system, in particular it should be used in conjunction with a face detector. The system has been implemented in Python using TenworFlow (Abadi et al., 2015) and OpenCV (Itseez, 2015). The Viola-Jones object detection framework (Viola & Jones, 2001) has been used as a face detector and two CNNs of type B trained on the AFLW dataset as head pose estimator (pitch and yaw). Camera frames with a resolution of 640×480 have been acquired from a commercial webcam, and then the faces have been isolated using the Viola-Jones algorithm. Finally the isolated faces have been given as input to the CNNs obtaining the head pose. The whole system runs at 15 frames per second on a standard laptop (intel core i5, 8 GB RAM) without the use of a GPU. It must be pointed out that an inadequate face detector can be a significant bottleneck. In this case removing the face detector from the loop allows executing the head pose estimator at 20 frames per second. Moving to the real-world is not straightforward and different problems can arise. For instance, the major problem experienced during the tests was a reduced accuracy when the face detector returned a subframe that was not well centred on the face. To address this problem a valuable solution is data augmentation. Multiple random shifts can be introduced while cropping the image, and the resulting subframes stored in the training set. A neural network trained on this extended dataset will be more robust at inference time.

Although the proposed approach is highly competitive, it can be further improved. Other factors can play an important role as pointed out by recent literature. Future research should particularly focus on the impact of weight initialization (Ribeiro et al., 2016) and sample selection (Yang et al., 2016). Moreover, the results can be further improved once in-the-wild datasets with more images and an extended range of poses are available.

Other important considerations are related to the use of this approach in HRI and in particular in trust-based HRI. Different applications may require different architectures that have to be fine tuned for specific goals. We can think of multiple use cases with different error thresholds. For instance, a robot suggesting products in a store can be effective even with a coarse evaluation of the customers' head pose. On the other hand, a robotic arm operating in a supply chain requires a fine head pose estimation in order to avoid collisions with human operators during pick and place tasks. In the first example an error of a few degrees is tolerated, in the second case misinterpreting the focus of attention of the human may slow down the production line and it may even be dangerous. The system proposed in this chapter can be effective in the first case, but it will require additional modules or additional sensors in order to work properly in the second scenario.

Chapter 6

A Cognitive Architecture for Trust in Humanoid Robots

6.1 Introduction

In this chapter the problem of trust-based learning is directly faced, and the proposed solution integrated into a cognitive architecture. The first step in this direction is to provide again the definition of trust that has been given in Chapter 1: the agent x is the relying agent, who feels trust, and the agent y is the trustee (Falcone & Castelfranchi, 2001). It is important to notice that the agent y is not necessarily a cognitive agent (it can be a machine or a virtual assistant). Since the actions of y are useful to x this means that x is delegating some goal to y. This definition can be easily applied to the developmental experiment described in Section 6.3. In this case the agent x is the child, and the agent y is the adult informant. Moving from an adult-adult context to a child-adult context introduces many additional dynamics that must be taken into account. As stated in Chapter 2, by 4 years of age children are able to identify the most accurate informant (Fusaro et al., 2011; Kim et al., 2016). However, an immature Theory of Mind (ToM) (Premack & Woodruff, 1978) may interfere with those abilities generating evident errors in cognitive reasoning.

In this chapter an AC framework inspired by biological observations, using a BN (Patacchiola & Cangelosi, 2016) as internal environment and the ERA (Morse et al., 2010) as function approximator, is presented. The ERA architecture is a flexible system capable of exhibiting a wide range of psychological effects and operating transparently with concepts. Such transparency makes it easy to extend the architecture with the integration of other systems. This approach can scale up beyond simple scenarios, it is not tailored to specific domains, and displays a developmental trajectory (Morse et al., 2010). The ERA architecture is formed by the structured association of multiple SOMs, each one typically receiving inputs from a subset of neighbours, and each with a single winning unit. Couple of winning unit are associated using a bidirectional connection weighted with positive Hebbian learning. Additional details about the functioning of the ERA architecture are provided in Section 6.2.4.

The methodology followed in this chapter is in tune with the developmental robotics approach (Cangelosi & Schlesinger, 2015; Asada et al., 2001, 2009), and is based on the idea of reproducing a developmental experiment with the iCub humanoid robot (Metta et al., 2008). The use of a robot allows evaluating the role of embodiment in trust and it is the starting point for additional work (see Section 6.5). The aim of this chapter is twofold: on the one hand to define a cognitive model of trust development in children by testing theories on robotic platforms; on the other hand to embed the essential principles of human cognition involved in trust-based learning into artificial agents. More specifically, the problem addressed in this chapter is trust-based learning without an external reward. This problem has received limited attention in artificial intelligence compared to similar topics such as standard reinforcement learning. When an external reward is missing the agent is left to itself with limited information and a decision to be made. The agent has access to previous memories and to a record of past interactions that can be used to make an inference. The inference process produces an internal cost that is used as feedback to drive learning. It must be pointed out that even though the agent does not rely on an explicit external reward, it is still referring to rewards acquired in previous occasions, with the cost estimation built on top of them. These considerations are the essence of the chapter, and the reader is invited to take them into account when the formal introduction of the model is presented in Section 6.2.

The rest of the chapter is organised as follows: in Section 6.2 a detailed explanation of the proposed architecture is provided, in Section 6.3 the model is used to replicate a developmental experiment, in Section 6.4 the results obtained in the experimental phase are discussed, finally in Section 6.5 an overview of the work is presented and future research directions are proposed.

6.2 Model Description

The cognitive architecture presented in this work can be decomposed in four main parts. (i) The external environment, where part of the inputs to the system are acquired. (ii) The internal environment, that returns a cost based on reputation, long term memories, and current state. (iii) An actor unit, that represent the policy to adjust. (iv) A critic unit, that evaluates an error measure used in order to optimise the actor. The interaction between these parts is coordinated through an intrinsically motivated reinforcement learning mechanism. Reinforcement learning is the problem faced by an agent that learns through trial-and-error interactions with a dynamic environment (Kaelbling et al., 1996). In the standard approach the learning process is based on a reward obtained after acting in the world. However we often learn without any explicit reward, using only implicit feedback derived from past experiences. Chapter 2 described how an internal reward signal can be modulated thanks to the anterior cingulate cortex and the medial prefrontal cortex. To formalise the influence of these brain areas, an internal environment is introduced. This design choice is crucial and allows representing the learning problem in terms of intrinsic motivation (Singh et al., 2004). The internal environment may or may not interact with the external one. Here, the focus is on the particular case of an agent interacting with its internal environment only, since trust-based learning of object-label pairs has been defined as a process that does not involve any external reward. The model can be extended to the non-trivial interaction between internal and external environments. Figure 6.1 shows the difference between a standard reinforcement learning agent, an intrinsically motivated agent (Singh et al., 2004), and the particular case considered here.

In the next section is provided a functional overview of the cognitive architecture. Technical details about the AC reinforcement learning loop are given in Section 6.2.3. In Section 6.2.4 is explained how the ERA has been used to model the function approximators in the actor and in the critic. Lastly, in Section 6.2.5 gives more details on the internal environment and on the BN.

6.2.1 Comparison with neurobiological correlates

Starting this chapter with a comparison between the proposed model and the biological counterpart is a good way to give an high-level intuition of the model functioning.

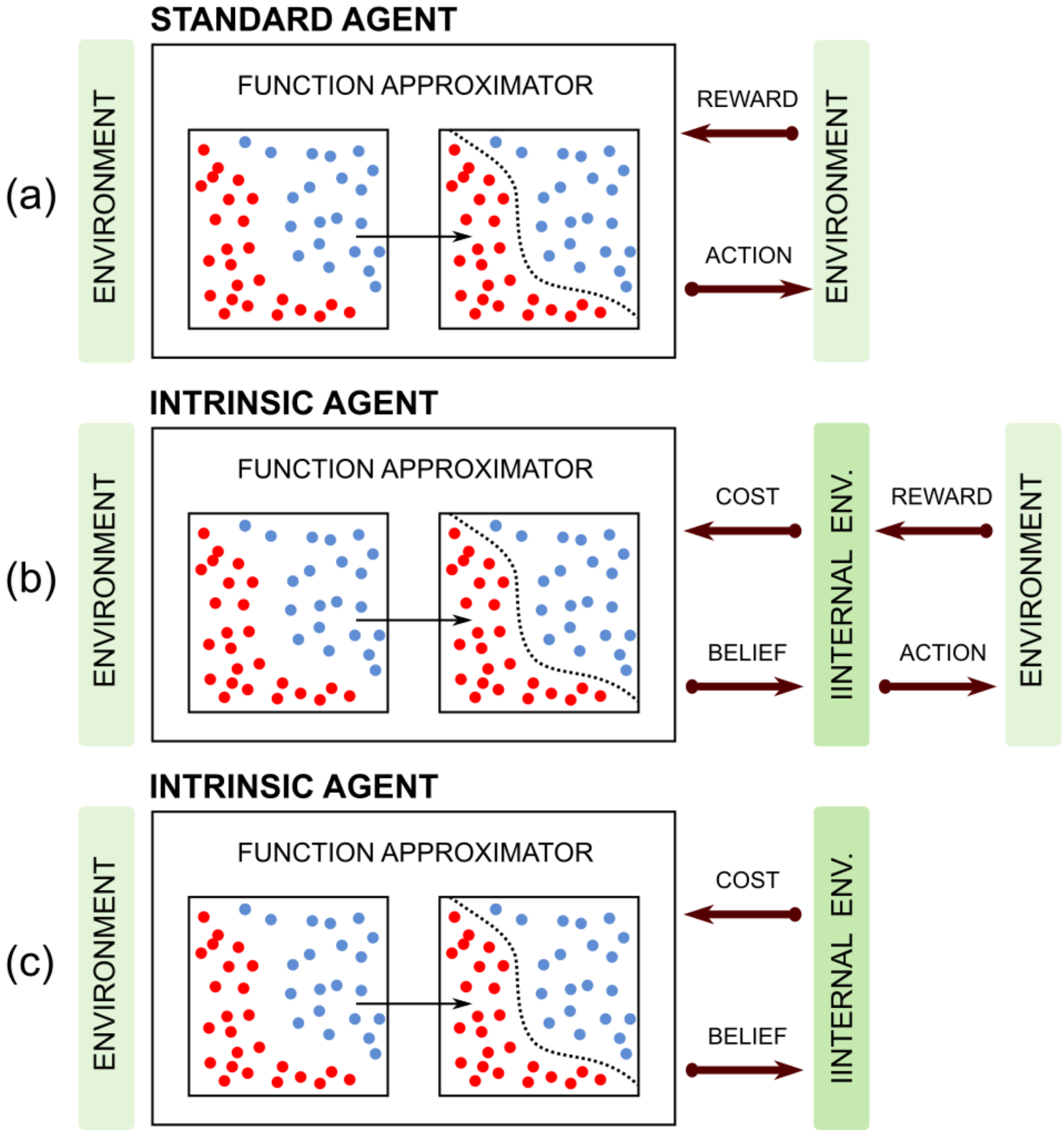

STANDARD AGENT
(a)
ENVIRONMENT
FUNCTION APPROXIMATOR
REWARD
ACTION
ENVIRONMENT
INTRINSIC AGENT
(b)
ENVIRONMENT
FUNCTION APPROXIMATOR
COST
BELIEF
IINTERNAL ENV.
REWARD
ACTION
ENVIRONMENT
INTRINSIC AGENT
(c)
ENVIRONMENT
FUNCTION APPROXIMATOR
COST
BELIEF
IINTERNAL ENV.

Figure 6.1: Difference between a standard agent (a), an intrinsically motivated agent (b) and the particular case considered here (c). The standard agent (a) acquires the state of the world from the external environment and performs actions in the external environment which returns a reward. The intrinsically motivated agent (b) has both an internal and an external environment, it can convert beliefs to actions and it can evaluate the weight of external rewards based on an internal cost. In the case considered here (c) there is no feedback from the external environment and learning is achieved only through an internal cost measure.

Both, the computational model and the biological model, are based on an actor-critic loop, that is controlled by the dopamine and the temporal-difference error. The projections from the ACC and the medial frontal cortex influence the activity of the critic. These projections have been modelled as a Bayesian network. Different kind of function approximators can be used in the actor and in the critic, here Self-Organising Maps trained via Hebbian learning have been used. An in-depth description of the neurobiology of trust has been provided in Chapter 2, the reader is invited to review that material in order to better understand this section. A graphical comparison of the computational and biological model is presented in Figure 6.2.

6.2.2 Functional overview

In this section is provided a more detailed overview of the model functioning. In particular, is showed how the different parts of the cognitive architecture cooperate during the interaction of the agent with the informant. To simplify the exposition the learning procedure has been divided in six steps which are summarised below. Moreover a graphical representation of the architecture is presented in Figure 6.3 and the step of the algorithms are presented in Algorithm 1.

1. **Information acquisition.** The agent acquires the information from the environment using its internal sensors. The information in this particular case are the object features, the label associated to the object, and the informant suggestion. The visual and auditory information activates specific neurons in the vision and vocabulary modules in both actor and critic units.

2. **Actor unit activation.** Forward pass in the actor ERA unit. The activation spreads across the network generating the raw output in the belief module.

3. **Belief normalisation.** The raw output of the actor unit is normalised using a softmax function. A belief is sampled from the probability distribution.

4. **Reputation update and cost estimation.** The confidence and reputation distribution in the BN are updated. A forward pass in the BN returns a cost that is passed to the critic.

5. **Temporal differencing.** The temporal differencing update rule is used to update the connection in the critic ERA unit. The error δ is estimated and passed to the actor

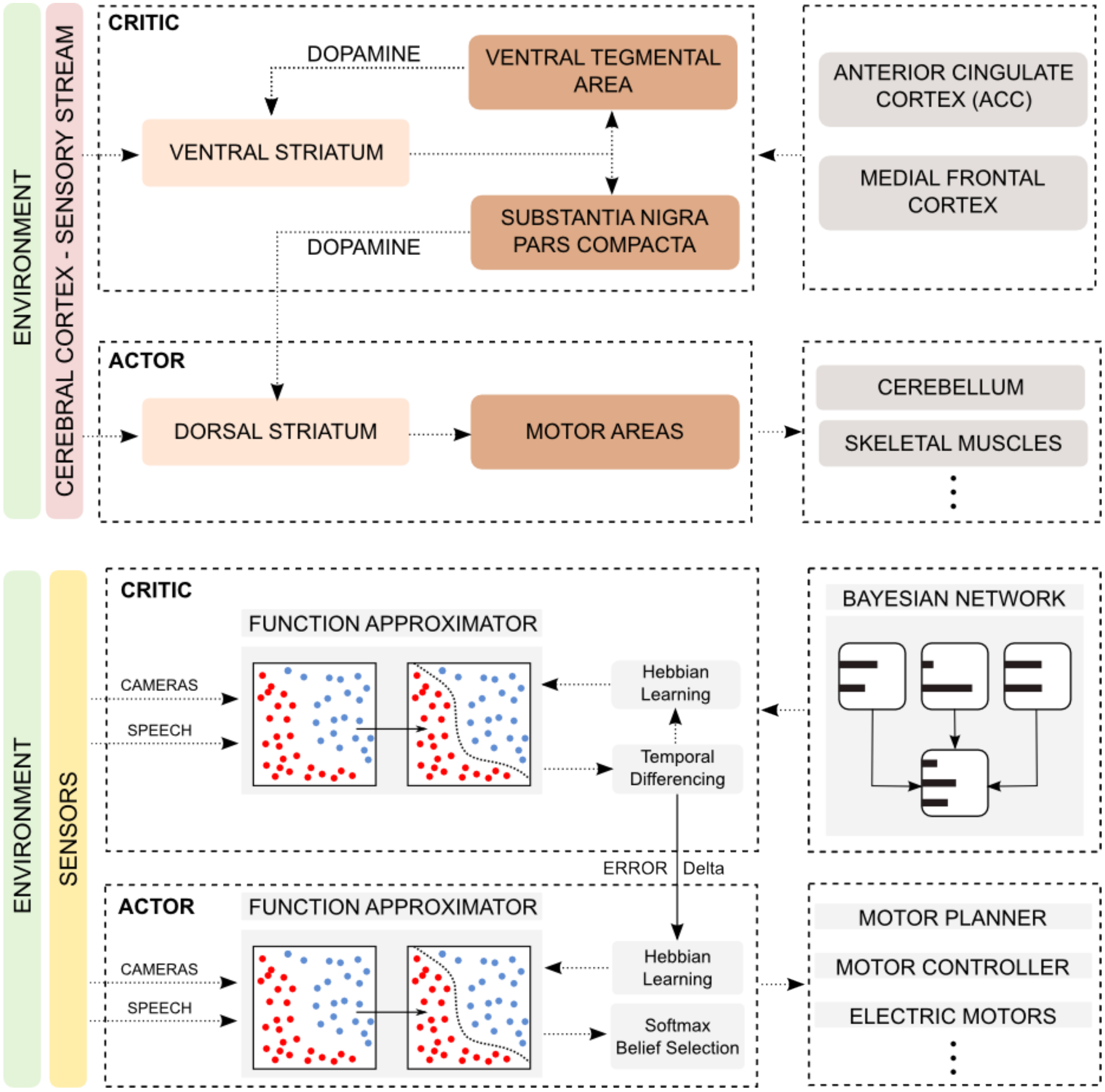

ENVIRONMENT
CEREBRAL CORTEX - SENSORY STREAM
CRITIC
DOPAMINE
VENTRAL TEGMENTAL AREA
VENTRAL STRIATUM
SUBSTANTIA NIGRA PARS COMPACTA
DOPAMINE
ANTERIOR CINGULATE CORTEX (ACC)
MEDIAL FRONTAL CORTEX
ACTOR
DORSAL STRIATUM
MOTOR AREAS
CEREBELLUM
SKELETAL MUSCLES
ENVIRONMENT
SENSORS
CRITIC
FUNCTION APPROXIMATOR
CAMERAS
SPEECH
Hebbian Learning
Temporal Differencing
BAYESIAN NETWORK
ERROR
Delta
ACTOR
FUNCTION APPROXIMATOR
CAMERAS
SPEECH
Hebbian Learning
Softmax Belief Selection
MOTOR PLANNER
MOTOR CONTROLLER
ELECTRIC MOTORS

