Egocentric Representations for Autonomous Navigation of Humanoid Robots

Hendry Ferreira Chame

Thèse de Doctorat

Hendry FERREIRA CHAME

Mémoire présenté en vue de l'obtention du
grade de Docteur de l'École centrale de Nantes
sous le label de l'Université de Nantes Angers Le Mans

École doctorale : Sciences et technologies de l'information, et mathématiques

Discipline : Automatique, productique et robotique
Unité de recherche : Institut de Recherche en Communications et Cybernétique de Nantes (IRCCyN)

Soutenue le 11 Janvier 2016

Représentations Ego-centrées pour la Navigation Autonome d'un Robot Humanoïde

JURY

Président :	**M. Philippe MARTINET**, Professeur des universités, École Centrale de Nantes, IRCCyN
Rapporteurs :	**M. François CHARPILLET**, Directeur de recherche, INRIA Villers-lès-Nancy
	M. David FILLIAT, Docteur HDR, ENSTA ParisTech
Examinateurs :	**M. Yannick AOUSTIN**, Professeur des universités, École Centrale de Nantes, IRCCyN
	M. Philippe LUCIDARME, Maître de conférences, Université d'Angers, LARIS
	M. Alexandre PITTI, Maître de conférences, Université de Cergy-Pontoise, Cergy-Pontoise
Directrice de thèse :	**M^{me} Christine CHEVALLEREAU**, Directrice de recherche, CNRS, École Centrale de Nantes, IRCCyN

Dedication

To the memory and legacy of my grandfathers.

Acknowledgments

There is no way this work would be finished without the important contributions from individuals, institutions, and places that I am pleased to acknowledge:

I start by expressing my gratitude to my beloved family for inspiring me, providing the basis, and supporting me in this journey.

I thank the Ecole Centale de Nantes (ECN) in the person of the headmaster, Mr Arnaud Poitou, for having me at such a prestigious institution.

I thank the Institut de Recherche en Cybernétique de Nantes (IRCCyN) of the CNRS, in the person of the director, Mr Michel Malabre, for receiving me.

I would like to express my gratitude to my supervisor, Mme Christine Chevallereau, whose expertise, cordiality, and patience, added considerably to my personal and professional growth.

I thank the members of the jury, Mr François Charpillet, Mr David Filliat, Mr Alexandre Pitti, Mr Yannick Aoustin, Mr Philippe Lucidarme, and Mr Philippe Martinet; for kindly accepting the invitation.

I express my gratitude to Mr Wisama Kalil from the ECN for the good advices. I also thank Mr Gaëtan Garcia for inviting me to lecture the ARPRO module for the master EMARO+/ARIA.

I thank the funding institutions. The Ecole Centrale de Nantes (ECN) and EQUIPEX ROBOTEX, of France; and the CAPES Foundation, Ministry of Education of Brazil, Brasília - DF 700040-020, Brazil.

I thank my research colleagues for the interesting discussions at lunch time, the trips, and the fun. I also express my gratitude to my dear friends from Nantes and abroad, for the good vibes, the support, and the positive influence.

I thank our mother earth. The countries of France, Venezuela, and Brazil for providing me wonderful experiences and opportunities. And life for giving me the energy and the passion to fulfill this dream. Cheers!

Hendry Ferreira Chame

Résumé étendu

La recherche sur l'automatisation du comportement a mis en évidence divers défis technologiques pour parvenir aux performances d'un système biologique. Il devient de plus en plus clair que les caractéristiques des organes sensoriels et moteurs humains sont essentiels pour atteindre certains objectifs. Malgré l'intérêt croissant en matière de solutions robotiques pour des applications de service et d'assistance, une machine qui soit polyvalente et qui imite de façon réaliste le corps anthropomorphe de l'être humain n'est pas encore disponible. Actuellement, le domaine de l'intelligence artificielle (IA) passe par des reformulations importantes. L'approche cognitiviste de l'IA n'a pas abouti à des modèles et des stratégies de représentation adaptés pour fournir un système de résolution de problème universel. Pendant les dernières décennies, la recherche en cognition incarnée (Embodied Cognition (EC)), où la représentation de la connaissance est fondée sur l'interaction physique avec l'environnement s'est développée offrant une alternative pour l'étude du comportement naturel. Toutefois, l'adoption de la méthodologie EC pose également des défis importants pour les roboticiens. Notamment, lorsqu'elle vise à satisfaire les exigences imposées par l'hypothèse du fondement physique (physical grounding hypothesis). Ainsi, son utilisation dans les applications de robotique de service n'est pas encore très développée.

Cette étude a pris un point de vue intermédiaire entre la méthodologie cognitiviste et l'EC. Ce travail porte sur l'aspect architectural du comportement et se concentre sur l'exploration des sources locales d'information pour obtenir des solutions flexibles et robustes vis-à-vis des applications en robotique de service. Lors de ce travail une compétence fondamentale a été considérée comme cas d'étude : il s'agit de l'utilisation de l'ego-localisation pour se rapprocher et se positionner par rapport à des cibles visuelles. Pour cela, d'une part, on adopte l'hypothèse cognitiviste selon laquelle le robot peut se servir des représentations indépendantes-de-l'action (sous la forme de schémas perceptifs) pour faire la reconnaissance visuelle de la cible. Alors que, d'autre part, une fois que le robot s'engage dans une tâche sensorimotrice il aura recours à des représentations locales sous la forme de sensations corporelles afin d'anticiper les conséquences de l'action, de discriminer les objets, de réagir à des circonstances imprévues, d'apprendre à partir d'expériences passées, et d'évaluer le progrès et le succès de la mission. Ainsi, à partir d'une approche multidisciplinaire, ce travail porte sur différents aspects : les architectures de comportements, l'attention visuelle ascendante et descendante, la vision par ordinateur, la localisation égocentrique embarquée, la sélection d'action, l'intégration multisensorielle, et l'apprentissage par renforcement.

Contents

1 Introduction **1**
- 1.1 Humanoid assistants . . . 1
- 1.2 A shift in artificial intelligence research . . . 3
- 1.3 The research problem . . . 4
- 1.4 Overview of Chapters . . . 4
- 1.5 Contributions . . . 5
- 1.6 Notation . . . 6

2 Humanoid navigation **9**
- 2.1 Introduction . . . 9
- 2.2 Humanoid robots . . . 10
  - 2.2.1 Some milestones . . . 10
  - 2.2.2 Worldwide research . . . 11
  - 2.2.3 French projects . . . 15
  - 2.2.4 The humanoid Nao . . . 15
- 2.3 Challenges in humanoid research . . . 18
  - 2.3.1 Bipedal locomotion . . . 18
  - 2.3.2 Perception . . . 19
  - 2.3.3 Human-robot interaction . . . 20
  - 2.3.4 Dexterous manipulation . . . 20
  - 2.3.5 Learning and adaptive behavior . . . 21
- 2.4 Autonomous navigation . . . 22
  - 2.4.1 Robot localization . . . 23
  - 2.4.2 Spatial cognition in the brain . . . 24
- 2.5 The action selection problem . . . 26
  - 2.5.1 Deliberative models . . . 27
  - 2.5.2 Reactive models . . . 28
  - 2.5.3 Hybrid models . . . 29
  - 2.5.4 Behavior-based models . . . 30
- 2.6 Conclusions . . . 31

3 Visual attention **33**
- 3.1 Introduction . . . 33
- 3.2 Theories of attention . . . 34
  - 3.2.1 The filter theory . . . 34
  - 3.2.2 The spotlight theory . . . 36
  - 3.2.3 The FIT and GS theories . . . 37
  - 3.2.4 Inspiring robotics solutions . . . 37
- 3.3 Machine vision . . . 38

3.3.1 The camera sensor . . . 39
3.3.2 The human eye . . . 40
3.3.3 Perspective projection . . . 41
3.3.4 Visual feature extraction . . . 42
3.4 Case studies . . . 48
3.4.1 Materials and resources . . . 48
3.4.2 CS-I: Semi-automatic color-based segmentation . . . 48
3.4.3 CS-II: Top-down color-based segmentation . . . 50
3.4.4 CS-III: Bottom-up segmentation based on optical flow . . . 55
3.5 Conclusions . . . 62

4 Visually-guided locomotion 65
4.1 Introduction . . . 65
4.2 Related work . . . 66
4.3 Visual servoing . . . 67
4.3.1 The interaction matrix . . . 68
4.3.2 Controlling the robot effectors in joint space . . . 69
4.4 Task definitions . . . 70
4.4.1 The Walk task . . . 70
4.4.2 The Look-at task . . . 71
4.5 Egocentric localization . . . 72
4.5.1 Sensory ego-cylinder . . . 72
4.5.2 Vision-based localization . . . 73
4.5.3 Object models . . . 74
4.6 Case studies . . . 76
4.6.1 Materials . . . 76
4.6.2 CS-I: Simulation of the approach to a salient object . . . 77
4.6.3 CS-II: Placement for the spatial reference system . . . 79
4.6.4 CS-III: Approaching a real object . . . 89
4.7 Conclusions . . . 91