Figure 6.2: Comparison between the biological model of trust (top), an the computational model presented here (bottom). Both the architectures are based on an actor-critic loop, that is controlled by the dopamine and the temporal-difference error. The projections from the ACC and the medial frontal cortex influence the activity of the critic, they have been modelled as a Bayesian network. The ERA architecture has been used as function approximator in a actor-critic mechanism.

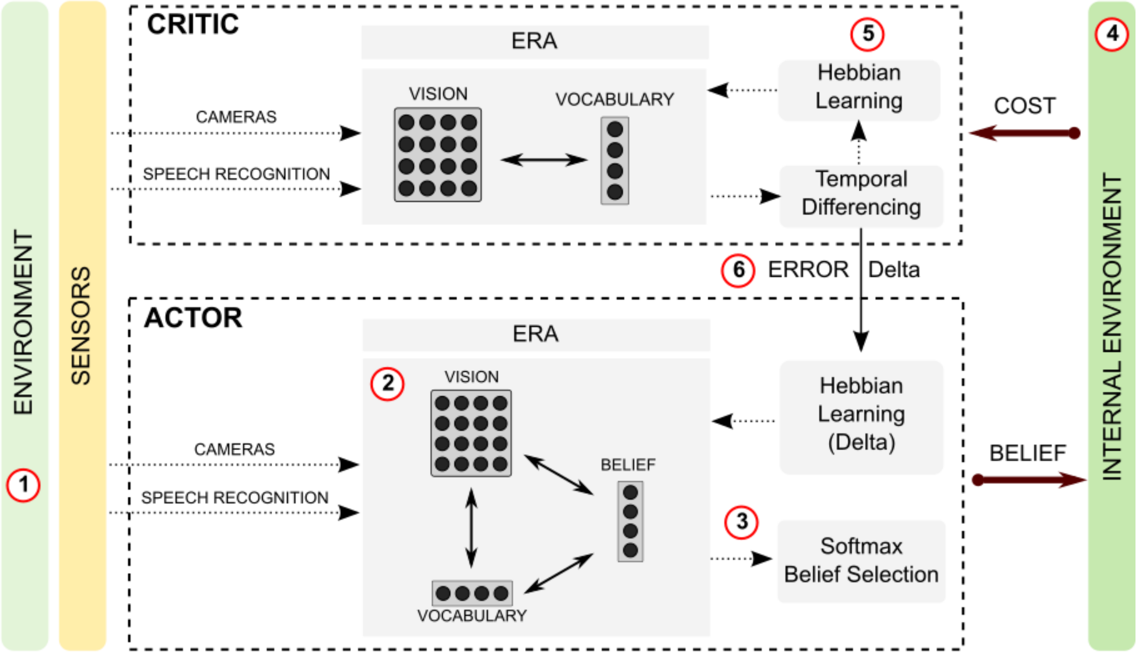

CRITIC
ERA
VISION
VOCABULARY
CAMERAS
SPEECH RECOGNITION
⑤
Hebbian Learning
Temporal Differencing
COST
④
ENVIRONMENT
①
SENSORS
⑥ ERROR
Delta
ACTOR
ERA
②
VISION
BELIEF
VOCABULARY
CAMERAS
SPEECH RECOGNITION
Hebbian Learning (Delta)
③
Softmax Belief Selection
BELIEF
INTERNAL ENVIRONMENT

Figure 6.3: The cognitive architecture implemented in the model. The learning process is represented by 6 steps. In step 1 the information acquired from the environment is used to activate the SOMs in actor and critic ERA units. In step 2 the output of the actor ERAunit is computed based on the activation of vision and vocabulary SOMs. In step 3 a Softmax function is used to squash the raw output into a probability distribution. In step 4 the internal environment estimates the cost using the BN forward step. In step 5 the temporal differencing update rule is used to stabilise the Hebbian connections of the critic unit. In step 6 the error Delta is used to stabilise the actor unit.

unit.

6. **Actor updated.** The error estimated in the previous step is used in order to update the Hebbian connections in the actor ERA module as described in Equation 3.4.

6.2.3 The actor-critic reinforcement learning loop

The reinforcement learning problem is defined through a Markov decision process. The agent starts from an initial state s_0 and it performs a single action taken from a predefined set $A = \{a_0, a_1, ..., a_n\}$. The action leads to a new state s_{t+1} based on a transition model $T(s_t, a_t, s_{t+1})$ which in a model-free scenario is not given. After the action the environment returns a reward r_t following a function $R(s_t, s_{t+1})$. The goal of the agent is to maximise the sum of discounted reward.

In the standard approach at each time step t the state-action function $Q(s_t, a_t)$ in the actor is called and a utility vector is returned. However, since we are dealing with an internal environment the state-action function is defined as a state-belief function. The terms action

Algorithm 1 Trust-based learning algorithm. The pseudo-code shows initialisation, learning loop, and the AC update rule.

```
Initialise: ERA critic unit $E_c$ with random parameters $\theta_c$
Initialise: ERA actor unit $E_a$ with random parameters $\theta_a$
Initialise: BN nodes with uniform distributions
for t=1 to $T$ do
    Wait: interaction with informant $k$
    Read: suggestion $i_t$ from informant $k$
    Read: state $s_t$ from external environment
    Get: belief distribution from $E_a$ through forward pass
    Sample: $b_t$ from belief distribution through softmax
    Estimate: confidence distribution $P_F(1)$ and $P_F(0)$
    Sample: agent confidence $F$ from $P_F$
    if $F = 1 \wedge b_t = i_t$ then
        $N_R^{(k)} \leftarrow N_R^{(k)} + 1$
    else if $F = 1 \wedge b_t \neq i_t$ then
        $N_U^{(k)} \leftarrow N_U^{(k)} + 1$
    end if
    Update: informant reputation $P_I^{(k)}(1) \leftarrow \frac{N_R^{(k)}}{N_R^{(k)} + N_U^{(k)}}$
    Set: evidence in BN agreement node
    Get: cost $c_t$ through BN forward pass
    Set: error term as $\delta \leftarrow c_t$
    Update: $E_c(s_t;\theta_c) \leftarrow E_c(s_t;\theta_c) + \delta$
    Update: $E_a(s_t, b_t;\theta_a) \leftarrow E_a(s_t, b_t;\theta_a) + \delta$
end for
```

and reward are used when the agent interacts with the external environment, and the terms belief and cost are used when the agent interacts with its own internal environment. The term belief must be intended as a precursor of the action which is not performed in the real environment, whereas the cost must be intended as cognitive cost of having a specific belief in the current state. The state-belief function returns the utility of a given belief b_t in a the current state s_t and is defined as $Q(s_t, b_t)$. The beliefs are taken from the set $B = \{b_0, b_1, ..., b_n\}$ which has the same cardinality of A.

Here the agent is following a softmax policy π that gives more weight to beliefs with higher utility. The raw belief vector returned from the state-belief function is passed through a softmax function and squashed into a multinomial probability distribution:

$$P\{b_t = b | s_t = s\} = \frac{e^{p(s,b)}}{\sum_k e^{p(s,k)}} \tag{6.1}$$

In a standard agent the critic is a value function $V(s_t)$ which estimates the utility of taking a specific action a_t in a specific state s_t based on the reward at s_{t+1}. The actor is a value-action function $Q(s_t, a_t)$ that returns the utility of producing an action a_t given the current state s_t. The presence of an internal environment adds one more factor to take into account, namely the cost c_t returned by the internal environment. The cost is a scalar estimated through an internal model $M(s_t, b_t, s_{t+1})$. The cost is summed to the reward during the update step through the following equation:

$$V(s_t) \leftarrow V(s_t) + \alpha\big[r_t + c_t + \gamma\, V(s_{t+1}) - V(s_t)\big] \tag{6.2}$$

Where γ is the discount factor and α is the learning rate. To update the actor the following equation has been used:

$$\delta_t = r_t + c_t + V(s_{t+1}) - V(s_t) \tag{6.3}$$

Equation 6.2 and 6.3 must be reviewed based on the particular case faced here. We are interested in trust-based learning of object-label pairs without any external reward. This problem can be considered a particular case of the most generic one described above. The agent starts from an initial state s_0, representing a particular object-label pair acquired from the external environment. At s_0 the agent estimates a belief b_0 using the state-belief function $Q(s_0, b)$. Moreover it acquires the suggestion of the informant i_0. Both b_0 and i_0 correspond to one of two states: accept or reject. The belief and the informant suggestion are then passed to the internal environment that returns a cost. The cost is used to update both the critic and the actor function. It must be pointed out that in the specific case considered here the reward term is always equal to zero because there is no feedback from the external environment. Taking into account all these considerations it is possible to rewrite Equations 6.2 as follows:

$$V(s_t) \leftarrow V(s_t) + \alpha\, c_t \tag{6.4}$$

The error estimation of Equation 6.3 reduce to:

$$\delta_t = c_t \tag{6.5}$$

From the two Equations 6.4 and 6.5 it is clear that the cost c_t is a crucial factor for updating the system. Moreover it is clear that the goal of the agent is no more maximising the expected reward but minimising the cost. The contribution of the informant can be considered as the contribution of an external policy μ. The policy μ can be optimal/sub-optimal in case of a reliable informant, or adversarial in case of an unreliable informant. The agent should converge to the optimal policy π^* discriminating between the informants and using as much as possible the advice from the reliable source.

6.2.4 The ERA architecture as function approximator

The actor and the critic components of the reinforcement learning mechanism are based on two functions, the value function $V(s)$ and the state-belief function $Q(s, b)$. Both of them can be represented by a lookup table or with a function approximator (e.g. a neural network). However, the use of a lookup table is problematic in large state spaces due to combinatorial explosion, and function approximators are not guaranteed to converge to stable policies. Here, a compromise between the two approaches is proposed and the ERA (Morse et al., 2010) is used as function approximator.

The flexibility of the ERA permits the arrangement of the different components of a unit in many ways. For instance, we are dealing with a state space of image-label pairs that we can represent as a unit composed of a vision and a vocabulary module. The ERA has been described in Chapter 3 so here its not described any further. However, it may be useful to briefly review the main passages necessary to train this architecture:

- *Stabilisation*. The SOMs in the ERA unit must be prestabilised presenting randomly generated inputs which cover a wide range of values (Morse et al., 2010). This step consists in presenting random RGB pixel values to the vision module. The effect is to adjust the centroids in the SOM so to reach a stable configuration that allows clusterising the input space.
- *Learning*. The learning phase in ERA consists of strengthening or weakening the connections between two Best Matching Units (BMUs). This step is the core of ERA and it allows the network to converge to an equilibrium in a few iterations.
- *Estimation*. Once the SOMs have been prestabilised and the learning phase has adjusted the connections it is possible to do a forward pass and estimate the values of the output units. For instance given a specific visual input and a label it is possible to find the BMU in the belief module. This is obtained through the weighted sum of

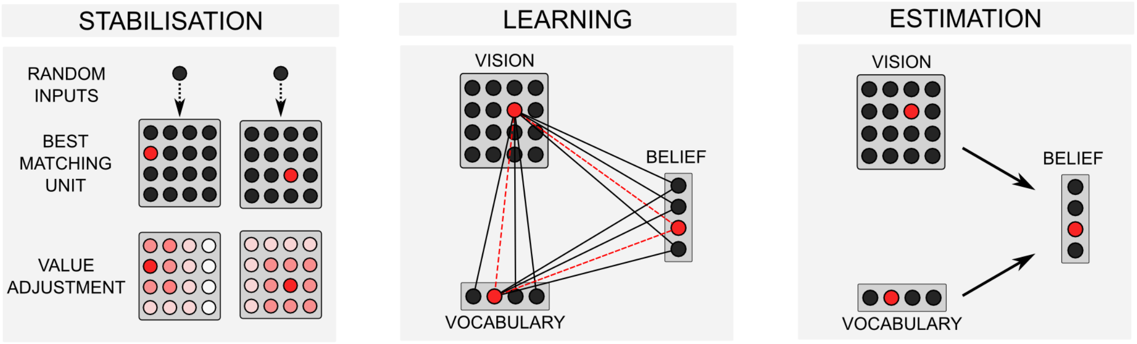


Figure 6.4: The stabilisation, learning and estimation phases in the BN architecture. In the stabilisation phase random inputs are used in order to stabilise the values of a SOM. In the learning phase the connections between active units are strengthened (positive Hebbian learning) or weakened (negative Hebbian learning). In the estimation phase the active units in a subset of the SOMs are used to estimate the activation value of inactive units.

the BMUs in the visual and vocabulary module.

6.2.5 Internal environment and Bayesian networks

The model is based on an inner environment that provides an estimated cognitive cost. The representation of the internal environment is composed of two modules. The first module (upper) estimates the cost of having a specific belief in a specific state and directly project to the critic. This module represents the contribution of brain areas involved in the dopamine regulation with projections to the ventral striatum (e.g. anterior cingulate cortex and medial prefrontal cortex). As discussed in Chapter 2 these areas modulate a feedback loop to the critic, and they directly influence the trial-and-error learning process. The upper module is extremely important, because at each state it evaluates what to do with the incoming suggestions using the current belief, the confidence level, and a record of past interactions with the informant. The module assigns a high cognitive cost to suggestions given by unreliable informants and a low cost to suggestion given by reliable ones. Moreover, it can estimate when an informant is reliable based on the confidence level and the similarity between the internal belief and the incoming suggestion.

The second module (lower) is responsible for evaluating the current belief and it can inhibit or refine an outgoing action produced by the actor. In biological terms this module represents the influence of areas involved in motor control and inhibition (e.g. cerebellum). In the particular case faced here the lower module does not have an active role because the agent does not physically interact with the external environment.