5 Embodied perception 93
5.1 Introduction . . . 93
5.2 Grounding vision-based locomotion . . . 94
5.3 EC-based task analysis . . . 95
5.3.1 Mimicking human walking style . . . 95
5.3.2 Resources available . . . 97
5.3.3 Modeling a behavior scheme . . . 98
5.4 Behavior autonomy . . . 102
5.4.1 Embodied features . . . 103
5.4.2 Bayesian network for information fusion . . . 105
5.5 Case studies . . . 107
5.5.1 Materials and resources . . . 107
5.5.2 Behavior scheme implementation . . . 108
5.5.3 Hybrid architecture implementation . . . 109
5.5.4 CS-I: Model parameters estimation . . . 110
5.5.5 CS-II: Simulation of object redundancy . . . 113
5.5.6 CS-III: Approaching a real can . . . 117
5.6 Designing reliable approach tasks in six-steps . . . 121
5.7 Conclusions . . . 123

6 Reactive walking **125**
6.1 Introduction . . . 125
6.2 Related work . . . 126
6.3 The framework iB2C . . . 128
6.3.1 Model components . . . 128
6.3.2 Behavior coordination . . . 130
6.3.3 The sequence node extension . . . 131
6.4 Reinforcement Learning . . . 132
6.5 Case studies . . . 134
6.5.1 Materials and resources . . . 135
6.5.2 Behavior-based models implementation . . . 135
6.5.3 CS-I: Action-oriented approach . . . 136
6.5.4 CS-II: Object approach and obstacle avoidance . . . 144
6.5.5 CS-III: Learning-based approach . . . 152
6.6 Conclusions . . . 161

7 Conclusions **163**
7.1 Research perspectives . . . 165

List of publications **167**

References **169**

Appendixes **183**

List of Tables

2.1 Characteristic of functional humanoids . . . 17
2.2 Characteristics of the Nao humanoid . . . 17
2.3 Sensors available in Nao . . . 17
2.4 Localization problems taxonomy . . . 24

3.1 Scene interpretation workflow . . . 45
3.2 Survey on bottom-up tracking methods . . . 48

4.1 The studied placements for the reference frame. . . . 79
4.2 Modified Denavit & Hartenberg parameters for Nao . . . 80
4.3 Dependent variables under study . . . 83
4.4 Comparison on alternatives versions of the Walk task . . . 87

5.1 Resources available to solve the task. . . . 98
5.2 Embodied features . . . 103
5.3 Filtering features . . . 104
5.4 Motion profile evaluation . . . 111
5.5 Feature signals vs. delay . . . 112
5.6 Embodied filtering . . . 112
5.7 Comparison on the Bayesian network policies . . . 116
5.8 CS-III experimental cases results . . . 118

6.1 Behavior inputs and outputs . . . 129
6.2 Components of a RL problem . . . 133
6.3 CS-I task parameters . . . 142
6.4 CS-II task parameters . . . 150
6.5 State arbitration profiles . . . 155
6.6 Arbitration events . . . 155
6.7 MDP state descriptions . . . 157
6.8 CS-III task parameters . . . 158
6.9 Rewards for the full set of actions . . . 160

List of Figures

1.1 Speedy's dilema . . . 2
1.2 World population aged 60 or over . . . 3

2.1 Honda humanoids . . . 11
2.2 The MIT Cog project . . . 11
2.3 Final round of the DARPA Challenge . . . 14
2.4 Aldebaran humanoids . . . 15
2.5 The robot Nao . . . 16
2.6 Nao's pelvis design . . . 16
2.7 Kinematics of Nao . . . 18
2.8 Passive dynamics research . . . 19
2.9 Sociable robots . . . 20
2.10 Robot hands . . . 21
2.11 Spatial representations . . . 25
2.12 Human planes of motion . . . 26
2.13 Architectures for mobile robot control . . . 27
2.14 Deliberative models . . . 28
2.15 Reactive model . . . 28
2.16 Deliberative vs. reactive models . . . 28
2.17 Hybrid model . . . 29

3.1 The filter theory of attention . . . 35
3.2 The spotlight attention model . . . 36
3.3 Bottom-up architecture for saliency detection . . . 38
3.4 Photosite charge wells and incident photons . . . 39
3.5 Simplified cross section of the human eye . . . 41
3.6 High dynamic range in photography . . . 42
3.7 Central perspective geometry . . . 43
3.8 Blob centroid illustration . . . 44
3.9 Contour central moment . . . 44
3.10 Scene interpretation workflow . . . 45
3.11 The aperture problem . . . 47
3.12 K-means segmentation from color and topology . . . 51
3.13 K-means vs. top-down segmentation . . . 52
3.14 First-order neighborhood system . . . 52
3.15 Segmentation of a natural scene . . . 55
3.16 Segmentation of colored objects . . . 55
3.17 Segmentation under camera motions . . . 56
3.18 Algorithm for displacement estimation . . . 59
3.19 Polynomial basis and signal reconstruction . . . 61

3.20 Optical flow segmentation 61

4.1 Definition of the reference frames 70
4.2 Top view of the localization parameters 71
4.3 Top view of the localization prediction 71
4.4 Illustration of the Look-at task 72
4.5 Representation of the ego-cylinder localization 73
4.6 Cylindrical object model 74
4.7 Estimation of the object's depth 75
4.8 The approach task modeled in Webots 78
4.9 Egocentric visualization of the localization 78
4.10 Evolution of the localization error 79
4.11 Evaluated placements for the base frame B 80
4.12 Geometrical model of the robot Nao 81
4.13 Definition of the ground frame 82
4.14 Frame placement initial conditions 83
4.15 Box plots results of frame placement 85
4.16 Body posture comparison between E_{g} and G 85
4.17 Top view of egocentric localization 86
4.18 Top view of object positions for T_{g} and E_{g} 86
4.19 Comparison on the evolution of the neck yaw 87
4.20 Box plot results for the noise experimental condition 88
4.21 Evolution of the localization error under noise for frame G 88
4.22 Yellow card approach experiment 90
4.23 On-board view of the yellow card 90
4.24 Multicolor saliency 91

5.1 HMW non-holonoic angular motion 97
5.2 First-order description of the walk 98
5.3 Behavior scheme for the approach task 99
5.4 Embodied filtering illustration 101
5.5 Hybrid task architecture 103
5.6 Embody features illustration 104
5.7 Bayesian network for contextual information fusion. 105
5.8 Dynamic policies for the Bayesian network 107
5.9 Deliberation state automate 110
5.10 Localization discrepancy vs delay 112
5.11 Evolution of the embodied features 113
5.12 Simulation of the task with object redundancy 114
5.13 Comparison between the network policies 115
5.14 Task demonstration 119
5.15 Visual saliency degradation 119
5.16 Experimental evaluation with object redundancy 120
5.17 On-board views of the approach sequence 121

6.1 Behavior module 128
6.2 Group behavior 130
6.3 Conditional behavior stimulator module 131
6.4 Graphical representation of MDPs 134
6.5 CS-I Two-level hierarchy model 137

6.6 CS-I behavior scheme model . . . 138
6.7 Arbitration for walk control . . . 140
6.8 CS-I sparse flow task model . . . 142
6.9 Reactive object approach . . . 142
6.10 CS-I evaluation of the bilateral symmetry condition . . . 143
6.11 On-board view of the bilateral symmetry condition . . . 143
6.12 On-board view of the sparse flow evaluation . . . 143
6.13 CS-II task model . . . 145
6.14 Reactive obstacle avoidance . . . 148
6.15 Non-reactive vs. reactive approach . . . 150
6.16 Model performance . . . 151
6.17 Evaluation of motion consistency . . . 151
6.18 Bottom-up segmentation . . . 153
6.19 Visual encoding . . . 154
6.20 State automate for motion arbitration . . . 155
6.21 Top view of the action primitives . . . 156
6.22 MDP task model . . . 156
6.23 Visual encoding neighborhood . . . 157
6.24 kinesthetic demonstrations for scene encoding . . . 158
6.25 Encoding transition matrix . . . 159
6.26 Panoramic flow detection . . . 159
6.27 CS-II vs CS-III implementation . . . 160
6.28 Evaluation of the condition of multiple objects and obstacles . . . 160
6.29 RL policy learning . . . 161
6.30 CBR experience retrieval . . . 161

1

Introduction

Humanoid assistants

According to the historical review by Ichbiah [88] the word *robot* was popularized in 1920 in the science fiction Czech play *R.U.R.* by Karel Čapek. In the play artificial people are manufactured to help and free humanity from the slavery of manual labor, but they turn against their creators. These creatures would be made from synthetic organic matter so they could be easily mistaken for humans. Thus, perhaps under the effects of the severe consequences of the First World War, the literature of the 20-30s pictured a distopic view of robots as a menace and a replacement for mankind.

This negative connotation started to change through the work of Issac Assimov. The American Russian-born biologist contributed to a positive view of robots as our allies, servants and assistants; simply because we can choose it to be so. The famous introduction of the three laws of robotics in the short story *Runaround* (Assimov [13]), became an influential step towards ethics in robotics, accordingly:

1. *A robot may not injure a human being or, through inaction, allow a human being to come to harm.*
2. *A robot must obey the orders given to it by human beings, except where such orders would conflict with the First Law.*
3. *A robot must protect its own existence as long as such protection does not conflict with the First or Second Laws.*

The story takes place in a mining station on the planet Mercury. The photo-cell banks that would provide life support to the base station was short on selenium. Given that the robot SPD-13 (also called Speedy) could withstand Mercury's high temperatures, the crew asked it to get some selenium from the nearest pool. Speedy got confused about its mission, suffering from what is described as the "robotic equivalent of drunkenness". The situation is illustrated in Fig. 1.1. The robot walked uninterruptedly a huge circle

around the selenium pool when it was found by the crew. After failing to recover Speedy by a voice command, the astronaut Powell eventually realized that the selenium source contained unforeseen danger to the robot, so the second rule (to obey an order given by a human) and the third rule (to preserve itself) were in conflict. Thus, the momentary equilibrium in Speedy's behavior is finally broken when Powell decides to put himself in danger by going out in the heat, hoping that the first law would force the robot to overcome the cognitive dissonance and saving his life, what indeed happened.

Figure 1.1 – Illustration of Speedy's confusion, as described in the short story Runaround by Assimov [13]. Speedy cannot decide whether to execute the mission order or to protect himself from danger. The robot then oscillates between the two behavioral modes of approaching and avoiding the selenium pool.

Leaving aside for the moment the fascination with fiction, one might ask if there are legitimate reasons that justify the interest in the research of humanoid applications. This issue is tackled by Behnke [17], when he argues that the increasing popularity of humanoids research is motivated by the vision of creating a tool that cooperates with humans to solve problems in their same environment. That is, since our everyday tasks are human-centered designed, humanoids are believed to be more suited to move (e.g. climbing stairs) or to dexterously manipulate tools. In addition, a robot that is able to synthesize speech, to move the eyes, or to gesticulate; would favor a more intuitive and fluid communication with human beings, so increasing its adaptation and acceptance to the home or the office environment. The anthropomorphic body is also advantageous to facilitate programming by demonstration and automatic learning from imitation, since the actions would be more or less equivalents.

The development of service robotics has indeed been viewed as a promising way to provide assistance to elderly people, given the fact that the world population is aging. According to the UN [180], over the first half of the current century the global population 60 or over is projected to expand by more than three times, to reach nearly 2 billion in 2050 (see Fig. 1.2). Moreover, conforming to the projections of the International Monetary Fund and the World Bank (see Carone & Costello [33]), the aging of population would threaten the prospects for economic growth by exerting severe pressure on public expenditure. Therefore, humanoid robots could provide assistance and health care for elderly people so increasing their quality of life and autonomy.

However, despite the growing interest in robotics solutions to service and assistance applications, which is evidenced by ambitious funding to humanoid research projects in some developed countries; a general-purpose machine that realistically mimics the

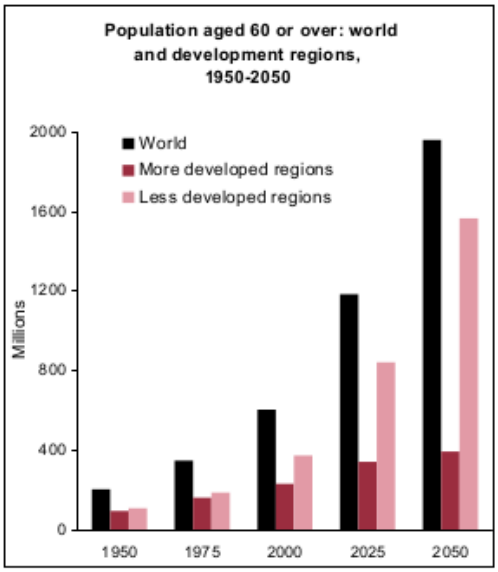


Figure 1.2 – Population aged 60 or over (UN [180]).

anthropomorphic body of human beings - until the point of being mistaken by humans as in the sci-fi literature - is still on the way. The research on behavior automation has pointed out to the technological challenge to reach the performance of the biological system, and to how the characteristics of human sensory and motor organs are crucial for the accomplishment of certain behaviors. In fact, by drawing attention to such differences, the research on humanoids (and robotics in general) has revealed itself as a useful means to understand human cognitive processes, and has also contributed to advances in the field of artificial intelligence.

A shift in artificial intelligence research

The field of artificial intelligence (AI) is going through important reformulations. The traditional view of AI (also known by *cognitivist* AI), under the influence of Cartesian dualism, has tended to look at physical and mental processes as belonging to different realms. Efforts in this direction have not come to a satisfactory end, notably, when proposing models and representation strategies to endorse a general-purpose problem solver with knowledge to deliberate on the task. A criticism has been formulated from what is known as the *Moravec's paradox* (Russell & Cohn [158]), that is, the discovery by AI and robotics researchers that, contrary to traditional assumptions, high-level reasoning requires very little computation, but low-level sensory-motor skills require enormous computational resources. Therefore, the humanoid robot has not been able to leave the environment of the lab, under strict control of extraneous variables.

In the last decades, a different perspective has been adopted to study natural behavior from the research on *embodied cognition* (EC), where knowledge representation is thought to be grounded in the physical interaction with the environment. Unlike the cognitivist approach, behavior is considered to emerge from multiple concurrent processes, so behavior would not be globally representable nor planed. The analysis of the sensory-motor coupling in natural tasks, from a dynamic system perspective, is becoming a promising research direction that can provide more efficient, robust, and autonomous solutions. However, adopting the EC methodology also poses important challenges to

robotists, in particular, when fulfilling the requirements underlying the *physical grounding hypothesis* (Brooks [29]). Firstly, the autonomous development of the behavior, as it happens in natural beings, would ideally occur under a phylogenetic architecture that can modify itself; which is hard to obtain for an artificial body. Secondly, the ontology of the system must be flexible enough to ensure knowledge acquisition for diverse purposes, by fusing information from different sensory modalities. Lastly, the development of cognitive skills is conditioned to sensory-motor coupling and interaction with the environment, thus knowledge acquisition is a slow process analogous to natural learning.

In view of the advantages and the challenges encountered in the aforementioned research approaches, this work has opted for an intermediate perspective for behavior automation, by acknowledging the importance of obtaining an adequate balance between generality and autonomy in applications of service robotics. In this sense, it takes into account the aspects of deliberation and reaction in the context of the action selection problem. For example, getting back to Speedy's dilemma (see Fig. 1.1), the deliberative aspect of the mission would be to accomplish the general plan of bringing some selenium to the crew, whereas the reactive aspect would be to handle the unexpected situations encountered, such that the emergence of danger. Both aspects are important for the mission, and the robot should ideally be able to detect the contradictory effects of these behavioral modes, and eventually to stop and to ask for guidance.

The research problem

This research focuses on the architectural aspect of the behavior of a humanoid robot, and concentrates on the exploration of local sources of information for obtaining more flexible and robust solutions to service applications. It has taken as a case study the fundamental skill of approaching and positioning in relation to visual stimuli. Thus, the problems of top-down and bottom-up visual attention, knowledge representation, and action selection, are investigated. For this, the work adopts the cognitivist assumption that action-independent knowledge (in the form of perceptive schemes) can be employed for recognizing stimuli. But, as an embodied being, when the robot engages in sensory-motor activities, it can efficiently resort to local representations in the form of bodily sensations, in order to anticipate the consequences of action, to discriminate the object, to react to unexpected circumstances, and to assess the progress and success on the mission.

Overview of Chapters

From the multidisciplinary approach adopted in this work, this manuscript reports on different topics of interest. Thus, **Chapter 2** starts by presenting an overview on humanoid robotics research and the main challenges encountered. Then, it focuses on the problem of humanoid navigation and localization, and the problem of deliberation and reactivity (i.e. the action selection problem). Different aspects of the top-down (i.e. deliberation) and bottom-up (i.e. reaction) processing are discussed, including the advantages and disadvantages of pure deliberative or reactive schemes.

Chapter 3 deals with the topic of visual attention. It starts by reviewing some models derived from cognitive science research. The focus is placed over the aspect of

the efficient management of the visual information available. The more commonly used sensor technologies are reviewed, and contrasted to the human eye, in order to illustrate potential challenges for artificial solutions. A review on the literature of machine vision is also presented, where the whole scene segmentation and the feature tracking approaches are described. Three case studies are developed in order to evaluate potential top-down and bottom-up approaches.