The upper module of the internal environment has been formalised in terms of causal relationship between random variables and modelled as a BN similarly to the method discussed in Chapter 4. As discussed before the use of a BN is in line with the theory-theory framework and is a probabilistic representation of how children reasons (Gopnik et al., 2001, 2004; Gopnik & Meltzoff, 1997). The role of the BN is to filter the suggestion given by the informant based on the belief of the agent. The BN incorporates for each informant a reputation distribution that can be considered separate from the reinforcement mechanism. Using a BN is convenient because it simplifies the representation of the relationship between different discrete random variables. The analysis is limited to the case of finite and countable values with discrete random variables. Capital letters indicate random variables and small letters indicate the possible state or event of the variables. For example, the random variable X is discrete and it can assume four states $\{a, b, c, d\}$. First of all it is considered a Bernoulli random variable F representing the agent confidence. The random variable F can take the value 1 (knowledgeable) and value 0 (non-knowledgeable). The probability distribution of F is estimated at each time step through the distribution B obtained from Equation 6.1. There are two possible edge cases. First, B is a uniform distribution and the agent is not knowledgeable at all. Second, one of the states has probability 1 meaning that the agent is completely sure about its own belief. It is possible to obtain the probability distribution of F using the normalised entropy estimated from each one of the N states of B, as follows:

$$\begin{aligned} P_F(0) &= -\frac{1}{N} \sum_{i=1}^{N} P_B(i) \log P_B(i) \\ P_F(1) &= 1 - P_F(0) \end{aligned} \tag{6.6}$$

In a similar way it is defined the Bernoulli random variable I representing the reputation of the informant. Here it is assumed that the agent saves in the long term memory a separate distribution for each informant, and that this memory is accessible from the internal environment. The random variable I can take the value 1 (reliable) with probability p and value 0 (unreliable) with probability $q = 1 - p$. The random variable is initialised with a uniform distribution $p = q$ because it is supposed that the agent does not have any particular prejudice against a new informant. At each step the algorithm updates the distribution in accordance with the informant's suggestion, the agent belief and confidence.

Only when the agent is confident ($F = 1$) it updates the reputation distribution for that informant. The informant is considered reliable when the belief of the agent and the suggestion of the informant agree ($b_t = i_t$). If the suggestion and the belief disagree ($b_t \neq i_t$) the informant is considered unreliable. Defining N_R as the total number of time the informant has been reliable, and N_U as the total number of time it has been unreliable, it is possible to update the reputation distribution through MLE as discussed in Section 3.2:

$$\begin{aligned} P_I(1) &= \frac{N_R}{N_R + N_U} \\ P_I(0) &= 1 - P_I(1) \end{aligned} \tag{6.7}$$

The agent confidence, the informant reputation and the agreement are integrated in the BN. The three nodes are connected with the cost node, that is modelled as a multinomial random variable C with three states (negative, null, positive). A graphical representation of the BN is shown in Figure 6.5.

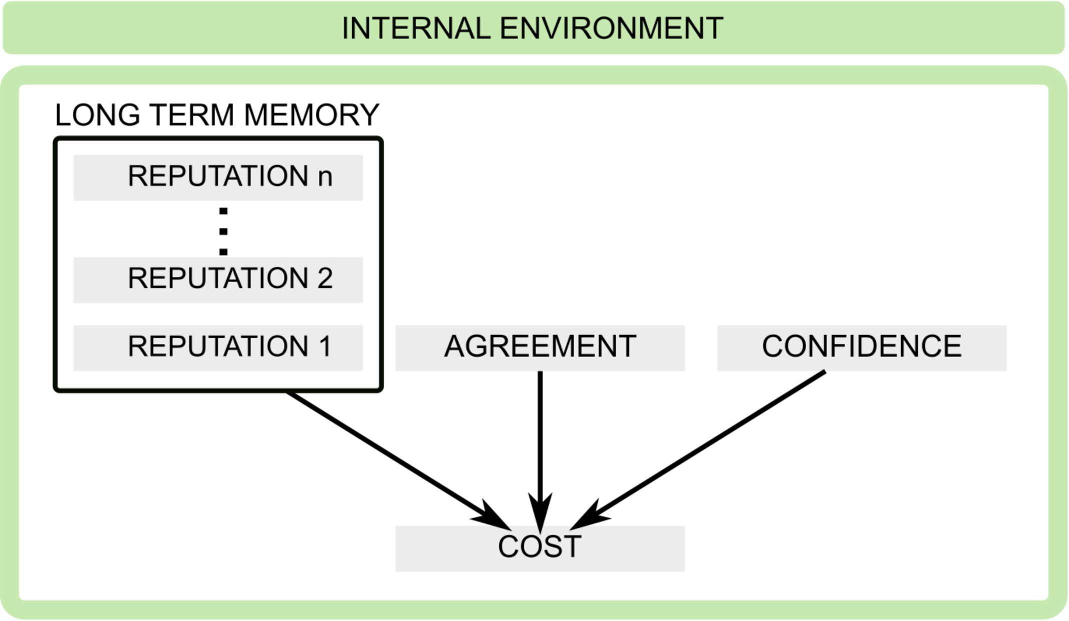


Figure 6.5: The Bayesian network used in the internal environment. For each informant a reputation distribution is kept in a long term memory storage. The reputation, agreement and confidence distributions are used to estimate a measure of cost.

In this section it has been showed how the probability distributions of the parents node in the BN can be estimated, but it was not showed how the conditional probability table associated with the cost node has been defined. Before discussing this point it is necessary a premise. The model is based on the assumption that it is expensive from the cognitive point of view to acquire information that could be inexact or inaccurate. To better understand this point we can consider this particular situation. The agent and the informant have different beliefs, the agent is not confident and the informant is reliable. In this case for the agent there is a high cognitive cost in keeping its own belief, because it is different from the opinion of a knowledgeable informant. The posterior distributions of the BN in this particular case are represented in Figure 6.6. In the same situation the suggestion of a non-knowledgeable informant will not provide any useful information, leading to a null cognitive cost. The BN outputs of this second case are represented in Figure 6.7. The conditional probability table and the cost associated to each discrete condition are summarised in Table 6.1. These assumptions are supported by a consistent literature. Vanderbilt et al. (2011) showed how 4 and 5-year-olds prefer advice from reliable informants when looking for a hidden reward. Fusaro et al. (2011) observed how pre-school children prefer to learn the names and functions of unfamiliar objects from the more accurate informant. Jaswal & Neely (2006) observed the same phenomenon in the interaction with peers instead of adults in 4-year-olds. The hypothesis that children can collect statistical information for tracking the reliability of informants is in line with recent research (Kushnir et al., 2010) that showed how young children can use statistical information, particularly a violation of random sampling, to infer the preferences of an adult for certain type of toys. This result was observed also in 20-month-old infants, confirming that it is an early developmental mechanism. This research seems to be consistent with the internal environment proposed in this chapter and can be considered its theoretical foundation.

6.3 Experiments

An experimental test is always necessary in order to validate a model. However here we are dealing with complex behaviours that rely on multiple cognitive abilities and we must be careful in comparing results obtained on human subjects with results obtained through a computational model. It is possible to deal with this issue focusing on a well-defined subset of scientific hypotheses, and using an experiment to prove or disprove those hypotheses. Here those hypotheses are summarised in three points:

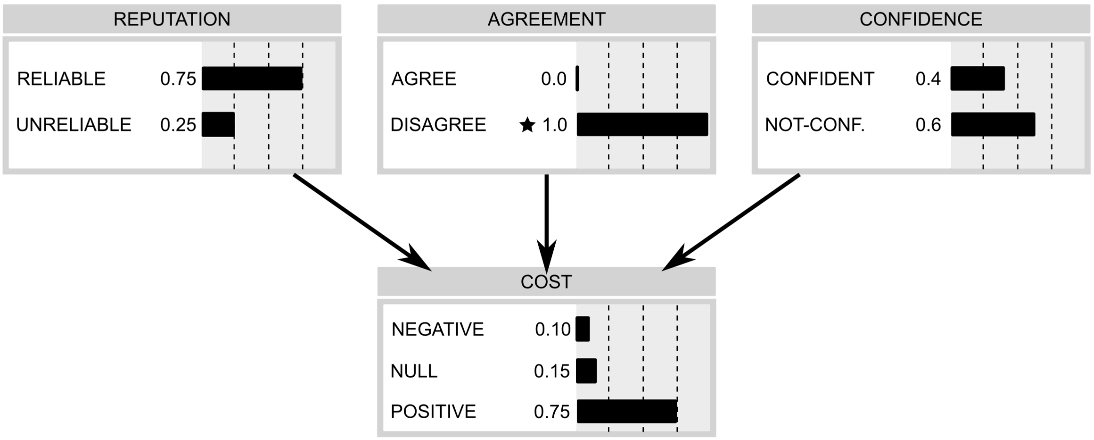


Figure 6.6: The Bayesian network with the associated node distributions when a reliable informant suggestion is in not in accordance with the agent belief and the agent is not confident. The star symbol used in the agreement node represent an evidence (agree). The cost sampled from the posterior distribution will be with 15% probability null, with 10% probability negative and 75% positive.

1. In accordance with biological observations, brain areas involved in trust are part of an actor-critic mechanism and they modulate an internal feedback directed to the critic.
2. Trust and reputation influence learning, meaning that information provided by reliable sources is kept and information provided by unreliable sources is discarded.
3. An immature ToM is linked to a bias in the causal model of the internal environment and it influences the temporal difference error produced by the critic.

In order to validate those hypotheses a developmental robotics approach has been used, replicating the experiment of Koenig & Harris (2005). This experiment has been chosen because it covers the three aspects reported above, and because it highlights particular aspects of children reasoning in trust estimation In this experiment children integrated the advice of a reliable and unreliable informant who presented conflicting names for novel objects. The results reported in Koenig & Harris (2005) show that 3 and 4-year-olds can discriminate between reliable and unreliable informants, but only the 4-year-old group was able to internalise the name given by the reliable source. It is reasonable to hypothesise that 3-year-olds may lack tools needed to explain false statements produced by the unreliable informants. It is necessary to stress the fact that 3-year-olds can successfully identify the unreliable source, meaning that their difficulty was not in differentiating between the two

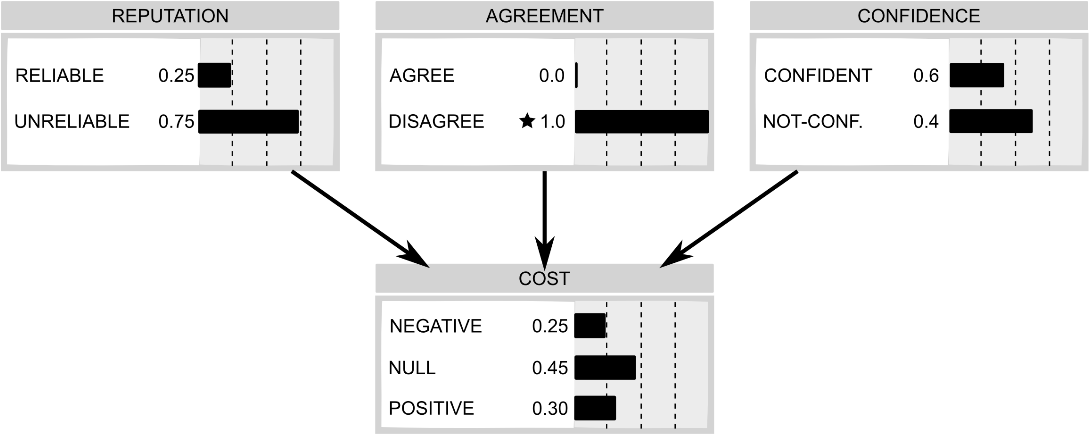


Figure 6.7: The bayesian network with the associated node distributions when an unreliable informant suggestion is not in accordance with the agent belief and the agent is not confident about the current state. The star symbol used in the agreement node represent an evidence (disagree). The cost sampled from the posterior distribution will be with 45% probability null, with 25% probability negative and 30% positive.

informants. Here in addition to replicate the results obtained on the 4-year-old group the model has been used to understand which factor may have influenced the answers of the 3-year-olds. The technical details are discussed in the next section.

6.3.1 Methods

To replicate Koenig & Harris (2005) the model has been embodied in the iCub humanoid robot (Metta et al., 2008). The iCub is an open-source robotic platform, widely used by the research community that has been described in Chapter 3. The iCub has 53 degrees of freedom, the head and the eyes can move independently in a bio-plausible way and can track moving targets in the environment. Initial iCub experiments were carried out in the open source iCub simulator (Tikhanoff et al., 2008), and then adapted on the physical robot platform. The vocal commands to the robot were interpreted using Pocketsphinx (Huggins-Daines et al., 2006) an open-source real time language recognition system. The learning phase was based on simple grammar rules that allowed the interpretation and segmentation of the commands. The robot has two RGB cameras (reolution 320×240 pixels) that have been used to capture the scene. To reduce the noise, a subframe of size 50×50 pixels (fovea) was given as input to the visual SOM. In the experiments the visual SOM has been prestabilized with random RGB values in the range [0,255] as recommended by Morse et al. (2010). The focus of attention of the robot was defined at

reliable			
agree		disagree	
confident	unconfident	confident	unconfident
Negative	**Negative**	**Positive**	**Positive**

unreliable			
agree		disagree	
confident	unconfident	confident	unconfident
Negative	**Null**	**Negative**	**Null**

Table 6.1: Conditional probability table of the cost node in the BN of the internal environment. The first three rows of both upper and lower tables represent the parent node values and the last row (in bold) the value assumed by the cost. The opinion of the unreliable informant is discarded (null cost) when the agent is unconfident, and it is acquired (negative cost) in case of confidence and disagreement.

reliable			
agree		disagree	
confident	unconfident	confident	unconfident
Negative	**Negative**	**Positive**	**Positive**

unreliable			
agree		disagree	
confident	unconfident	confident	unconfident
Negative	**Negative**	**Positive**	**Positive**

Table 6.2: Conditional probability table of the cost node in the BN of the internal environment in case of an immature ToM. The first three rows of both upper and lower tables represent the parent node values and the last row (in bold) the value assumed by the cost. The opinion of the unreliable informant is taken into account with the same weight of the reliable informant.

runtime identifying relevant colours in the environment. To have an effective interaction with the user it is important to take into account pointing and gazing (Rouanet et al., 2013). For this reason the focus of attention of the informant has been estimated through the Deepgaze library with the method described in Chapter 5.

To model the 3-year-olds behaviour it has been hypothesised that the lack of ToM had an influence on the cost estimation of the internal environment. This bias has been modelled introducing a different conditional probability table which is shown in Table 6.2. This table is based on the assumption that 3-year-olds cannot use information about informants' past inaccuracy when deciding whom to trust in the future. This assumption is in accordance with the hypothesis of Koenig & Harris (2005) and it should be considered an attempt to investigate the underlying mechanism involved in the 3-year-olds bias.

Since the architecture is based on a probabilistic model the robot can give different answers every time. For this reason the same procedure has been repeated 25 times using the standard architecture, and 25 times using the modified version which implemented the ToM bias in the cost node (see Table 6.2). Each time the parameters of the model were randomly initialised and the answers given from the robot were recorded. The initial randomisation mimics the interaction of different children with the two informants. Like in Koenig & Harris (2005) an independent sample t-test (two-tailed) has been used to validate the results. Moreover, has been made the same statistical assumption that a mismatching label in the explicit judgement trial corresponded to a chance score of 1, and that a mismatching label in the endorsement trial corresponded to a chance score of 2. The experiment was divided in four different phases: object learning, familiarisation, explicit judgement, endorsement. All of these phases but the object learning are part of the original experiment of Koenig & Harris (2005). In the following sub-sections each phase is described.

Object learning

Unlike children the robot started as a blank slate, for this reason it was necessary to give it a basic knowledge of the world and in particular of common objects. This phase has been called object learning. This pre-learning period is crucial for species not born at an advanced level of neural and motor maturation. In infants learning starts in the very first hours of life and it allows the newborn to rapidly learn vital cues from the mother's body, and to identify voices, faces and odours (Nagy & Molnar, 2004). During the object learning the experimenter showed to the robot six known objects (ball, cup, book, shoe, dog, chair). These objects are the same used in Koenig & Harris (2005). Here the experimenter is considered as a caregiver having a high reliability. To model the high reliability the reputation distribution of the caregiver was initialised as not uniform (99% reliable). This is reasonable, since during the infancy the caregivers are considered reliable source of information. The object learning is illustrated in Figure 6.8.

Familiarisation

In the original experiment a set of the familiar objects was presented to the child. Two informants gave different names to those objects. One informant always gave the right name, whereas the other informant always gave the wrong name. In the experiment each

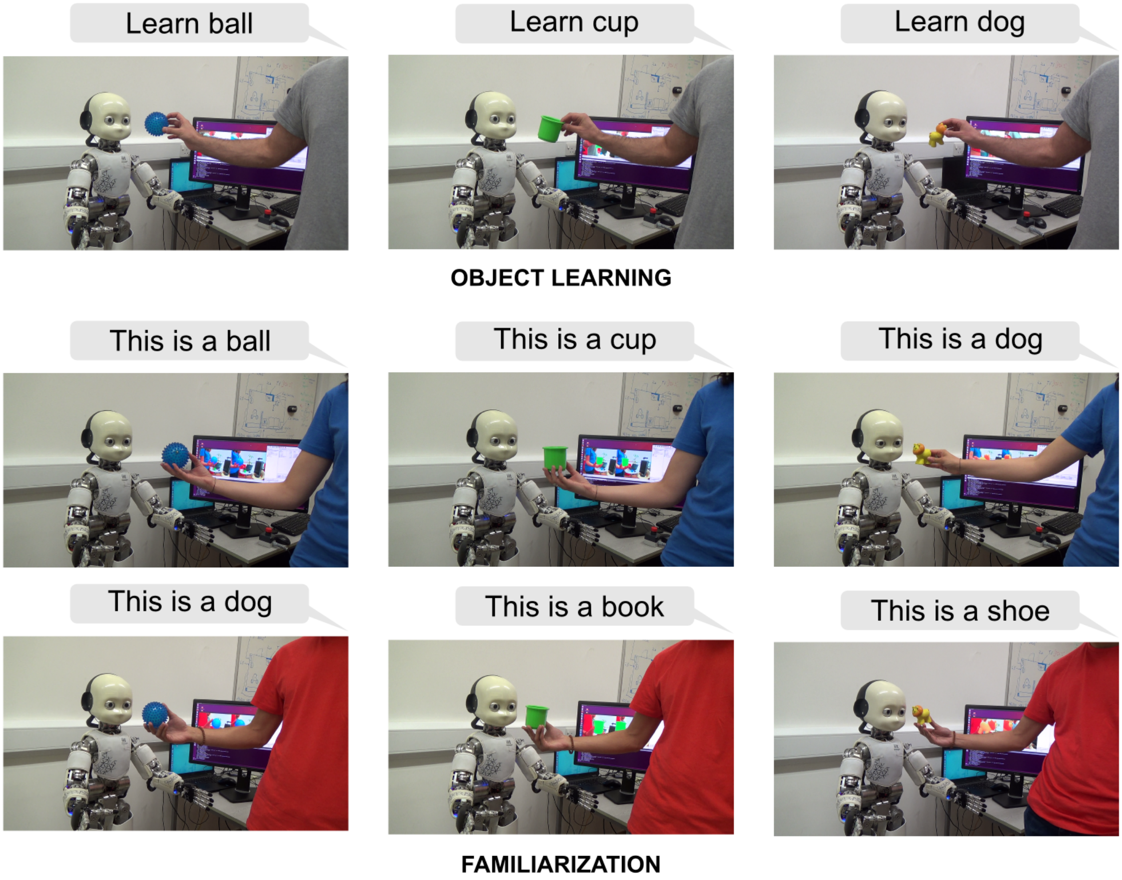


Figure 6.8: The object learning and familiarisation phases. In the object learning phase the caregiver (grey t-shirt) teaches the names of common objects to the robot. The image shows three of those objects: ball (blue), cup (green) and dog (yellow). In the familiarisation phase both the reliable and unreliable informants show familiar objects to the robot. The reliable informant (blue t-shirt) always gives the correct name to the object. The unreliable informant (red t-shirt) always gives a wrong name.

session was initialised with one of the two informants (randomly selected). The informant showed to the robot a sub-set of the objects learned during the object learning and then pronounced the name of the object. Like in the original experiment one of the informant was always right, the other always wrong. During the familiarisation the cognitive model passed through all the different steps and the reputation distribution of each informant was updated using MLE (Equation 6.7). The familiarisation is illustrated in Figure 6.8.

Explicit Judgement

After familiarisation the experimenter asked the children "One of these people was not very good at answering questions. Which one was not very good at answering questions?". In the experiment this situation was replicated. The experimenter asked the same question following a simple grammar rule and the answer of the robot was recorded. As in the

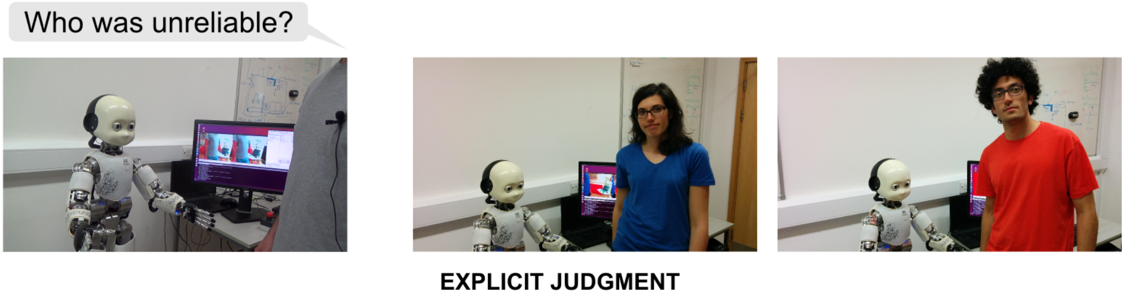


Figure 6.9: The explicit judgement phase. the caregiver (grey t-shirt) asks to the robot which informant was not god at answering questions. The robot recalls from the long term memory the reputation distribution returning the name of the unreliable informant.

original experiment the answer was coded as correct only if the robot appropriately indicated the name of the unreliable informant. The robot answers were produced accessing the long term memory, and sampling from each K distribution $P_F^{(k)}$ a random value which represented the reliability of a specific informant. If only one informant was considered reliable, the informant name was produced by the robot. When both informants were considered reliable (or unreliable) at the same time, a random name was sampled. An illustration of this phase is shown in figure 6.9.

Endorsement

In the endorsement trial the two informants gave different labels for a new unfamiliar object. For example, in reference to the first novel object one informant said "That's a mido", the other said "That's a Ioma". Based on the informants feedback the children had to choose a name for the novel object. In the experiment this situation has been replicated. The two informants showed a new object to the robot and then presented the name using the rule "That is a [Object Name]". In a second phase the researcher showed the same object to the robot asking "What is this?". To record the answers the same procedure of Koenig & Harris (2005) has been used. The answer of the robot was scored from 0 to 3, depending on how often (out of three trials) it indicated the name given from the reliable source. The answer was considered incorrect if the robot supplied the name given from the unreliable source or a different name. In the cognitive architecture this step corresponds to a forward pass in the actor ERAnetwork. Giving as input the object color in the vision unit the corresponding output in the vocabulary unit was estimated. The endorsement phase is illustrated in figure 6.10.

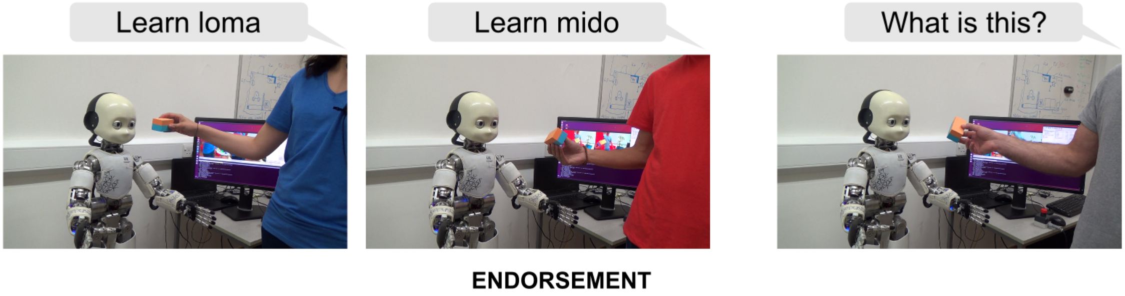


Figure 6.10: The endorsement phase. Both the reliable (blue t-shirt) and the unreliable (red t-shirt) informants suggest a name for a new object. After the interaction the caregiver (grey t-shirt) asks for the name of the object.

6.3.2 Results

The results for the judgement trial are summarised in Figure 6.11, whereas the results for the endorse task are summarised in Figure 6.12. The results reported in Koenig & Harris (2005) showed that 4-year-olds correctly discriminated the reliable and the unreliable informant. In particular they could select the reliable source in a decision making task and correctly estimate the beliefs. In the explicit judgement phase the children correctly identified the unreliable informant *M=0.83, STD=0.30, t(17)=4.76, p<.001*. Similar results have been obtained with the iCub robot *M=0.76, STD=0.43, t(25)=2.59, p<.01*. In the endorse trial the 4-year-olds could associate the label given by the reliable informant above chance, *M=2.11, STD=1.08, t(17)=2.40, p=.028*. Similar results have been obtained using the iCub robot *M=2.0, STD=0.75, t(25)=2.82, p<.01*.

The results reported in Koenig & Harris (2005) regarding the 3-year-olds showed that children could identify the unreliable informant in the explicit judgement phase *M=0.74, STD=0.37, t(20)=2.91, p<.01*, however they were more prone to accept advice from both informants in the endorsement test *M=1.19, STD=1.03, t(20)=1.38, ns*. Similar results were obtained using the model with the bias in the conditional probability table of the cost node. In the judgement trial the robot identified the unreliable informant above chance, *M=0.76, STD=0.42, t(25)=2.63, p<.01*, and in the endorsement test it accepted advice from both informants *M=1.4, STD=0.98, t(25)=-0.49, ns*.

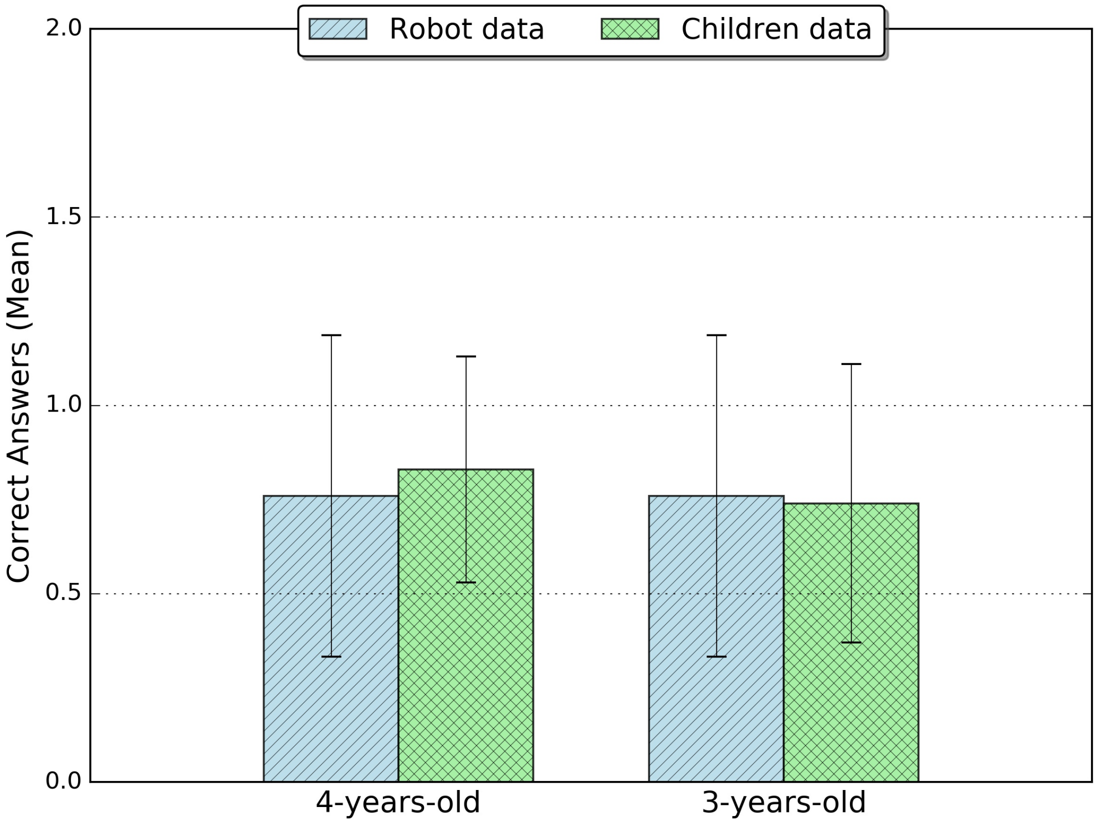


Figure 6.11: The results obtained in the explicit judgment task (mean and standard deviation) in two consecutive trials. In this task children and robot have to identify the unreliable informant. Both 3-year-olds and 4-year-olds performed significantly above chance meaning that they could identify the unreliable informant. The robot performed similarly to children and above chance level.

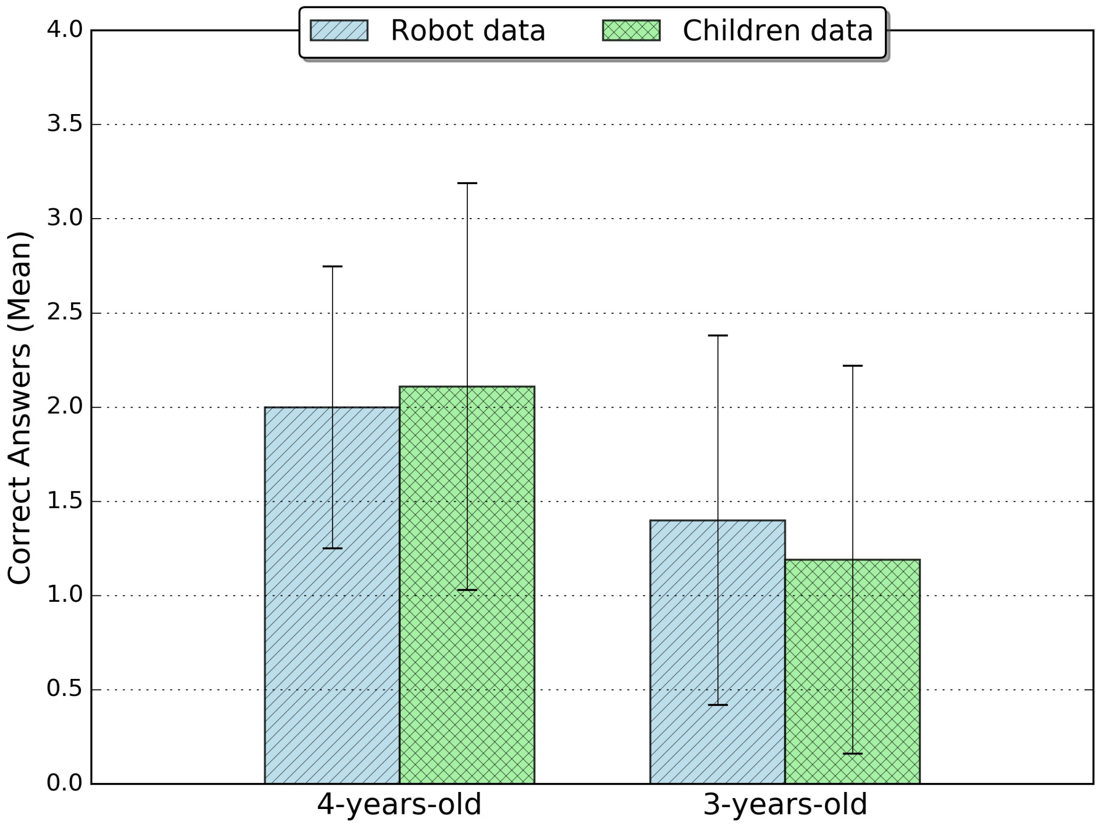


Figure 6.12: The results obtained in the endorse task (mean and standard deviation) in three consecutive trials. In this task children and robot have to provide the same label given by the reliable informant to an unknown object. Only 4-year-olds performed above chance providing the right label. The robot performed similarly to children in both conditions.