In **Chapter 4** the problem of egocentric on-board localization for autonomous walk is investigated. The visually-guided approach task is defined and modeled. For this, a distributed solution relying on visual servoing and motion primitives is studied. A cylindrical ego-sensory structure is defined for processing the localization, and different placements for this structure are compared. Several case studies in simulation and a real experiment are conducted in other to assess the autonomous execution of the task, under a restricted scenario to a single salient object.

Chapter 5 focuses on the aspect of attention selection. A more realistic solution to the approach task is proposed by defining a behavior scheme according to the EC research methodology. Thus, from a first-person perspective analysis of the sources of information available, the agent is given a non-holonomic walking style that mimics human motion. In order to ensure robustness and reliability in the task, the behavior scheme is integrated to a hybrid architecture in charge of monitoring the execution. This functionality is obtained from the design of a Bayesian network in charge of information fusion for attention selection. The chapter finishes by presenting the proposal of a six-steps methodology to develop robust visually-guided approach tasks.

The central topic of concern in **Chapter 6** is reactive walking. For this, it includes the study of embodiment, knowledge representation, and learning; under different action selection scenarios. By keeping the first-person perspective adopted throughout the work, and the proposal of distributed models for the task; a behavior-based framework is selected to study concurrency in the access of available resources, so emergent behavior is produced. Local action-oriented representations of the task are studied, thus the agent can approach the object while avoiding obstacles. Visual encoding is proposed as an embodied description of the task, so more efficient solutions can be learned.

Contributions

From a personal perspective on the various topics studied, the technical and conceptual proposals presented next are original.

Chapter 3. The improvement of the MRF color-based segmentation technique described in Sec. 3.4.3 for a use of top-down saliency detection, in the context of real-time processing of visual inflow.

Chapter 4. The behavior scheme in Sec. 4.4 based on ego-cylindrical localization, combining the IBVS and PBVS modeling techniques, relying on a low-frequency acquisition rate, from robust color-based MRF segmentation. The embodied evaluation of the sensory ego-space for stimuli persistence in Sec. 4.6.3, so the hybrid control policy based on body- and eye-centered references is proposed to obtain a computationally more efficient solution, by heuristically exploiting the posture adopted by the robot while moving on a plane.

Chapter 5. The HMW egocentric first-order description of the walk in Sec. 5.3.1,

that mimics human motion style, when positioning in relation to the frontal face of an object of interest, by moving on a plane surface. The distributed scheme for visual selection in Sec. 5.3.3, that combines ideas from the information processing models in cognitive science, and the EC methodology. The embodied and filtering features set detailed in Sec. 5.4.1, that register diverse information about the stimulus of interest and the body context. The hybrid model for action selection in Sec. 5.5.3 based on motion primitives, so remote resources can be safely used in the task, from the probabilistic evaluation of the degree of confidence and the discriminative power of the attention selection process. The Bayesian network model for information fusion in Sec. 5.4.2, under static and dynamic estimation of the features certainty. The six-steps methodology for designing reliable approach tasks in Sec. 5.6.

Chapter 6. The behavior model in Sec. 6.5.3 that exploits embodiment and local heuristics, so a solution for the task is obtained by relying on high-frequency dense optic flow processing, and distributed action-oriented representations of the stimulus of interest. The behavior scheme in Sec. 6.5.4 for obstacle avoidance and object approaching, combining top-down and bottom up attention selection, relying on the processing of dense optic flow and whole scene segmentation, from on-board acquisitions in a humanoid robot. The visual encoding proposed as an embodied description of the task, so the arbitration of behavioral modes is produced, and actions in the form of walk primitives can be learned.

Notation

- Lowercase letters in boldface "$\mathbf{u}$" denote vectors which are always column vectors.
- Uppercase letters in boldface "$\mathbf{M}$" are used for matrices.
- The i^{th} element of a vector is denoted by "$\mathbf{u}_i$". In case a particular element is referenced in a matrix, a double index notation "$\mathbf{M}_{ij}$" is used (i.e., the element at the i^{th} row and the j^{th} column). A single index can be used for matrices denoting the i^{th} row "$\mathbf{M}_i$" of the matrix.
- The transpose of a real matrix or vector is denoted "$\mathbf{M}^{\text{t}}$".
- The inverse of a matrix is "$\mathbf{M}^{-1}$"
- The pseudo-inverse of a matrix is "$\mathbf{M}^{+}$".
- Vectors and matrices columns are delimited by brackets (e.g. $\mathbf{u} = [u_1\ u_2\ ...\ u_k]^{\text{t}}$).
- The inner product operator for vectors is "$\cdot$", so $r = \mathbf{u} \cdot \mathbf{v}$. The cross product operator of vectors is "$\times$", so $\mathbf{i} = \mathbf{u} \times \mathbf{v}$.
- The euclidean norm of a vector is obtained from the inner product, such that

$$\|\mathbf{u}\| = \sqrt{\mathbf{u}^{\text{t}} \cdot \mathbf{u}} \tag{1.1}$$

- The absolute norm of a vector is obtained such that

$$|\mathbf{u}| = \sum_i |\mathbf{u}_i| \tag{1.2}$$

- The observation of a variable o is "$\hat{o}$".
- The estimation or prediction of a variable s is "$\tilde{s}$".
- The saturation of a measurement k is "$\bar{k}$", so $|k| < \epsilon$ for the saturation value ϵ.

- Points in the 3D Cartesian space are uppercase so "$B = (X, Y, Z)$", the coordinates components are also uppercase. The fact that a point B is expressed in a given reference frame G is denoted by ${}^{G}B$.
- Image projections of points are represented by "$B' = (x, y)$", with pixel coordinates in lowercase.
- The axis of a frame is lowercase so "$\vec{\mathrm{x}}$".

2

Humanoid navigation

Contents

2.1 Introduction . . . 9
2.2 Humanoid robots . . . 10
2.2.1 Some milestones . . . 10
2.2.2 Worldwide research . . . 11
2.2.3 French projects . . . 15
2.2.4 The humanoid Nao . . . 15
2.3 Challenges in humanoid research . . . 18
2.3.1 Bipedal locomotion . . . 18
2.3.2 Perception . . . 19
2.3.3 Human-robot interaction . . . 20
2.3.4 Dexterous manipulation . . . 20
2.3.5 Learning and adaptive behavior . . . 21
2.4 Autonomous navigation . . . 22
2.4.1 Robot localization . . . 23
2.4.2 Spatial cognition in the brain . . . 24
2.5 The action selection problem . . . 26
2.5.1 Deliberative models . . . 27
2.5.2 Reactive models . . . 28
2.5.3 Hybrid models . . . 29
2.5.4 Behavior-based models . . . 30
2.6 Conclusions . . . 31

Introduction

The research is humanoid robotics has been intensifying over the last decades, including international collaboration in the form of annual meetings, conferences, and robotic

challenges. Thus, an interesting question to be asked is: how far are these robots from reaching the performance of human beings? To answer this question the first part of the chapter presents an overview on the field, including important milestones and challenges encountered in different topics, such that: locomotion, perception, human-machine interaction, dexterous manipulation, learning and adaptation. The problem of autonomous navigation is reviewed in more detail, since it is on the core of the research interests. For this, the distinction between allocentric and egocentric spatial representations is established. In view of the stochastic nature of the studied task, the problem of action selection (i.e. deciding what to do next given the task constraints) is reviewed, and existent approaches are classified into deliberative, reactive, hybrid and behavior-based. The relative advantages of these models are analyzed in the context of autonomous walk.

Humanoid robots

In a broader sense, a robot is a goal-oriented machine with capabilities of sensing, planning and acting on the environment (Corke [51]). According to the goal, diverse body configurations have been proposed to automate different industrial mass production processes. A humanoid is a robot with an anthropomorphic body plan and human-like senses (Behnke [17]). Although the concept of human-like automata is relatively old in the literature and arts, the appearance of the humanoid robot had to wait until late 20th century for the advances in digital computing.

Some milestones

The key initial contributions were produced both in Japan and the USA, although the projects had very different focus and background. In 1986 the Japanese company Honda started the confidential *Humanoid Project* with the goal of developing a robot that would coexist and cooperate with human beings. The evolution of the prototypes is illustrated in Fig. 2.1 (see Hirose & Ogawa [82]). The first versions corresponded to the E-Series, which focused exclusively on the automation of biped locomotion. Full-body humanoids appeared in the P-Series. The project was made public with the announcement of the P2 prototype in 1996. The release of the model P3 in 1997 was undoubtedly an important milestone. Like its predecessor, the robot was able to walk on flat floors and climb stairs, but it could also kneel, stand up, keep balance when disturbed, and move gracefully at the human speed. Since 2001 the latest series is called Asimo [1].

By 1993 in the MIT (USA) Rodney Brooks and his team started to construct the upper-body Cog (see Fig. 2.2). The project differed significantly from the standard assumptions of artificial intelligence, that viewed human as a general purpose individuals in possession of full monolithic control and internal models. Instead, the project adopted a multidisciplinary approach to robotics, with strong influence of cognitive science, systems theory, philosophy, and linguistics; under the hypothesis that: "human-level intelligence requires gaining experience from interacting with humans, like human infants do" (Brooks et al. [28]). Thus, the control for Cog is implemented as a heterogeneous network of different processors types operating at distinct levels in the control hierarchy, ranging from small micro-controllers for joint-level control, to digital signal processor (DSP) networks for audio and visual preprocessing. The Cog project was active until 2003, though it was

1. project's website http://asimo.honda.com

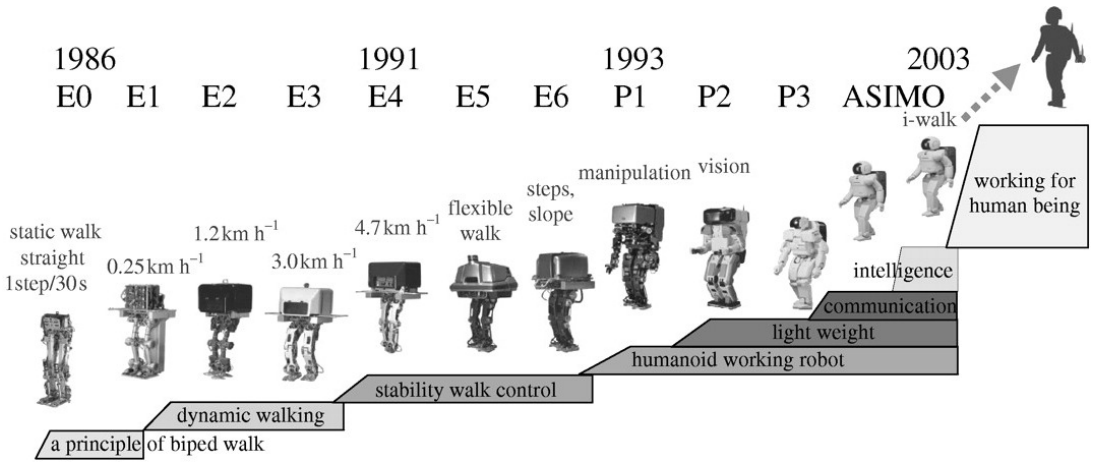


Figure 2.1 – Historic evolution of Honda's humanoid achievements [82].

very influential to other humanoid projects. This is the case of the robot iCub developed by the Italian Institute of Technology (IIT), as part of the EU RobotCub project (see Metta et al. [119]). The development counted on the collaboration of former participants of the Cog project.

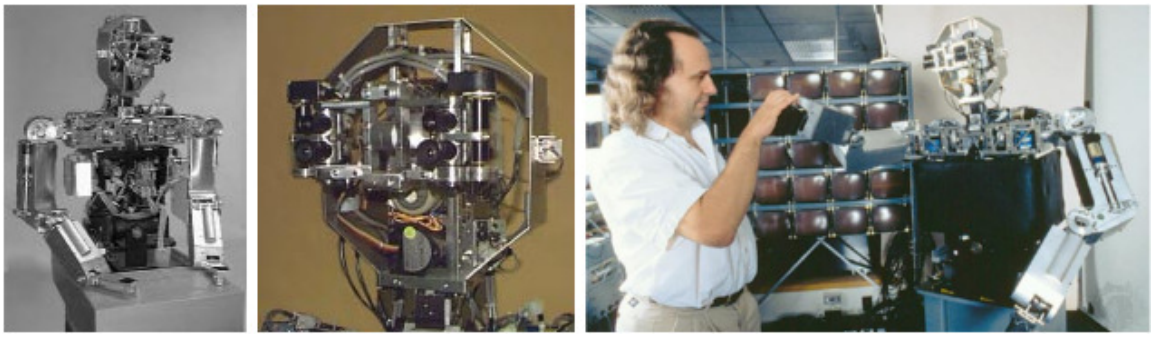


Figure 2.2 – The MIT Cog project. On the left, the upper-torso humanoid robot. Cog has twenty-one degrees of freedom. It is equipped with visual, vestibular, auditory, and tactile senses. In the center the robot's head. On the right, Rodney Brooks is interacting with Cog.

The upper-body Hadaly-2 was relatively contemporary with Cog. Hadaly-2 was developed in 1997 by the University of Waseda (Japan). The robot was given skills to interact with the environment, such that visual processing, conversation (e.g. voice recognition and synthesis), and gesticulation (Hashimoto et al. [79]). The project focused on the human morphology beyond the simple imitation of the anthropomorphic shape (e.g. the mobility of the eyes, neck and hands). However, the approach taken for behavior automation was not as innovative as in the Cog project and followed the standard AI assumptions. In this sense, the behaviors were explicitly modeled by the engineers and counted on extensive knowledge data-bases, including the 3D model of the scene.

Worldwide research

Japan has undoubtedly excelled in full-body humanoid innovation. After being left behind in the personal computer industry run, both the Japanese private and the public sector have striven for getting ahead on humanoids research. This has been in fact the case when Sony released the ludic 60 centimeters tall robot Qrio, that was at the forefront of

Asimo. The robot is able to dance, recognizing faces, detecting obstacles, climbing stairs, and running, among other skills. Some other important projects in the private sector are: the Toshiba series Partners, Fijitsu's HOAP-1 and HOAP-2, Kawada Industries' Isamu, and Kitano Symbiotic Systems' Pino, Sig2, Morph and Morph3. At the public sector in 1998 the Ministry of Economy, Trade and Industry of Japan launched the famous Humanoid Robotics Project (HRP). An important release was the prototype HRP-2, which can walk at two third human speed (2.5 km/h), move on narrow paths, cope with uneven surfaces, lie down, and get up by itself (Kaneko et al. [93]).

According to Ward [185], the USA has been left behind in the run for humanoids. Few investments have been done at the industrial level, due to non immediate profit return from the commercial point of view. A study in 2006 by the Technology Evaluation Center (see Ambrose et al. [5]) has compared the USA research activity with the rest of the world. It concluded that:

> *[...] The U.S. currently leads in such areas as robot navigation in outdoor environments, robot architectures (the integration of control, structure and computation), and in applications to space, defense, underwater systems and some aspects of service and personal robots. Japan and Korea lead in technology for robot mobility, humanoid robots, and some aspects of service and personal robots (including entertainment). Europe leads in mobility for structured environments, including urban transportation. Europe also has significant programs in eldercare and home service robotics. Australia leads in commercial applications of field robotics, particularly in such areas as cargo handling and mining, as well as in the theory and application of localization and navigation. In contrast with the U.S., Korea and Japan have national strategic initiatives in robotics; the European Community has EC-wide programs. [...] The U.S. lost its preeminence in industrial robotics at the end of the 1980s, so that nearly all robots for welding, painting and assembly are imported from Japan or Europe. The U.S. is in danger of losing its leading position in other aspects of robotics as well.*

Nevertheless, more recently some initiatives have been conducted in academia, space research, an defense. The country's first full-sized humanoid robot called CHARLI was developed by the Virginia Polytechnic Institute and State University (popularly known as Virginia Tech) in the Robotics and Mechanisms Laboratory (RoMeLa) [2]. The NASA has financed the Robonaut project [3] in collaboration with General Motors and Oceaneering. The current release is the highly dexterous model R2, build from multiple component technologies and systems (e.g. image recognition systems, sensor integrations, tendon hands, among others). The Defense Advanced Research Projects Agency (DARPA) has funded projects such that the robot Atlas developed by Boston Dynamics, and the Robotics Challenge DRC.

In relation to the research in China, the Beijing Institute of Technology (BIT) has been developing the BHR series, and the Zhejiang University (ZHU) has been working on the table-tennis-playing humanoid twins Kong and Wu; among other projects. In Korea there is the prestigious research team KAIST, who won the DARPA Robotics Challenge 2015. The team developed the robot DRC-Hubo, which is a semi-autonomous humanoid that presents a hybrid structure, it is capable of both biped and wheeled locomotion. In Thailand the King Mongkut's University of Technology Thonburi (KMUTT) has de-

2. http://www.romela.org/main/Robotics_and_Mechanisms_Laboratory
3. http://robonaut.jsc.nasa.gov/

veloped the robot Ka-Nok for playing football. In the Singapore Polytechnic (SP), the Advanced Robotics and Intelligent Control Centre (ARICC) research group has released the Robo-Erectur series. Australia has developed the robot GuRoo in the Mobile Robotics Laboratory of the University of Queensland.