6.4 Discussion

Here it is important to review the experimental results and discuss what we can take away from these results. In this sense it may be useful to restate the three hypotheses reported in Section 6.3. First, in accordance with biological observations, brain areas involved in trust are part of an actor-critic mechanism and they modulate an internal feedback directed to the critic. Second, trust and reputation influence learning, meaning that information provided by reliable sources is kept and information provided by unreliable sources is discarded. Third, an immature ToM is linked to a bias in the causal model of the internal environment and it influences the temporal difference error produced by the critic.

Overall it has been observed that the model supports trust-based learning, and that it can correctly associate object-label pairs in absence of biases. The results obtained by the unbiased 4-year-old group in the endorse task confirm that both children and robot were able to identify the objects using the labels given by the reliable informant. The model is then able to use the internal cost as a discriminant factor to drive learning. Those results support the first and second hypotheses and confirm the representational power of the internal environment. The use of an internal environment is not only plausible from the biological point of view, but it is also a flexible design choice that allows expanding the model in a modular way. For instance, when future research will outline the role of other brain areas it will be possible to further expand the BN and include those additional factors.

The results related to the third hypothesis shed light on the way the ToM bias can affect children reasoning. To discuss this hypothesis we have to focus on the structure of the BN. The reputation has been modelled as a Bernoulli distribution that is stored in a long term memory. In the causal model the reputation affects the cost estimation in one direction, meaning that it is possible to retrieve this information without computing a posterior distribution. On the opposite, estimating the cost requires to take into account reputation, agreement and confidence. Comparing the results obtained in the explicit judgement and in the endorse tasks we can understand how those factors are related. In the explicit judgement task the ToM bias is not directly affecting the way children recall memories about informants' reputation, since 3 and 4-year-olds have similar performances. In the endorse task is required to associate a name to a given object, choosing the label given by one of the informants. To take this decision a measure of cost must be computed. In the experimental setting a bias has been observed, in both children and robot, when this operation was performed.

6.5 Conclusions

In this chapter a cognitive architecture for trust-based learning has been proposed. The architecture is based on psychological and biological observations. Using a developmental robotics approach a previous experiment by Koenig & Harris (2005) was replicated, and similar results obtained. One of the most remarkable outcomes is that a computational replica of a biological substrate leads to an efficient trust-based learning, and that it can capture subtle differences between two groups of children. The model is based on the assumption that acquiring information from unreliable informants has a high cognitive cost. This cost is computed by a BN that takes into account the reputation of the informant, beliefs agreement, and confidence of the agent. It has been showed how introducing a bias in the way the information is processed in the BN leads to the same results obtained within the 3-year-old group. The 3-year-olds are able to articulate the distinction between speakers who are willing to help and those who are not, but they fail to put this understanding into practice. This bias is connected with the way informant's beliefs are perceived and then used by the child and it is related to an immature ToM.

Despite those achievements the proposed model can be further improved, other causes may play an important role in trust based learning. First of all, different forms of prejudice may interfere with the reputation estimation causing problems in the decisional process of the child. Prejudice could be modelled using a biased prior distribution for the reputation node, which can be initialised based on previous experiences. Secondly, unlike the experiment considered here (Koenig & Harris, 2005) in real world scenarios an informant may be reliable and unreliable at the same time. In literature there is not enough work on this subject and it is not clear how children would behave when interacting with an inconsistent informant. In addition we should not forget that trust has a social dimension. Group interaction and sensitivity to consensus may also have a main role in making children more or less vulnerable to misleading information. For example, children may make predictions about an informant on the basis of that person's membership in a particular group (Jaswal & Neely, 2006). In future work would be possible to extend the experiments and include additional factors. For instance, as discussed in Chapter 5 gaze and head pose has a primary role in trust. It is a well know fact that autistic children have deficits in interpreting others' belief and in gaze processing. Using a robotic platform it would be possible to investigate how gaze influences trust evaluation in the autistic population.

In conclusion, the results obtained are promising but further research in developmental psychology and cognitive modelling should be carried on in order to address the unanswered questions. Additional considerations are presented in Chapter 8.

Chapter 7

Trust in Human-Robot Interaction

7.1 Introduction

In parallel with the development of a cognitive architecture for trust-based learning the author of this thesis also collaborated with another PhD student Debora Zanatto in planning the HRI experiments. Zanatto was involved in the experimental design, data collection and analysis. The role of the author instead, was to define the setup of the experiments based on the capability of the robots and to develop a complete system architecture for managing the different phases of the experiments. The architecture in its final form was a complex system, based on a modular approach that is described in Section 7.3. Here, the aim of the experiments and the results achieved are briefly introduced.

In this series of experiments a perspective shift is applied in comparison with the rest of the thesis. In this chapter the robot is considered the trustee and the person is the relying agent (Falcone & Castelfranchi, 2001). Several factors have been investigated. Firstly, whether eye gaze had a rule in the interaction with non anthropomorphic robots. Secondly, how the perception of anthropomorphism can influence the capability to accept recommendation from a robot. To investigate this factor, different robotics platform have been used: iCub, SCITOS G5, Baxter, NAO, Pepper (see Figure 7.1). Finally, how the robot behaviour and voice can influence the participants' choice. To test these effects we used robots with different levels of anthropomorphism, manipulating their behaviours in several experimental conditions, some of these are reported below:

1. *social VS non-social gaze*: with mutual gaze between the robot and the participant, or with the robot just looking at the stimuli.

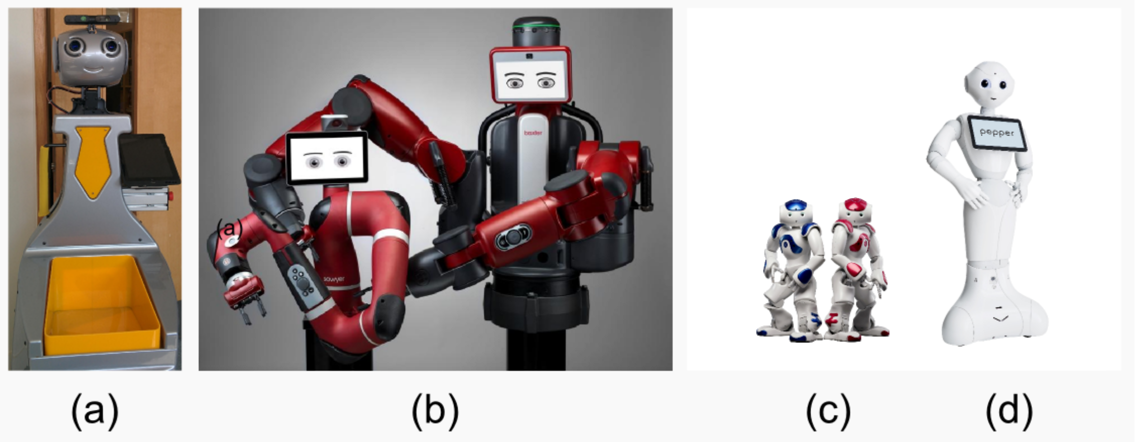


Figure 7.1: The robots used in the HRI experiments. (a) SCITOS G5. (b) Baxter (right). (c) two NAO robots. (d) Pepper.

2. *social VS non-social behaviour*: with the robot using its body language to communicate with the participant, or with the robot just standing still.
3. *natural VS synthetic voice*: with the robot using a natural human voice or a synthetic one.
4. *anthropomorphic VS non-anthropomorphic*: the embodiment of the robot was manipulated using different platforms (iCub, NAO, Pepper, Baxter, Scitos G5).

In the next section the methods and results obtained in the first experiment are reported, with the aim of investigating how the appearance of the robot influenced the choice of participant in a price judgement task.

7.2 Experiment

In HRI the main contributors to the development of trust were identified as depending on the robot performance and its attributes, whereas environmental factors played a less important role (Hancock et al., 2011). However Hancock et al. (2011) recommended the collection of novel empirical data to better understand the role of the robot attributes in building trustworthy human-robot interaction. In line with this recommendation, the first step of the project was the preparation of a psychological experiment to study how humans perceive robots.

We investigated the perceived credibility of statements made by robots, hypothesising that people would be more likely to believe robots with humanlike characteristics than those that are less anthropomorphic. We also examined whether prior experience with a humanlike robot would lead people to extend this advantage to the less-anthropomorphic robot. A measure of credibility was provided by agreement on the pricing of objects, where participants negotiated with either a more (iCub) or less anthropomorphic robot (Scitos G5) that was engaged in more (using social gaze) or less-humanlike (fixed gaze) social behaviour. In the first experiment participants only interacted with the Scitos G5, in the second they interacted with the Scitos G5 only after having first interacted with the iCub. Results showed that the iCub was more credible than the Scitos G5, and was the only robot to benefit from the use of social gaze. It was also found that the credibility of the Scitos G5 was higher after participants were primed by prior exposure to the iCub.

7.2.1 Methods

In the first part of the experiment 15 participants between 18 and 30 years interacted only with Scitos G5. The robot initially described the object on the table and gave two price options to the participants. Participants selected their choice. The robot commented the initial selection, either positively or negatively. In case of negative feedback the participants had the opportunity to change their decision and make a final selection. During the experiment the robot could perform two different gaze behaviours (social and nonsocial). In the social gaze behaviour the robot looked first at the object on the table and then moved its gaze to the participant before starting describing the object. In the nonsocial gaze behaviour the robot looked constantly at the stimulus. The robot answer to each stimulus was provided from a randomized script, with the same pattern of responses provided to each participant, irrespective of their choices. In the second experiment 15 participants between 18 and 30 years interacted first with iCub and then immediately repeated the game with Scitos G5. All participants were psychology students with no reported direct experience of robots.

7.2.2 Results

Social interaction with the robot was quantified through analysis of trials in which the robot disagreed with the price choice of the participant. A change rate was calculated as the proportion of these trials where the participants changed their selection to agree with the robot, compared to those in which they stuck with their original decision. Change rates for

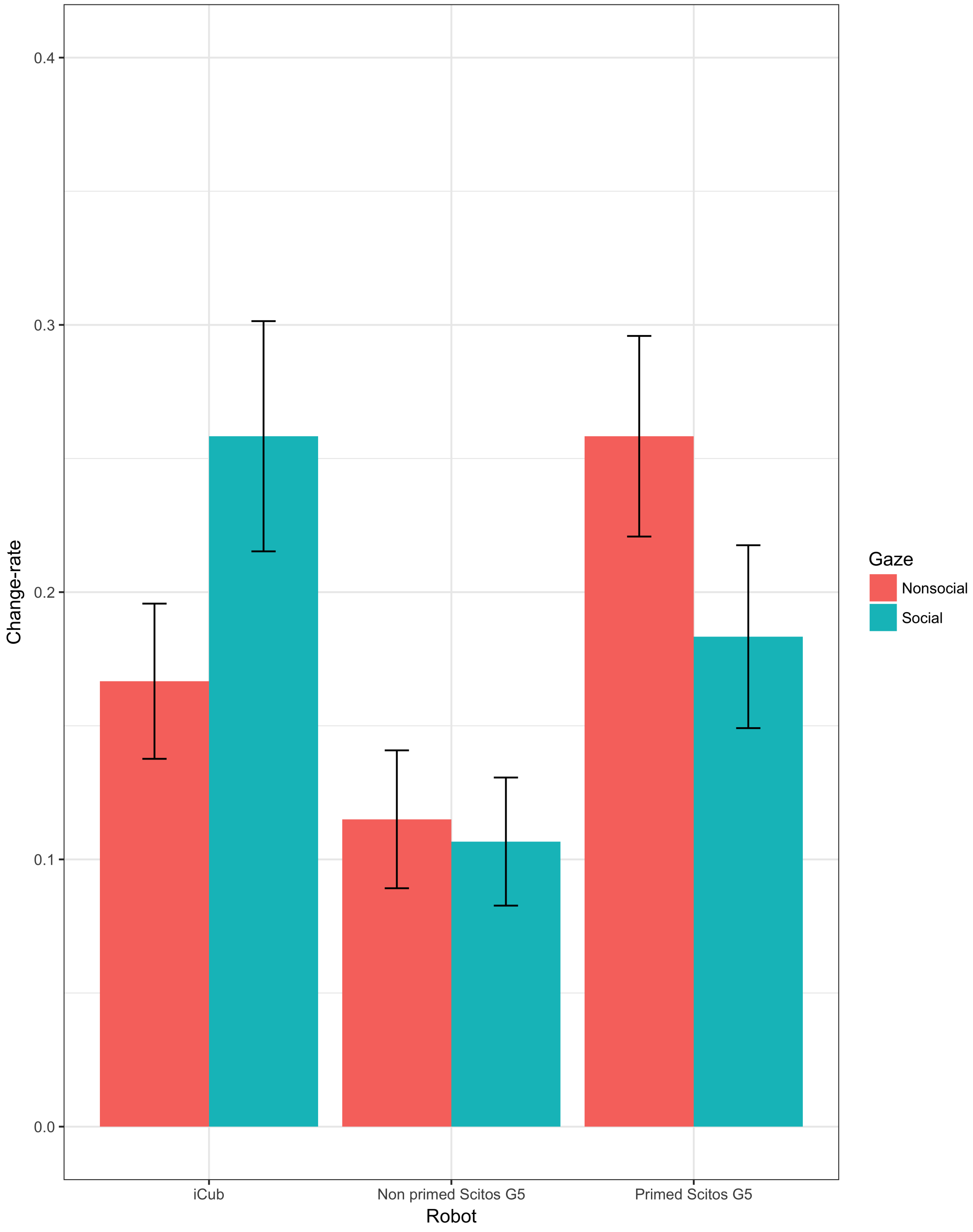


Figure 7.2: The results obtained in the experimental session.

iCub and the primed and nonprimed presentations of G5 are shown in Figure 2, presented for both the social and nonsocial gaze conditions. Three separate pairwise ANOVA analyses of change rate were conducted between each of the three robot conditions, each with factors of robot and gaze. The comparison between iCub and the nonprimed Scitos G5 revealed a significant effect of robot, (iCub=0.198 vs nonprimed Scitos G5= 0.112; $F(1,28) = 6.71$, $p = .008$), with participants changing their decisions more often with iCub than with nonprimed Scitos G5. A significant interaction between robot and gaze was also found ($F(2,28) = 8.13$, $p = .015$). Posthoc testing revealed that change rate was significantly higher ($p < .05$) for the social than for the nonsocial gaze condition in the iCub, but not the nonprimed Scitos G5 ($p = .775$). Comparisons between iCub and primed Scitos G5 did not reveal a significant difference in change rate between the two robots ($p = .48$), but did show a significant interaction between gaze behaviour and robot ($F(1,14) = 22.35$, $p = .000$). Posthoc testing revealed that change rate was significantly higher ($p < .05$) for the social than for the nonsocial gaze condition in the iCub, but not in the primed Scitos G5 ($p = .086$). There was no significant main effect of robot ($p = .48$). The final comparison between nonprimed and primed Scitos G5 showed a significant main effect of robot (nonprimed Scitos G5= 0.111 vs primed Scitos G5= 0.221; $F(1,28) = 8.99$, $p = .007$), where participants changed idea more often when Scitos G5 was primed by iCub than when it was presented in isolation. The main effect of gaze ($p = .068$), and interaction ($p = .14$) were both non-significant. The results are reported in Figure 7.2.