European projects are diverse with Germany probably leading the way. The Technische Universität München (TUM) in Munich has developed Johnnie. In the same city the University of Bundeswehr has worked on the robot HERMES. The Karlsruhe Institute of Technology (KIT) has developed the ARMAR series for collaborative tasks. The NimbRo team [4] from the University of Bonn was the best ranked from Europe (coming at the fourth overall place) in the DARPA Robotics Challenge, with the robot Momaro (a four-legged humanoid torso). Other developments of the team are: the NimbRo-OP Humanoid Open Platform, Copedo, Dynaped and Bodo. In the Netherlands the Delft University of Technology is doing research on the concept of passive dynamics for energy storage in biped walking, and many robots have been released (e.g. Flames and Fides). In the context of the UK, the Imperial College of London developed an upper torso LUDWIG. In Sweden the Chalmers University has been developing several robots (e.g. Priscilla, Elvina, HR 2), also the University of Uppsala has released Murphy. In Russia the company New Era in cooperation with the St. Petersburg State Polytechnical University, have developed the robots ARNE and ARNEA. In Italy the Polytechnic University of Turin has released the Issac Robot. The humanoid iCub was also developed at the Italian Institute of Technology (IIT), as part of the EU project RobotCub and subsequently adopted by more than 20 laboratories worldwide [5]. The project is an open source initiative for the research in human cognition and artificial intelligence. The motivation behind the humanoid design is strongly based on the embodied cognition research, so the dimensions of the iCub are similar to those of a 2.5 year-old child.

Challenges and competitions

There are important international events around the humanoid community that have encouraged progress in the field. Indeed, many challenges and competitions have become application domains for these robots, since their commercialization to the wide public has not excelled yet. There are several participation modalities, the teams can either build their own robots or use available commercial kits. Some relatively famous events are for instance the soccer competitions RoboCup [6] and FIRA [7]. The goal is to develop fully autonomous robot teams that play together. The RobotChallenge [8] in the Humanoid Sprint modality requires the robots to complete a course walking or running as fast as possible. Other popular competition is Robo-One in Japan [9], where teleoperated robots engage in martial arts. In the competition RoboCup@home [10] the goal is to develop service and assistive technologies, with the emphasis on household and personal applications.

As already mentioned, DARPA has financed the Robotics Challenge (DRC) [11]. The teams that reached the final round are shown in Fig. 2.3. The objective of the compe-

4. http://www.nimbro.net/Humanoid/robots.html
5. http://www.icub.org/
6. http://www.robocup.org/robocup-soccer/
7. http://www.fira.net/main/
8. http://www.robotchallenge.org/competition/
9. http://www.robo-one.com/
10. http://www.robocupathome.org/
11. http://www.theroboticschallenge.org

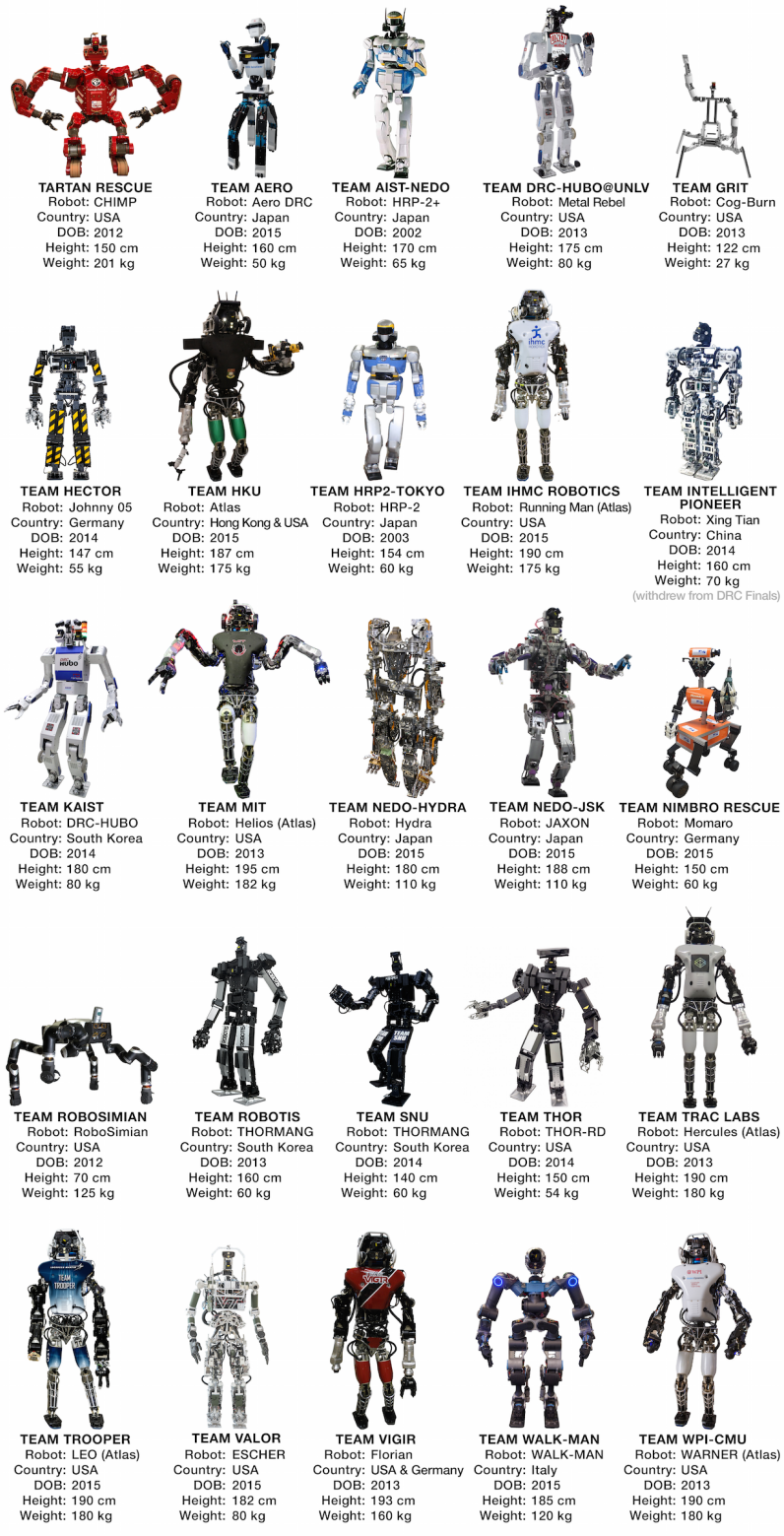


Figure 2.3 – Darpa Robotics Challenge finals 2015 [1]. The KAIST team from the Republic of Korea won the competition.

tition is to promote the development of semi-autonomous ground robots (most of them are anthropomorphic but it is not a requirement), for accomplishing complex tasks in dangerous, degraded, human-engineered environments. Examples of tasks are driving a vehicle, opening doors, clearing obstacles on the way, maneuvering a valve, and so on.

For this, the robot has to operate in environments it has not encountered previously, or to flexibly using human tools without requiring extensive reprogramming. The autonomy must be enough to ensure operation under degraded communications with the mission operator.

French projects

In France initial efforts started in 2000 with the project INRIA BIB. The objective was to research on various aspects related to the control of complex robotic systems, including walking machines. Due to shortages in the budget these efforts ceased in 2002 and were latter incorporated to the BIBOP project, which is currently active [12]. In 2001 the Centre National de la Recherche Scientifique (CNRS) founded the project Robea (Robotique et Entités Artificielles), which produced the RABBIT testbed platform (see Chevallereau et al. [45]). The project [13] was a joint effort between several French labs (including the IRCCyN), and some international contributions. The main goal was to build a prototype for studying dynamic motion control for high speed walking and running. The CNRS has also signed an agreement with the Japanese Humanoid Robotics Projects, so the robot HRP-2 is available at the Laboratoire d'Analyse et d'Architecture des Systèmes (LAAS) since 2006. The interactive robotics team of the Laboratoire des Systèmes Intégrés de Versailles (LISV), has been working since 2006 on HYDROïD [14], a humanoid robot for medical applications. In the private sector, Aldebaran Robotics has been successful worldwide with the launch of the robot Nao in 2004, which substituted since 2007 Sony's Aibo in the RoboCup Standard Platform League (SPL) [15]. The latest developments of the company are the robot Romeo and Pepper (see Fig. 2.4).

Figure 2.4 – Aldebaran humanoids. From left to right the robots Nao, Romeo and Pepper.

The humanoid Nao

The humanoid robot Nao by Aldebaran Robotics is the platform considered in this work (see Fig 2.5), so it is presented in detail. As described in Gouaillier et al. [75], Nao

12. http://www.inria.fr/equipes/bipop/%28section%29/activity
13. http://www.gipsa-lab.grenoble-inp.fr/projet/Rabbit/English/
14. http://www.uvsq.fr/hydroid-8201-un-robot-humanoide-au-service-de-la-sante-173507.kjsp
15. http://www.robocup.org/robocup-soccer/standard-platform/

is an innovative lightweight and compact robot, that is 0.57 meters tall and weights about 4.5 kg. The body mass index (BMI) is about 13.5 kg/m^2, which means that it is very light compared to other robots of the same height. Moreover, the walk speed is similar to the speed of a child of the same size, that is about 0.6 km/h.

Figure 2.5 – The robot Nao by Aldebaran Robotics (Gouaillier et al. [75])

Distinctive aspects of Nao are its pelvis kinematics design (see Fig. 2.6), its proprietary actuation system based on brush DC motors, its electronic, computer, and distributed software architecture. Nao is also affordable when compared to other platforms. As shown in Tab. 2.1, these robots are somewhat expensive, making the price for Nao (by 2008) a plausible alternative for research teams with moderate budget. The platform is also extensible and easy-to-handle, where the user can change the embedded software or add some applications to make the robot adopt specific behaviors. The robot's head and forearms are modular and can be changed to promote further evolution. The comprehensive and functional design is one of the reasons so Nao substituted the AIBO quadruped in the RoboCup standard league.

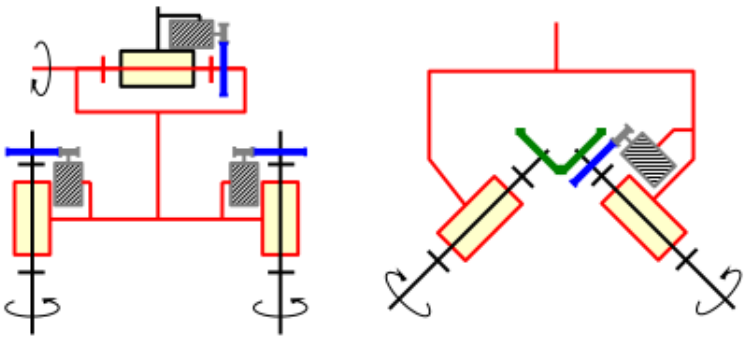


Figure 2.6 – Nao's pelvis design. On the left the classical set of three rotary joints, one horizontal axis at the waist and two vertical axis for the legs. On the right the coupled inclined axis rotary joints (at 45° towards the body) for the Nao pelvis (Gouaillier et al. [75]).

Table 2.2 summarizes the characteristics of Nao. It has a total of 25 degrees of freedom (DOF), 11 DOF for the lower part that includes the legs and pelvis, and 14 DOF for the upper part that includes the trunk, arms and head. Each leg has 2 DOF at the ankle, 1 DOF at the knee and 2 DOF at the hip. Figure 2.7 gives the kinematics details. Tab. 2.3 shows the sensory modalities included in the platform.

	Height (m)	**Weight (kg)**	**BMI (kg/m^2)**	**Price**
KHR-2HV	0.34	1.3	10.9	1K US $
HOAP	0.50	7.0	28.0	50K US $
Nao	**0.57**	**4.5**	**13.5**	**10K euros**
QRIO	0.58	6.5	19.0	NA
ASIMO	1.30	54.0	32.0	NA
REEM-A	1.40	40.0	20.4	400K US $
HRP-2	1.54	58.0	24.5	(5 year lease)
Human	1.5-2	50-100	18-25	NA

Table 2.1 – Characteristic of functional humanoids (Gouaillier et al. [75]). BMI: body mass index = w/h^2, NA: not available.

Body	
Height (m)	0.57
Weight (kg)	4.5
Battery	
Type	Lithium-ion
Capacity	55 Wh
Degrees of freedom (DOF): 25	
Head	2 DOF
Arms	5 DOF X 2
Pelvis	1 DOF
Leg	5 DOF X 2
Hands	1 DOF X 2

Masses (g)	
Chest	1217.1
Head	401
Upper Arm	163
Lower Arm	87
Thigh	533
Tibia	423
Foot	158
Total	4346.1

Table 2.2 – Characteristics of the Nao humanoid (Gouaillier et al. [75]).

Type	**Number**
30 FPS CMOS videocamera	1
Gyrometer	2
Accelerometer	3
Magnetic rotary encoder (MRE)	34
FSR	8
Infrared sensor (emitter/receiver)	2
Ultrasonic sensor	2
Loudspeaker	2
Microphone	4

Table 2.3 – Sensors available in Nao (Gouaillier et al. [75]).

The Aldebaran Robotics software framework is the architecture NaoQi, which is a modular and distributed environment that can deal with a variable number of executable binaries, depending on the user's architectural choices. The advantages of a distributed environment are diverse. It allows the user to run behaviors locally or remotely. Robot functionalities such that motion, vision, among others, can be run standalone or interacting with other modules on other computers. The development of applications is easier in a distributed environment, since the same code can be compiled on different platforms and cross-compiled for embedded execution. A distributed environment also allows the user to look at variables and running methods on any real or simulated robot from the programming interfaces.

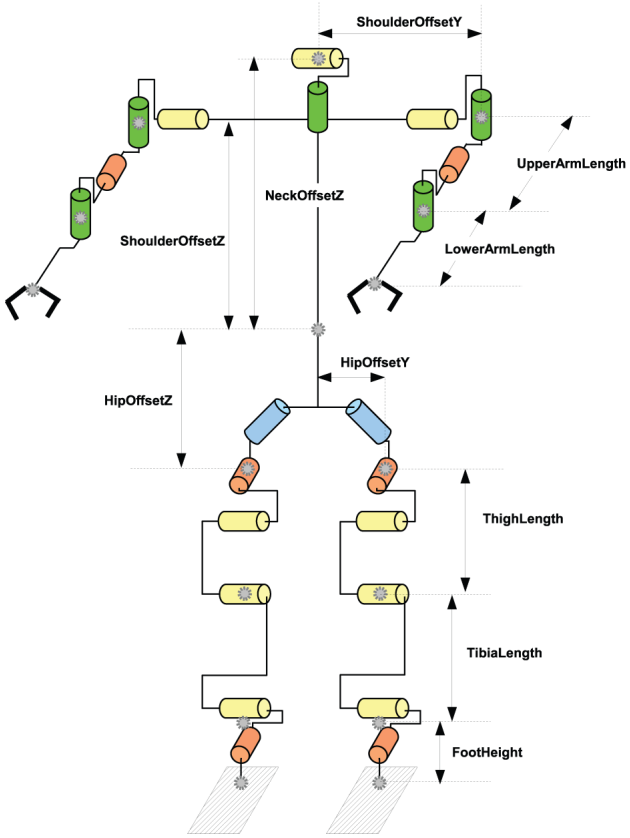


Figure 2.7 – Kinematics of Nao. The wrist joint is not represented (Gouaillier et al. [75]).

Challenges in humanoid research

Although it may look like the most important challenges for the conception and control of humanoids have been solved, the current capabilities of these robots are rather limited when compared to humans beings. Below some of these limitations are discussed.

Bipedal locomotion

One of the distinctive features of full-body humanoids is bipedal locomotion. Human beings can walk and run with apparently ease, though these skills have been proven hard to obtain in humanoids. According to Behnke [17] there are two opposing approaches to bipedal walking. One is based on the concept of zero-moment-point theory (ZMP), that is defined as the point on the horizontal plane about which the sum of the moments of all the active forces equals zero (Vukobratović & Borovac [183]). Dynamic stability can be evaluated such that, if the ZMP is within the convex hull (i.e. the support polygon) of all contact points between the feet and the ground, the system is stable. This was a major advance over the center-of-mass projection criterion describing static stability. Many robots (e.g, Asimo, Nao, HRP, etc) employ ZMP-based control, however they are not energy-efficient since they do not recycle energy stored in elastic elements in the way humans do.

The other approach for bipedal locomotion is to use the robot dynamics. A work by McGeer [118] has shown that for planar walking it is possible to down a slope without control or actuation. The idea of passive dynamic walking has inspired the study of walk

on level ground (see Colins et al. [48]). As shown Fig 2.8, these robots are very efficient and easy to control since their actuators only support the inherent machine dynamics. However, there are problems to be solved such that the autonomous starting and stopping of the walk, and the change of speed or direction. Furthermore, since round feet are employed, these machines cannot stand still. Perhaps a promising research direction is to combine ZMP theory with passive dynamics, though many aspects are still to be investigated (e.g. walking over uneven terrain and multi contact with the environment, among others).

Figure 2.8 – Passive dynamics research. Three level-ground powered walking robots based on the ramp-walking designs (Collins et al. [48]). On the left the Cornell biped, in the center the Delft biped, on the right the MIT learning biped.

Perception

Humanoid robots are equipped with sensory devices for perceiving their own state and the environment. Usually joint encoders, force sensors, or potentiometers, are available for proprioceptive feedback. Contact with the ground is detected through Force Sensitive Resistors (FSR) placed at the feet, which register resistance changes according to the pressure applied. More recently, some robots have been covered with force-sensitive skin (e.g. Stiehl & Breazeal [171], and Elkmann et al. [62]). Capacitive sensors are also used for detecting contact with the environment, they are usually placed at the hands, head and fore-arms. Super-human senses, such as laser range-finders or ultrasonic distance sensors may be available for exteroceptive feedback. Vision and audition are perhaps the most important modalities. Frequently robots are equipped with two movable cameras, the architecture may also include on-board computers for image interpretation. Though, the interpretation of real-world images is still an unsolved problem since the cameras employed are mostly general-purpose, differing significantly from the characteristics of the human visual system, which is much more efficient handling noise. Thereby, many vision-based tasks work well only under controlled conditions. Frequently, key objects are color-coded to ease their perception (e.g. in Moughlbay et al. [127]). Similar difficulties arise when interpreting the audio signals captured by on-board microphones (e.g. Allen et al. [4]). One major problem is the separation of the sound source of interest (e.g. a human communication partner) from other sound sources and noise. So far there is probably no audio recognition system that is infallible.