7.2.3 Discussion

The results of our study indicate that people are more socially engaged with robots when they are more-humanlike, similar to the findings of other studies (Fink, 2012). We found that participants were more likely to change their valuation of an object to agree with the humanlike iCub robot than with the less-anthropomorphic Scitos G5, when that robot was presented in isolation. Moreover, it was found that this change rate increased when the iCub engaged in a more-humanlike social gaze behaviour, compared to the use of a fixed gaze. Conversely, this social gaze behaviour had no effect on interactions with the Scitos G5, where the change rate was not significantly different from the use of a fixed gaze. This suggests that the credibility of anthropomorphic robots benefits from the automatic activation of social stereotypes that are normally absent during interactions with less-anthropomorphic variants. However, it would appear that once participants had

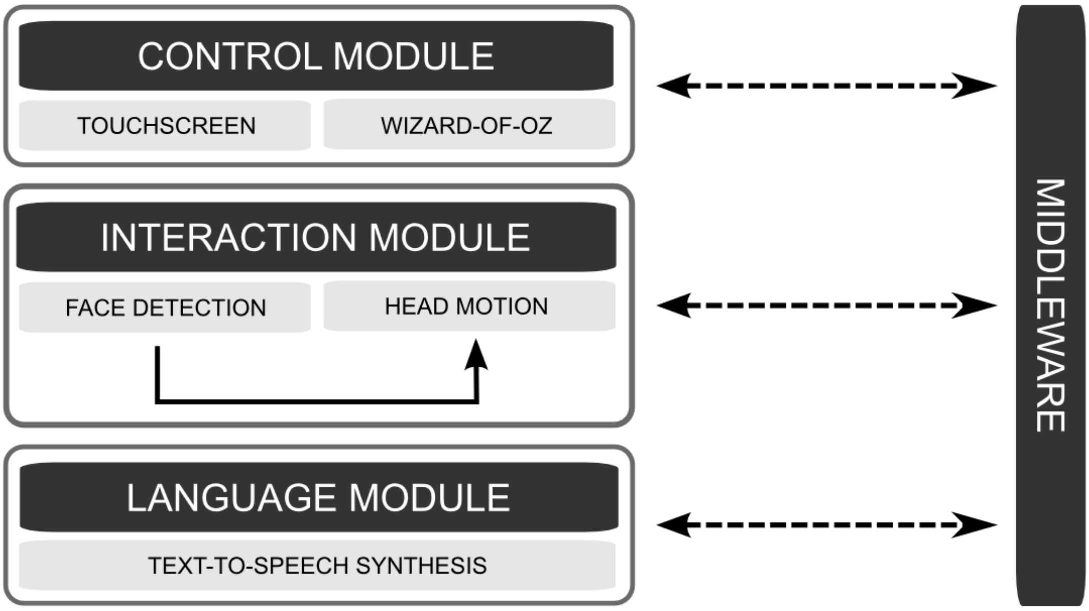
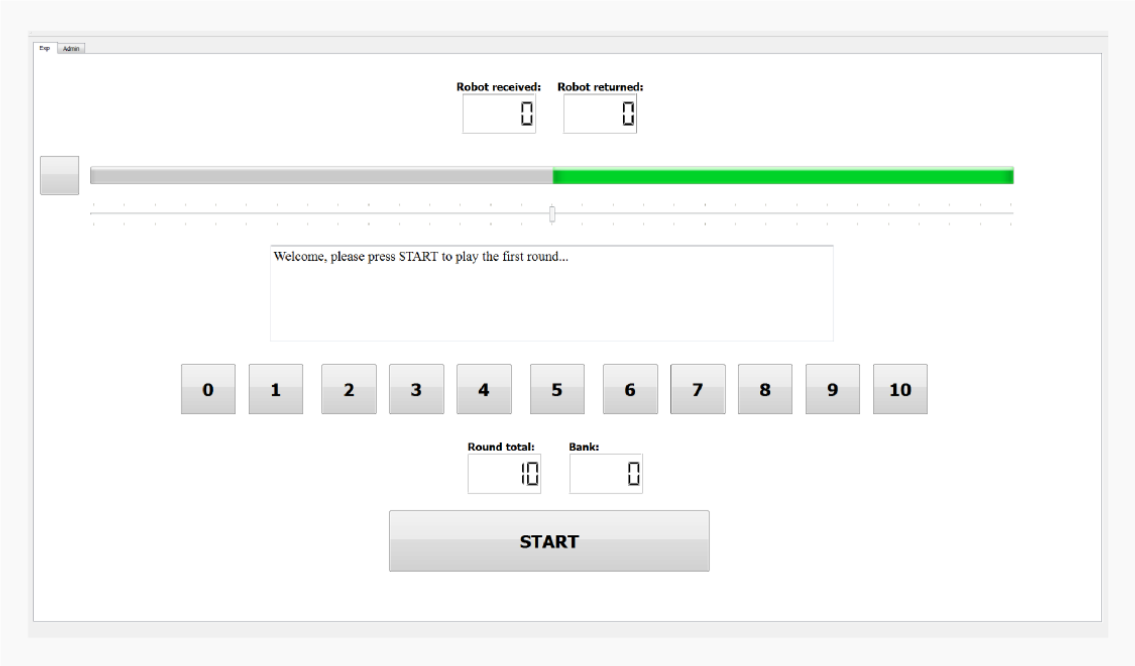


Figure 7.3: The architecture of the system used in the HRI experiments.

experience of an anthropomorphic robot they were willing to extend this classification, and the associated social benefits, to subsequent interactions with a less-humanlike robot. We found that participant's change rate with the Scitos G5 was significantly higher if participants had first interacted with the iCub robot. Also, once the less-humanlike Scitos G5 had been primed by the anthropomorphic iCub there was no longer any significant difference in change rate between these two robots. It should be noted however that even this anthropomorphically primed Scitos G5 did not benefit from the use of social gaze, probably due to the lack of congruence between this behaviour and the robot physical form. This finding has important consequences for the credibility of robots and our acceptance of them, e.g. in the increasing need and advocacy for the use of robots to assist the elderly in their homes (Burgoon et al., 2000). However, the functionality required in this environment often precludes the use of anthropomorphic physical forms, which hinders acceptance (Broadbent et al., 2009). Our study suggests that if users were first primed with anthropomorphic robots they would be more accepting of their less-humanlike, but more functional, robotic relations.



Robot received: 0
Robot returned: 0

Welcome, please press START to play the first round...

Round total: 10
Bank: 0

Figure 7.4: An example of GUI used in the HRI experiments. The interface is essential and is composed of a start button, investment buttons (from 0 to 10), score of both robot and participant, an energy bar representing the investment of the robot, and a central window that guides the user through the different phases of the experiment. In another window the researcher can set the XML file and the data of the subject.

7.3 System architecture

The system used in this research is based on a modular approach and it is inspired by the control system of the Kismet robot described by (Breazeal, 2004), and widely used in HRI. The system is based on a middleware and five modules: control module, face detection module, motion module, language module. Before using the system on the real platforms it was tested on the iCub simulator software suite, on the Choregraphe visual programming tool for NAO and Pepper robots, and on the ROS suite (Quigley et al., 2009) for the SCITOS. The code was developed in Python and C++ using open source libraries widely adopted in robotics. A graphical representation of the system is provided in Figure 7.3. In the following sections each module is described:

- *Control Module* this module was the core component of the system, and it aimed to engage action patterns via the activation of other modules. Although it may be possible to support semi-autonomous capabilities in this layer, this has been avoided, since the experimental setting required very specific pattern of activation. In the first phase a Wizard-of-Oz (Kelley, 1984) procedure has been used to control the robot. The procedure was based on a human confederate who remotely activated the different phases of the experiment.. This procedure used the information provided through an Extensible Markup Language (XML) file to interrogate the other modules and organise the robot actions. The XML file contained a description of the object presented during the experiments, the sentences to reproduce with the language module, and the order of presentation. In the second version of the experiment an automatic action engagement has been implemented in this layer. The second version included a series of phases triggered by the interaction of the subject with a Graphical User Interface (GUI) on a touchscreen (see Figure 7.4 for an example). For each session a video was recorded from the robot cameras and a log file containing the participants choices was saved for later use.

- *Interaction Module: Face Detection* the face detection sub-module has been implemented using OpenCV, an open source library for artificial vision. The algorithm used for detecting faces was the Haar feature-based cascade classifiers. This algorithm is an effective object detection method proposed by Viola & Jones (2001). It is a machine learning based approach where a cascade function is trained from a lot of positive and negative images. In the first phase it needs positive examples (images of faces) and negative examples (images without faces). Then it extracts features from these examples using Haar-features. Each feature is applied on all the training images, and for each feature the algorithm finds the best threshold which will classify the faces to positive and negative. The features with the minimum error rate are selected, which means they are the features that best classify the face and non-face images.

- *Interaction Module: Head Motion* The head motion sub-module is part of the interaction module and it allows the robots to move the head in the direction of the participant's face. This module is platform specific and it uses different low level library provided for each robot. The motion module is strictly connected with the face detection module and uses the coordinates provided through the middleware for centring the human face in the field of view of the robot.

- *Language Module* this module allows the robot to speak using a synthetic voice. To implement this module the Acapela text to speech synthesiser has been used. The Acapela synthesiser is an online engine that permits reproducing words from a text.

The text to speech system begins by carrying out a sophisticated linguistic analysis that transposes written text into phonetic text. The library is cross-platform and provides a wrapper for Python and C++. In the experiment the same voice (neutral male) was used to produce the sentences. An XML document contained the tree with all the phases of the experiment and all the sentences produced in each phase.

- *Middleware* The middleware allows the comunication between the modules in each robotic platform. To control the NAO the NAOqi APIs have been used. The NAOqi Framework is the programming framework used to answer to common robotics needs including: parallelism, resources, synchronization, events. This framework allows homogeneous communication between different modules (motion, audio, video), homogeneous programming and homogeneous information sharing. It is cross-platform, which means that it is possible to use it on Windows, Linux or Mac. The framework is also cross-language, with an identical API for both C++ and Python and provides introspection, which means the framework knows which functions are available in the different modules and where. The Robot Operating System (ROS) has been used as middleware to allow the communication between the different nodes in the Scitos G5 robot. ROS is a flexible framework for writing robot software. It is a collection of tools, libraries, and conventions that aim to simplify the task of creating complex and robust robot behaviour across a wide variety of robotic platforms. It is cross-platform and cross-language.The middleware used in the iCub robot was YARP. YARP supports building a robot control system as a collection of programs communicating in a peer-to-peer way, with an extensible family of connection types (tcp, udp, multicast, local, MPI, mjpg-over-http, XML/RPC, tcpros, etc).

The system is made available to the research community at the following address: `https://github.com/mpatacchiola/naogui`

7.4 Conclusions

The study of how trust evolves in HRI is a new area of research that can have a tremendous impact on the development of social robots (Breazeal, 2004). A social robot should be able to understand people in social terms and to empathise with them. An important study by

Reeves & Nass (1996) showed that a social interface is better for many reasons. First, it is more user-friendly and people are more willing to interact with systems that incorporate it. Second, the interaction is more natural since humans evolved in a social environment and already have innate social skills. Third, via social learning it may be possible to directly teach the robot, breaking the technical barrier that separates the users from the mastery of this technology.

In line with these considerations, this chapter has shown that through manipulating the robot appearance and behaviour, it is possible to influence its perceived trustworthiness. This result may have practical applications, since it gives some hints on how a social robot should be designed.

Chapter 8

Conclusions

8.1 Overview

This chapter provides an overview of the findings and topics covered in this thesis. In particular, Section 8.2 is a summary of the contribution to knowledge that reviews the main goals of the thesis and discusses them in a wider context. Section 8.3 contains some possible research directions which aim to expand the current work. Moreover, it discusses the research that is currently under development as part of the THRIVE project, which is particularly connected with the work presented in this thesis.

8.2 Summary of the Contribution to Knowledge

This section will revisit the contributions outlined in the introduction (Chapter 1), with further expansion and explanation. The key goal of the work presented in this thesis was the advancement of cognitive architecture for trust-based learning in social robots. This high-level goal was divided into three sub-goals that have been described in Section 1.2. Here, it is provided a summary to describe how those three goals have been achieved:

1. The first goal has been achieved through the development of a flexible causal architecture. This took the form of a developmental Bayesian model which is able to predict the belief and action of children with different cognitive abilities in trust related tasks.
2. The second goal, has been achieved through the design of a robust head pose

estimator, based on deep learning techniques that can substantially improve the skills of social robots in HRI.

3. The third goal has been achieved with a cognitive architecture that integrates in a single framework previous theoretical work, and that takes inspiration from developmental psychology and neuroscience.

At first, the thesis explored a new approach for modelling the behaviour of children in trust-related tasks. The Bayesian model has been described in Chapter 4 and the underlying theory introduced in Chapter 3. This model is one of the few that links together ToM and trust in a unified framework coherently with the psychological literature (Gopnik & Meltzoff, 1997). This model has been developed with flexibility in mind. Causal models such as BNs can be easily adapted to meet different criteria.

The second step consisted of finding a way to exploit the embodiment provided by a robotic platform, in line with the long term objectives discussed in Chapter 1 and with the developmental robotics approach (Cangelosi & Schlesinger, 2015). In Chapter 5 the importance of body language, and in particular head orientation and gaze, in the establishment of trust has been discussed. For this reason a robust head pose estimation system based on deep learning methods has been developed. This system can be used as the main component of a social robot attentional system and it can guarantee the synchronisation of intentions and actions with the user. The system achieved state-of-the-art performances in a large variety of datasets, proving to be effective even in-the-wild. As a results of this work, an open source library called *Deepgaze* has been created and made available to the community (`https://github.com/mpatacchiola/deepgaze`). In its current release Deepgaze obtained more than 750 stars and 250 forks on GitHub, and has been used in scientific work such as Rieger (2018) and Ajay et al. (2018).

Finally, previous work has been integrated into a cognitive architecture and tested on the iCub humanoid robot, reproducing a developmental experiment. This cognitive architecture is based on a reinforcement learning approach and is inspired by neuroscientific observations of the reward system in the mammalian brain. The causal framework developed in Chapter 4 has been integrated as an intrinsic environment that provides an online estimation of the reliability of a particular informant. A psychological experiment of word-learning (Koenig & Harris, 2005) has been used as benchmark for evaluating the architecture. The results were coherent with the psychological data providing strong evidences for the plausibility of the model.

In Chapter 7 the contribution given by the author to a series of HRI experiments as a part of the THRIVE project has also been discussed. Although that work was not directly connected with the main goal of the research, it greatly helped in strengthening the background hypotheses, highlighting the importance of embodiment, gaze, and voice in the establishment of trust between humans and robots.

8.3 Future Work

The long term goal of this thesis was to give a contribution to the development of a cognitive architecture for trust and to integrate it in social robots. The previous sections provide an explanation of how this goal has been achieved. In this section some of the work still under development is introduced, and possible future research directions are suggested and discussed.

8.3.1 Adding an episodic memory in the Bayesian network model

An interesting research direction concerns the upgrade of the model presented in Chapter 4 via an episodic memory. This is currently under development, and the main components of the work done so far are introduced in this section.

The aim of an episodic memory is to accomplish the ability for the agent to build a personal character based on the way it has been treated in the past: a robot which has been tricked very often would learn to be more mistrustful and vice versa, as in the *trust vs mistrust* phase in child development (Erikson, 1993). A BN is generated on the fly every time a new informant interacts with the robot, then the parameters of the network are sampled from a reservoir of past experiences. The reservoir is called *replay dataset* and it is based on the following principles: (i) memories fade away with time; (ii) the more information is memorised, the more blurred the single details become; (iii) shocking and unexpected events, such as surprises and betrayals, are more difficult to forget than ordinary expected episodes. The episodic memory is based on an information theory approach, that quantifies the amount of information each episode represents. The idea is to find how much this value differs from the total entropy of the replay dataset, where an high difference means that the event is to be considered surprising and must be easier to recall than ordinary events. For example, if an informant who has always been trustworthy suddenly tricks the agent, this betrayal will be remembered with a greater impact. At the same time, all of the memories are subject to a progressive time degradation that tends to blur them with a timing dependent on their importance.

8.3.2 Social trust in a multi-agent scenario

This thesis focused on the analysis of a one-to-one interaction between the child and the informant and did not consider complex dynamics that may evolve in large communities.

Since the interaction between multiple robots has several limitations (both technical and economical) it may be better to investigate trust in a simulated environment. In the past researchers have used multi-agent simulations to investigate behavioural and sociological phenomena, such as intertemporal choice (Paglieri et al., 2015), evolution of language (Cangelosi & Parisi, 2012), goods exchange (Parisi & Carmantini, 2013) and storage (Biscione et al., 2015). The use of simulated environments offer practical advantages. Firstly, the number of agents can be rapidly increased or decreased. Secondly, the real-time actor can be accelerated and a large amount of episodes can be collected.

Trust in Multi-Agent System (MAS) originates at the level of the individuals or at the level of the system (Ramchurn et al., 2004). At the individual level, an agent has beliefs about the honesty of the other partners. At system level, the actors are forced to be trustworthy by the rules that regulate the system itself. It is at the system level that different forms of social trust can arise. Social trust can be defined as a form of shared authentication that allows a group to identify and isolate malevolent subjects. Social trust has been extensively studied in peer-to-peer systems but its spontaneous development in multi-agent systems has never been properly investigated (Pinyol & Sabater-Mir, 2013). The study of social trust has an important role in the increasingly crowded virtual worlds, where humans, bots, and hackers share the same spaces.

In order to investigate these factors a multi-agent Markov game called the *werewolf game* has been designed. In this game a specific number of agents N can trade in an open environment. Trading means that an agent can exchange symbolic tokens every time it is in the proximity of another agent, and the trading is agreed by both the parties. Each agent is identified by a unique colour and it can perceive a limited area around its frame (partially observable environment). Moving closer to each other, the agents can engage in exchanges that are modelled through a Bernoulli distribution. During an exchange each agent can return a positive reward to the partner sampling from its own probability function. Some agents are more generous than others. Among the agents there is a special one called the werewolf. Dealing with the werewolf leads to a negative reward and to a forced ban that last for several time steps. Since the colour of the agents randomly changes at the beginning of each episode, it is not possible to know in advance who the werewolf is. However, a board called *forum* can be used by every agent (included the werewolf) to share information. The forum is structured as a vector and passed in input to the function approximator of each agent. It is important to notice that the way the forum should be used is not specified. The main experimental hypothesis is that a form of social trust will emerge. The agents will use the forum to identify the malevolent agent and ban it from the trading.

A slightly different version of the werewolf game can be used to study prejudice. In this case the colour of the werewolf is chosen from a non-uniform distribution. That is, a specific colour is selected more often. If a form of prejudice emerges, then an innocent agent is banned by the community. How is it possible to mitigate such a prejudice? Providing an answer to this question may shed some light on the social processes involved in discrimination and racism.

8.3.3 On the predictive power of the model

The model presented in this thesis can be tested in additional conditions and then validated via experimental sessions. The developmental experiments presented in Chapters 4 and 6 are based on informants who are always right or always wrong, however in the real world such a case is extremely rare. Possible directions are related to mixed conditions where the informants provide an inconsistent information. This can be an ideal test bench for verifying the predictive power of the model, and can be easily achieved thanks to the flexibility of the internal environment, that can be reorganised to represent different situations.

Before concluding there is an important question that must be taken into account: would a simpler model provide the same results? A clear answer to this question is not possible given the lack of literature, however it is possible to speculate that a simpler model could in principle bring to similar results. Even tough such a model may exist, it should not only provide the same results but also be able to represent in a compact way the complexity of the neurobiological substrate of trust, as the proposed model does. This point is crucial and it puts more weight on the result achieved so far.

Bibliography

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., & Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.
URL `http://tensorflow.org/`

Aghajanian, J., & Prince, S. (2009). Face pose estimation in uncontrolled environments. In *BMVC*, vol. 1, (p. 3).

Ahn, B., Park, J., & Kweon, I. S. (2014). Real-time head orientation from a monocular camera using deep neural network. In *12th Asian Conference on Computer Vision*, (pp. 82–96). Singapore: Springer International Publishing.

Ajay, J., Song, C., Wang, A., Langan, J., Li, Z., & Xu, W. (2018). A pervasive and sensor-free deep learning system for parkinsonian gait analysis. In *Biomedical & Health Informatics (BHI), 2018 IEEE EMBS International Conference on*, (pp. 108–111). IEEE.

Alioua, N., Amine, A., Rziza, M., Bensrhair, A., & Aboutajdine1, D. (2013). Head pose estimation based on steerable filters and likelihood parametrized function. In *21st European Signal Processing Conference (EUSIPCO 2013)*, (pp. 1–5). Marrakech: IEEE.

Amodio, D. M., & Frith, C. D. (2006). Meeting of minds: the medial frontal cortex and social cognition. *Nature Reviews Neuroscience*, *7*(4), 268–277.

Anderson, J. R. (1996). Act: A simple theory of complex cognition. *American Psychologist*, *51*(4), 355.

Asada, M., Hosoda, K., Kuniyoshi, Y., Ishiguro, H., Inui, T., Yoshikawa, Y., Ogino, M., & Yoshida, C. (2009). Cognitive developmental robotics: A survey. *IEEE Transactions on Autonomous Mental Development*, *1*, 12–34.

Asada, M., MacDorman, K. F., Ishiguro, H., & Kuniyoshi, Y. (2001). Cognitive developmental robotics as a new paradigm for the design of humanoid robots. *Robotics and Autonomous Systems*, *37*, 185–193.

Baldassarre, G. (2011). What are intrinsic motivations? a biological perspective. In *Development and learning (icdl), 2011 ieee international conference on*, vol. 2, (pp. 1–8). IEEE.

Baxter, R. H., Leach, M. J. V., Mukherjee, S. S., & Robertson, N. M. (2015). An adaptive motion model for person tracking with instantaneous head-pose features. *IEEE Signal Processing Letters*, *22*, 578–582.

Bayliss, A. P., & Tipper, S. P. (2006). Predictive gaze cues and personality judgments: Should eye trust you? *Psychological Science*, *17*(6), 514–520.

Behrens, T. E., Hunt, L. T., & Rushworth, M. F. (2009). The computation of social behavior. *science*, *324*(5931), 1160–1164.

Behrens, T. E., Woolrich, M. W., Walton, M. E., & Rushworth, M. F. (2007). Learning the value of information in an uncertain world. *Nature neuroscience*, *10*(9), 1214–1221.

Berg, J., Dickhaut, J., & McCabe, K. (1995). Trust, reciprocity, and social history. *Games and economic behavior*, *10*(1), 122–142.

Berlyne, D. E. (1966). Curiosity and exploration. *Science*, *153*(3731), 25–33.

Bickmore, T., & Cassell, J. (2001). Relational agents: a model and implementation of building user trust. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, (pp. 396–403). ACM.

Biscione, V., Petrosino, G., & Parisi, D. (2015). External stores: simulating the evolution of storing goods and its effects on human behaviour. *Interaction Studies*, *16*(1), 118–140.

Breazeal, C. L. (2004). *Designing sociable robots*. MIT press.

Broadbent, E., Stafford, R., & MacDonald, B. (2009). Acceptance of healthcare robots for the older population: Review and future directions. *International Journal of Social Robotics*, *1*(4), 319.

Burgoon, J. K., Bonito, J. A., Bengtsson, B., Cederberg, C., Lundeberg, M., & Allspach, L. (2000). Interactivity in human–computer interaction: A study of credibility, understanding, and influence. *Computers in human behavior*, *16*(6), 553–574.

Butterfield, J., Jenkins, O. C., Sobel, D. M., & Schwertfeger, J. (2009). Modeling aspects of theory of mind with markov random fields. *International Journal of Social Robotics*, *1*, 41–51.

Cangelosi, A., & Parisi, D. (2012). *Simulating the evolution of language*. Springer Science & Business Media.

Cangelosi, A., & Schlesinger, M. (2015). *Developmental Robotics, From Babies to Robots*. Cambridge, MA: MIT Press.

Carpenter, M. (2009). Just how joint is joint action in infancy? *Topics in Cognitive Science*, *1*(2), 380–392.

Charman, T. (2003). Why is joint attention a pivotal skill in autism? *Philosophical Transactions of the Royal Society B: Biological Sciences*, *358*(1430), 315–324.

Delgado, M. R., Frank, R. H., & Phelps, E. A. (2005a). Perceptions of moral character modulate the neural systems of reward during the trust game. *Nature neuroscience*, *8*(11), 1611–1618.

Delgado, M. R., Miller, M. M., Inati, S., & Phelps, E. A. (2005b). An fmri study of reward-related probability learning. *Neuroimage*, *24*(3), 862–873.

Dementhon, D. F., & Davis, L. S. (1995). Model-based object pose in 25 lines of code. *International Journal of Computer Vision*, *15*, 123–141.

Dennet, D. C. (1987). *The Intentional Stance*. Cambridge, MA: MIT Press.

DiYanni, C., & Kelemen, D. (2008). Using a bad tool with good intention: Young children's imitation of adults' questionable choices. *Journal of Experimental Child Psychology*, *101*, 241–261.

Driver IV, J., Davis, G., Ricciardelli, P., Kidd, P., Maxwell, E., & Baron-Cohen, S. (1999). Gaze perception triggers reflexive visuospatial orienting. *Visual cognition*, *6*(5), 509–540.

Duchi, J., Hazan, E., & Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, *12*(Jul), 2121–2159.

Earl, W. L. (1987). Creativity and self-trust: A field study. *Adolescence*, *22*(86), 419.

Emery, N. J. (2000). The eyes have it: the neuroethology, function and evolution of social gaze. *Neuroscience & Biobehavioral Reviews*, *24*(6), 581–604.

Emery, N. J., Lorincz, E. N., Perrett, D. I., Oram, M. W., & Baker, C. I. (1997). Gaze following and joint attention in rhesus monkeys (macaca mulatta). *Journal of comparative psychology*, *111*(3), 286.

Erikson, E. H. (1950). *Childhood and Society: 1963*. WW Norton & Co., Inc., New York, NY.

Erikson, E. H. (1993). *Childhood and society*. WW Norton & Company.

Evers, V., Maldonado, H., Brodecki, T., & Hinds, P. (2008). Relational vs. group self-construal: untangling the role of national culture in hri. In *Human-Robot Interaction (HRI), 2008 3rd ACM/IEEE International Conference on*, (pp. 255–262). IEEE.

Falcone, R., & Castelfranchi, C. (2001). Social trust: A cognitive approach. In *Trust and deception in virtual societies*, (pp. 55–90). Springer.

Fanelli, G., Dantone, M., Gall, J., Fossati, A., & Gool, L. V. (2012). Random forests for real time 3d face analysis. *International Journal of Computer Vision*, *101*, 437–458.

Fanelli, G., Weise, T., Gall, J., & Gool, L. V. (2011). Real time head pose estimation from consumer depth cameras. *Pattern Recognition*, *6835*, 101–110.

Fink, J. (2012). Anthropomorphism and human likeness in the design of robots and human-robot interaction. In *International Conference on Social Robotics*, (pp. 199–208). Springer.

Frank, M., Leitner, J., Stollenga, M., Förster, A., & Schmidhuber, J. (2014). Curiosity driven reinforcement learning for motion planning on humanoids. *Frontiers in neurorobotics*, *7*, 25.

Freedy, A., DeVisser, E., Weltman, G., & Coeyman, N. (2007). Measurement of trust in human-robot collaboration. In *Collaborative Technologies and Systems, 2007. CTS 2007. International Symposium on*, (pp. 106–114). IEEE.

Fusaro, M., Corriveau, K. H., & Harris, P. L. (2011). The good, the strong, and the accurate: Preschoolers' evaluations of informant attributes. *Journal of Experimental Child Psychology*, *110*, 561–574.

Geronimo, D., Lopez, A. M., Sappa, A. D., & Graf, T. (2010). Survey of pedestrian detection for advanced driver assistance systems. *IEEE Transactions on Pattern Analisys and Machine Intelligence*, *32*, 1239–1258.

Goetz, J., Kiesler, S., & Power, A. (2003). Matching robot appearance and behavior to tasks to improve human-robot cooperation. In *Proceedings of the 12th IEEE Workshop on Robot and Human Interactive Communication*, (pp. 55–60). San Francisco, CA.

Goodman, N. D., Baker, C. L., Bonawitz, E. B., Mansinghka, V. K., Gopnik, A., Wellman, H., & Tenenbaum, J. B. (2006). Intuitive theories of mind: A rational approach to false belief. In *Proceedings of the Twenty-Eighth Annual Conference of the Cognitive Society*, (pp. 1382–1387). Vancouver, Canada: R. Sun.

Gopnik, A., Glymour, C., Sobel, D. M., Schulz, L. E., & Kushnir, T. (2004). A theory of causal learning in children: Causal maps and bayes nets. *Psychological Review*, *111*, 3–32.

Gopnik, A., & Meltzoff, A. N. (1997). *Words, thoughts, and theories*. Cambridge, MA: MIT Press.

Gopnik, A., & Schulz, L. E. (2004). Mechanisms of theory formation in young children. *TRENDS in Cognitive Sciences*, *8*, 371–377.

Gopnik, A., Sobel, D. M., Schulz, L. E., & Glymour, C. (2001). Causal learning mechanisms in very young children: Two-, three-, and four-year-olds infer causal relations from patterns of variation and covariation. *Developmental Psychology*, *37*, 620–629.

Gordon, R. M. (1986). Folk psychology as simulation. *Mind and Language*, *1*, 158–171.

Gourier, N., Hall, D., & Crowley, J. L. (2004). Estimating face orientation from robust detection of salient facial features. In *Proceedings of Pointing 2004, ICPR, International Workshop on Visual Observation of Deictic Gestures*, (pp. –). Cambridge, UK.

Gourier, N., Maisonnasse, J., Hall, D., & Crowley, J. L. (2006). Head pose estimation on low resolution images. *Lecture Notes in Computer Science*, *4122*, 270–280.

Hancock, P. A., Billings, D. R., Schaefer, K. E., Chen, J. Y. C., de Visser, E. J., & Parasuraman, R. (2011). A meta-analysis of factors affecting trust in human-robot interaction. *Human Factors*, *53*, 517–527.

Hebb, D. O., et al. (1949). The organization of behavior: A neuropsychological theory.

Hegde, C., Sankaranarayanan, A. C., & Baraniuk, R. G. (2012). Learning manifolds in the wild. *Preprint, July*, *1*(2), 4.

Heyman, G. D., Sritanyaratana, L., & Vanderbilt, K. E. (2013). Young children's trust in overtly misleading advice. *Cognitive Science*, *37*(4), 646–667.

Huggins-Daines, D., Kumar, M., Chan, A., Black, A. W., Ravishankar, M., & Rudnicky, A. I. (2006). Pocketsphinx: A free, real-time continuous speech recognition system for hand-held devices. In *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, vol. 1, (pp. I–I). IEEE.

Itseez (2015). Open source computer vision library. `https://github.com/itseez/opencv`.

Jaswal, V. K., & Neely, L. A. (2006). Adults don't always know best: Preschoolers use past reliability over age when learning new words. *Psychological Science*, *17*, 757–758.

Juvina, I., Lebiere, C., & Gonzalez, C. (2015). Modeling trust dynamics in strategic interaction. *Journal of Applied Research in Memory and Cognition*, *4*(3), 197–211.

Kaelbling, L. P., Littman, M. L., & Moore, A. W. (1996). Reinforcement learning: A survey. *Journal of artificial intelligence research*, *4*, 237–285.

Kelley, J. F. (1984). An iterative design methodology for user-friendly natural language office information applications. *ACM Transactions on Information Systems (TOIS)*, *2*(1), 26–41.

Kennedy, J., Baxter, P., & Belpaeme, T. (2015). Head pose estimation is an inadequate replacement for eye gaze in child-robot interaction. In *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction Extended Abstracts*, (pp. 35–36). ACM.

Kim, J. H., & Pearl, J. (1983). A computational model for combined causal and diagnostic reasoning in inference systems. In *Proceedings of the IJCAI-83.*, (pp. 190–193). Karlsruhe, Germany.

Kim, S., Paulus, M., & Kalish, C. (2016). Young children's reliance on information from inaccurate informants. *Cognitive Science*.

King, D., Rowe, A., & Leonards, U. (2011). I trust you; hence i like the things you look at: Gaze cueing and sender trustworthiness influence object evaluation. *Social Cognition*, *29*(4), 476–485.

King-Casas, B., Tomlin, D., Anen, C., Camerer, C. F., Quartz, S. R., & Montague, P. R. (2005). Getting to know you: reputation and trust in a two-person economic exchange. *Science*, *308*(5718), 78–83.

Kingma, D., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Koeller, D., & Friedman, N. (2009). *Probabilistic Graphical Models*. Cambridge, MA: MIT Press.

Koenig, M. A., & Harris, P. L. (2005). Preschoolers mistrust ignorant and inaccurate speakers. *Child Development*, *76*, 1261–1277.

Koestinger, M., Wohlhart, P., Roth, P. M., & Bischof, H. (2011). Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In *First IEEE International Workshop on Benchmarking Facial Image Analysis Technologies*.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, & K. Q. Weinberger (Eds.) *Advances in Neural Information Processing Systems 25*, (pp. 1097–1105). Curran Associates, Inc.

Kumaran, D., & Maguire, E. A. (2007). Which computational mechanisms operate in the hippocampus during novelty detection? *Hippocampus*, *17*(9), 735–748.

Kushnir, T., Xu, F., & Wellman, H. M. (2010). Young children use statistical sampling to infer the preferences of other people. *Psychological Science*, *21*, 1134–1140.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, (pp. 2278–2324). IEEE.

Lee, K., Cameron, C. A., Doucette, J., & Talwar, V. (2002). Phantoms and fabrications: Young childrenâĂŹs detection of implausible lies. *Child Development*, *73*, 1688–1702.

Leibo, J. Z., Zambaldi, V., Lanctot, M., Marecki, J., & Graepel, T. (2017). Multi-agent reinforcement learning in sequential social dilemmas. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, (pp. 464–473). International Foundation for Autonomous Agents and Multiagent Systems.

Leslie, A. M. (1994). Pretending and believing: issues in the theory of tomm. *Cognition*, *50*, 211–238.

Lia, D., & Pedrycz, W. (2014). A central profile-based 3d face pose estimation. *Pattern Recognition*, *47*, 525âĂŞ–534.

Lopes, A. T., de Aguiar, E., De Souza, A. F., & Oliveira-Santos, T. (2016). Facial expression recognition with convolutional neural networks: Coping with few data and the training sample order. *Pattern Recognition*.

Majorana, A. (1950). *Ricerche sull'apprendimento dei ratti in labirinto-Sul comportamento investigativo dei ratti*. Universita Istituto di psicologia.

Malagavi, N., Hemadri, V., & Kulkarni, U. (2014). Head pose estimation using convolutional neural networks. *International Journal of Innovative Sciencem Engineering and Technology*, *1*, 470–475.

Marr, D., & Vision, A. (1982). A computational investigation into the human representation and processing of visual information. *WH San Francisco: Freeman and Company*, *1*(2).

Mason, M. F., Tatkow, E. P., & Macrae, C. N. (2005). The look of love: Gaze shifts and person perception. *Psychological Science*, *16*(3), 236–239.

McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, *5*(4), 115–133.

Metta, G., Fitzpatrick, P., & Natale, L. (2006). Yarp: yet another robot platform. *International Journal of Advanced Robotic Systems*, *3*(1), 8.

Metta, G., Sandini, G., Vernon, D., Natale, L., & Nori, F. (2008). The icub humanoid robot: an open platform for research in embodied cognition. In *Proceedings of the 8th workshop on performance metrics for intelligent systems*, (pp. 50–56). ACM.

Minsky, M., & Papert, S. (1969). Perceptrons: AnIntroduction to computational geometry.

Montague, P. R., King-Casas, B., & Cohen, J. D. (2006). Imaging valuation models in human choice. *Annu. Rev. Neurosci.*, *29*, 417–448.

Morse, A. F., Greeff, J. D., Belpeame, T., & Cangelosi, A. (2010). Epigenetic robotics architecture (era). *IEEE Transactions on Autonomous Mental Development*, *2*, 325–339.

Moses, L. J., & Baldwin, D. A. (2005). What can the study of cognitive development reveal about children's ability to appreciate and cope with advertising? *Journal of Public Policy and Marketing*, *24*, 186–201.

Mukherjee, S. S., & Robertson, N. M. (2015). Deep head pose: Gaze-direction estimation in multimodal video. *IEEE Transactions On Multimedia*, *17*, 2094–2107.

Murphy-Chutorian, E., & Trivedi, M. M. (2009). Head pose estimation in computer vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *31*, 607–626.

Nagy, E., & Molnar, P. (2004). Homo imitans or homo provocans? human imprinting model of neonatal imitation. *Infant Behavior and Development*, *27*(1), 54–63.

Neff, G., & Nagy, P. (2016). Automation, algorithms, and politics | talking to bots: Symbiotic agency and the case of tay. *Internationtal Journal of Communication*, *10*, 17.

Ngo, H., Luciw, M., Förster, A., & Schmidhuber, J. (2013). Confidence-based progress-driven self-generated goals for skill acquisition in developmental robots. *Frontiers in psychology*, *4*, 833.

Nogueira, K., Penatti, O. A., & Santos, J. A. d. (2016). Towards better exploiting convolutional neural networks for remote sensing scene classification. *arXiv preprint arXiv:1602.01517*.

Osadchy, M., Cun, Y. L., & Miller, M. L. (2007). Synergistic face detection and pose estimation with energy-based models. *The Journal of Machine Learning Research*, *8*, 1197–1215.

Paglieri, F., Parisi, D., Patacchiola, M., & Petrosino, G. (2015). Investigating intertemporal choice through experimental evolutionary robotics. *Behavioural processes*, *115*, 1–18.

Parisi, D., & Carmantini, G. S. (2013). Robots that specialize and make exchanges. In *Artificial Life (ALIFE), 2013 IEEE Symposium on*, (pp. 120–125). IEEE.

Patacchiola, M., & Cangelosi, A. (2016). A developmental bayesian model of trust in artificial cognitive systems. In *Proceedings of the International Conference on Development and Learning and Epigenetic Robotics*.

Patacchiola, M., & Cangelosi, A. (2017). Head pose estimation in the wild using convolutional neural networks and adaptive gradient methods. *Pattern Recognition*, *71*, 132–143.

Pearl, J. (1982). Reversed bayes on inference engines: A distributed hierarchical approach. In *Proceedings of the Second National Conference on Artificial Intelligence.*, (pp. 133–136). Menlo Park, California: AAAI Press.

Pérolat, J., Leibo, J. Z., Zambaldi, V., Beattie, C., Tuyls, K., & Graepel, T. (2017). A multi-agent reinforcement learning model of common-pool resource appropriation. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.) *Advances in Neural Information Processing Systems 30*, (pp. 3643–3652). Curran Associates, Inc.

Piaget, J. (1955). *The child's construction of reality*. Routledge & Kegan Paul Limited.

Pinyol, I., & Sabater-Mir, J. (2013). Computational trust and reputation models for open multi-agent systems: a review. *Artificial Intelligence Review*, *40*(1), 1–25.

Premack, D. G., & Woodruff, G. (1978). Does the chimpanzee have a theory of mind? *Behavioral and Brain Sciences*, *1*, 515–526.

Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Wheeler, R., & Ng, A. Y. (2009). Ros: an open-source robot operating system. In *ICRA workshop on open source software*, vol. 3, (p. 5). Kobe, Japan.

Ramchurn, S. D., Huynh, D., & Jennings, N. R. (2004). Trust in multi-agent systems. *The Knowledge Engineering Review*, *19*(1), 1–25.

Rau, P. P., Li, Y., & Li, D. (2009). Effects of communication style and culture on ability to accept recommendations from robots. *Computers in Human Behavior*, *25*(2), 587–595.

Reeves, B., & Nass, C. I. (1996). *The media equation: How people treat computers, television, and new media like real people and places.*. Cambridge university press.

Ribeiro, D., Nascimento, J. C., Bernardino, A., & Carneiro, G. (2016). Improving the performance of pedestrian detectors using convolutional learning. *Pattern Recognition*.

Ricci, E., & Odobez, J.-M. (2009). Learning large margin likelihoods for realtime head pose tracking. In *16th IEEE International Conference on Image Processing (ICIP)*, (pp. 2593–2596). Cairo: IEEE.

Rieger, I. S. (2018). *Head Pose Estimation using Deep Learning*. Ph.D. thesis, University of Bamberg.

Rishwaraj, G., Ponnambalam, S., & Loo, C. K. (2017). Heuristics-based trust estimation in multiagent systems using temporal difference learning. *IEEE transactions on cybernetics*, *47*(8), 1925–1935.

Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, *65*(6), 386.

Rotter, J. B. (1967). A new scale for the measurement of interpersonal trust1. *Journal of personality*, *35*(4), 651–665.

Rouanet, P., Oudeyer, P.-Y., Danieau, F., & Filliat, D. (2013). The impact of human-robot interfaces on the learning of visual objects. *IEEE Transactions on Robotics*, *29*(2), 525–541.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986a). Learning internal representation by error propagation. *Nature*, *323*, 318–362.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986b). Learning representations by back-propagating errors. *nature*, *323*(6088), 533.

Russell, S., & Norvig, P. (2010). *Artificial Intelligence a Modern Approach*. Pearson, 3 ed.

Sabater, J., & Sierra, C. (2005). Review on computational trust and reputation models. *Artificial intelligence review*, *24*(1), 33–60.

Samejima, K., Ueda, Y., Doya, K., & Kimura, M. (2005). Representation of action-specific reward values in the striatum. *Science*, *310*(5752), 1337–1340.

Saxe, R. (2006). Uniquely human social cognition. *Current opinion in neurobiology*, *16*(2), 235–239.

Scassellati, B. (2002). Theory of mind for a humanoid robot. *Autonomous Robots*, *12*(1), 13–24.

Schillo, M., Funk, P., & Rovatsos, M. (2000). Using trust for detecting deceitful agents in artificial societies. *Applied Artificial Intelligence*, *14*(8), 825–848.

Schmidhuber, J. (1991). Curious model-building control systems. In *Neural Networks, 1991. 1991 IEEE International Joint Conference on*, (pp. 1458–1463). IEEE.

Schmidhuber, J. (2006). Developmental robotics, optimal artificial curiosity, creativity, music, and the fine arts. *Connection Science*, *18*(2), 173–187.

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, *61*, 85–117.

Shafto, P., Eaves, B., Navarro, D. J., & Perfors, A. (2012). Epistemic trust: Modeling children’s reasoning about others’ knowledge and intent. *Developmental Science*, *15*, 436–447.

Singh, S. P., Barto, A. G., & Chentanez, N. (2004). Intrinsically motivated reinforcement learning. In *NIPS*, vol. 17, (pp. 1281–1288).

Sobel, D. M., Tenenbaum, J. B., & Gopnik, A. (2004). ChildrenâĂŹs causal inferences from indirect evidence: Backwards blocking and bayesian reasoning in preschoolers. *Cognitive Science*, *28*, 303–333.

Spiegelhalter, D. J., Dawid, A. P., Lauritzen, S. L., & Cowell, R. G. (1993). Bayesian analysis in expert systems. *Statistical Science*, *8*, 219–247.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, *15*, 1929–1958.

Stiefelhagen, R. (2004). Estimating head pose with neural networks - results on the pointing04 icpr workshop evaluation data. In *Proceedings of Pointing, ICPR, International Workshop on Visual Observation of Deictic Gestures*, (pp. –). Cambridge, UK.

Sundararajan, K., & Woodard, D. L. (2015). Head pose estimation in the wild using approximate view manifolds. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, (pp. 50–58). Boston, Massachusetts.

Süßenbach, F., & Schönbrodt, F. (2014). Not afraid to trust you: Trustworthiness moderates gaze cueing but not in highly anxious participants. *Journal of Cognitive Psychology*, *26*(6), 670–678.

Sutton, R. S. (1984). *Temporal credit assignment in reinforcement learning*. Ph.D. thesis, UMass Amherst.

Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction*, vol. 1. MIT press Cambridge.

Takahashi, Y., Roesch, M. R., Stalnaker, T. A., & Schoenbaum, G. (2007). Cocaine exposure shifts the balance of associative encoding from ventral to dorsolateral striatum. *Frontiers in integrative neuroscience*, *1*(11).

Tieleman, T., & Hinton, G. (2012). Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural Networks for Machine Learning*, *4*(2).

Tikhanoff, V., Cangelosi, A., Fitzpatrick, P., Metta, G., Natale, L., & Nori, F. (2008). An open-source simulator for cognitive robotics research: the prototype of the icub humanoid robot simulator. In *Proceedings of the 8th workshop on performance metrics for intelligent systems*, (pp. 57–61). ACM.

Tollefsen, D. (2005). LetâĂŹs pretend! children and joint action. *Philosophy of the Social Sciences*, *35*(1), 75–97.

Tomasello, M., et al. (1995). Joint attention as social cognition. *Joint attention: Its origins and role in development*, *103130*.

Torki, M., & Elgammal, A. (2011). Regression from local features for viewpoint and pose estimation. In *2011 International Conference on Computer Vision*, (pp. 2603–2610). IEEE.

Trevarthen, C. (1998). The concept and foundations of infant intersubjectivity. *Intersubjective communication and emotion in early ontogeny*, (pp. 15–46).

Tu, J., Fu, Y., Hu, Y., & Huang, T. (2007). Evaluation of head pose estimation for studio data. *Lecture Notes in Computer Science*, *4122*, 281–290.

VandenBos, G. R. (2015). *APA Dictionary of Psychology*. Washington, DC: Maple Press.

Vanderbilt, K. E., Liu, D., & Heyman, G. D. (2011). The development of distrust. *Child Development*, *82*, 1372–1380.

Vertegaal, R. (1999). The gaze groupware system: mediating joint attention in multiparty communication and collaboration. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, (pp. 294–301). ACM.

Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1, (pp. I–511). IEEE.

Voit, M., Nickel, K., & Stiefelhagen, R. (2007). Neural network-based head pose estimation and multi-view fusion. In *1st international evaluation conference on Classification of events, activities and relationships*, (pp. 291–298).

Vrij, A., & Mann, S. (2001). Telling and detecting lies in a high-stake situation: the case of a convicted murderer. *Applied Cognitive Psychology: The Official Journal of the Society for Applied Research in Memory and Cognition*, *15*(2), 187–203.

Wang, C., & Song, X. (2014). Robust head pose estimation via supervised manifold learning. *Neural Networks*, *53*, 15–25.

Wellman, H. M., Cross, D., & Watson, J. (2001). Meta-analysis of theory-of-mind development: The truth about false belief. *Child Development*, *72*, 655–684.

Wellman, H. M., & Liu, D. (2004). Scaling of theory-of-mind tasks. *Child Development*, *75*, 523–541.

Widrow, B. (1960). *Adaptive adaline Neuron Using Chemical memistors*. N/A.

Wood, B. S. (1976). Children and communication: Verbal and nonverbal language development. *N/A*.

Wu, J., & Trivedi, M. M. (2008). A two-stage head pose estimation framework and evaluation. *Pattern Recognition*, *41*, 1138–1158.

Yang, W., Jin, L., Tao, D., Xie, Z., & Feng, Z. (2016). Dropsample: A new training method to enhance deep convolutional neural networks for large-scale unconstrained handwritten chinese character recognition. *Pattern Recognition*, *58*, 190–203.

Zanatto, D., Patacchiola, M., Goslin, J., & Cangelosi, A. (2016). Priming anthropomorphism: Can the credibility of humanlike robots be transferred to non-humanlike robots? In *Proceedings of the Eleventh Annual ACM/IEEE International Conference on Human-Robot Interaction.*, (pp. 543–544). Christchurch, New Zealand: IEEE Press.

Zeiler, M. D. (2012). Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.

Zhu, X., & Ramanan, D. (2012). Face detection, pose estimation, and landmark localization in the wild. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, (pp. 2879–2886). IEEE.