Human-robot interaction

Several research projects have focused on human-robot interaction. The general idea is to provide humanoids with capabilities normally present in natural face-to-face communications. This includes multiple sensory modalities such that speech, eye gaze, facial expressions, gestures with arms and hands, body language, and so on. According to Breazea [25], sociable robots are designed to communicate and interact with human, to understand and to relate to human or other robots in a personal way, by sharing social terms. Figure 2.9 shows some examples of these robots. They are equipped with expressive animated heads.

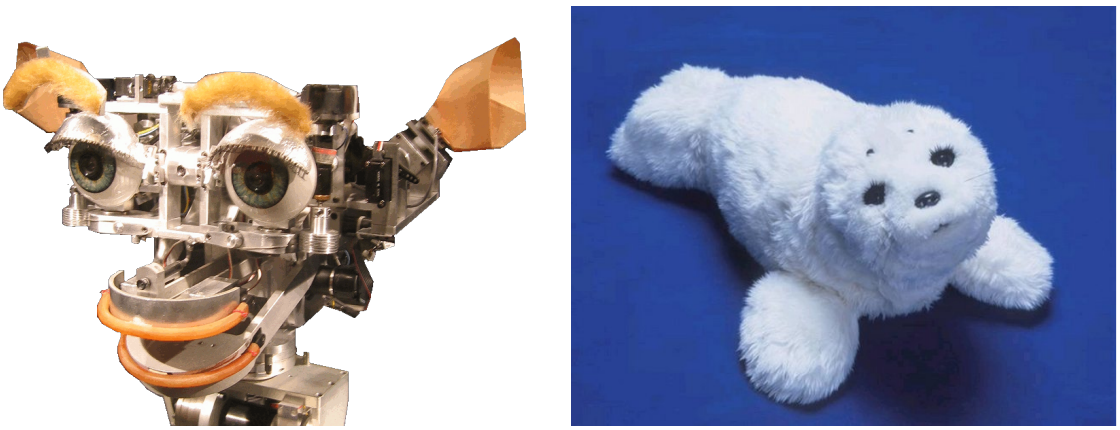


Figure 2.9 – Sociable robots. On the left, the robot Kismet developed in the MIT. On the right, the therapeutic baby harp seal Paro by AIST.

Perhaps the most extreme form of sociable robots are androids and gynoids, which exhibit a photo-realistic resemblance to humans. Their faces are covered with silicone skin, they have human-like hair, and are dressed as humans. Some of these robots are modeled after living persons, such that Repliee Q2 developed in Osaka (see Matsui et al. [117]). However, these robots may produce the *uncanny valley* effect (Mori [124]). Accordingly, as the appearance of a robot approaches the human, some observers' emotional response to the robot will become increasingly positive and empathic, until a point where there is a sudden drop in attractiveness close to perfect human-likeness. In Behnke's opinion [17], in these robots the synthesis-part of multi-modal interaction works reasonably well, but the insufficient performance in perception and action, and the lack of true meaning in the dialogue systems, prevent them so far from engaging in truly intuitive multimodal interactions with humans.

Dexterous manipulation

Another key human capability is dexterous manipulation. The human hand has about thirty degrees of freedom. It is not easy to reproduce its strength, flexibility, and sensitivity. As shown in Fig. 2.10, among the most advanced robotic hands is Shadow (Reichel [154]), which has 25 DOF and is capable of performing much of the motion of the human hand, including the curling of the palm. The actuation includes a flexible pneumatic system, acting as "air muscles". Other designs consider motorized biomimetic prosthetic hands based on tendon driven mechanisms (e.g. the iCub's hand, see Stellin et al. [170]). As pointed out by Amlie et al. [94], the study of human hand morphology reveals that developing an artificial hand with the dextereous capabilities of human hand

is an extremely challenging task. Furthermore, hand dexterous manipulation also requires hand-arm and visual coordination. In practice, most of the studies have considered tasks with known objects.

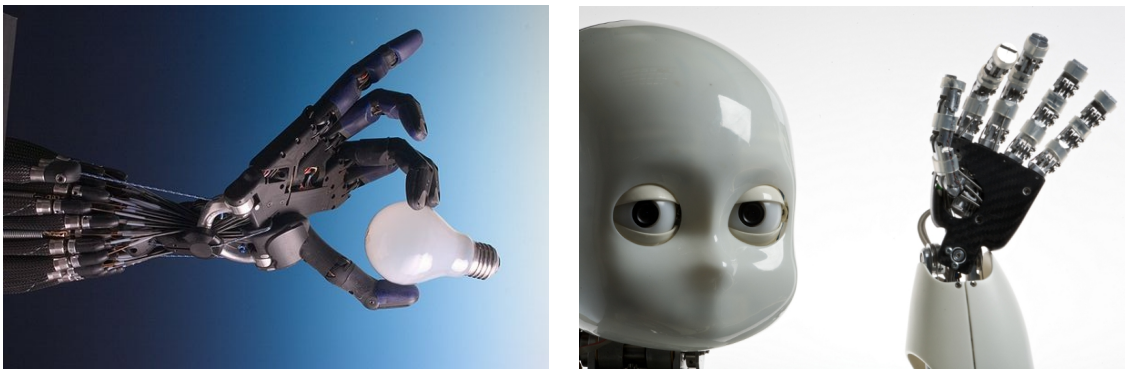


Figure 2.10 – Robot hands. On the left the hand developed Shadow Robot Company Ltd. On the right the iCub hand developed in the IIT.

Learning and adaptive behavior

Humanoids must ideally be flexible in their adaptation to the environment. Thus, it is desirable that they can autonomously acquire knowledge, extend current skills to solve related tasks, and cope with unexpected changes. An efficient way of learning is by imitation. According to Schaal [161], the study of imitation learning offers a promising route to gain new insights into mechanisms of perceptual motor control, that could ultimately lead to autonomous humanoid robotics solutions. The field focuses on three important aspects: motor learning, the connection between action and perception, and modular motor control in the form of motion primitives. There are several problems to be solved. One is obtaining a precise perception of the teacher. Other is the reliable mapping between the human body and the robot's body. That is, some human motions may not be possible for the robot given the body differences (e.g. in Munirathinam et al. [130]). The robot might have degrees of freedom that are not constrained by the captured motion, thus a useful technique to simplify imitation is kinesthetic teaching (e.g. in Kormushev [102]), where the teacher directly moves the limbs of the robot.

Programming by demonstration can also benefit the adaptability of the robot to the environment. According to Cypher [53] the motivation behind this methodology can be announced as follows: "if a user knows how to perform a task on the computer, that should be sufficient to create a program to perform the task". When extended to robotic agents, this method can be useful to teach procedural knowledge in the form of a task algorithm. Reinforcement learning (RL) has also been used to optimize the behavior of humanoid robots. A mode detailed discussion about RL is presented in Sec 6.4. In a nutshell, RL can be viewed as the mapping from situations to actions that maximizes a reward signal (Kaelbling et al. [92]). The learner is not told which actions to take, instead, it must discover those that yield the most reward by trying them. As pointed out by Russell & Norvig [159], an important challenge encountered in RL is the trade-off between exploration and exploitation. Specially because it cannot be assumed that the environment would generate a reward structure that is sufficient for the learning of complex tasks. Thus, the agent must try a variety of actions and progressively favor those policies that appear to be best.

A distinct perspective is adopted in the research of *enaction* (see Vernon [182]). The agent is given autonomy and an active role in learning through sensory motor coupling. Thus, knowledge is constructed from the interactions with the environment. Here, embodiment and stimuli affordances are fundamental for the robot's understanding of the world. That is, the physical properties of stimuli and the agent's body would provide the opportunity for learning actions. For example, the operations allowed by a knob are twisting and pushing, whereas a cord would afford pulling. By following these ideas autonomous sensory-motor coordination has been learned (e.g. eye-hand coordination in Fanello et al. [66]), and stimuli categorization (e.g. visual recognition in Morse et al. [125]), among other skills.

The aspects of learning and adaptation is of great interest for this research. Throughout chapters 4-6 different topics related to embodiment and emergent behavior are studied, where supervised demonstrations are fundamental to teach the agent the desired state of the task. The use of RL is also explored to increase the effectiveness in the task by selecting successful actions from previous experiences. The analysis of the sensory-motor coupling is a central topic, so the agent can rely on egocentric representations that it is able to obtain on-board in an autonomous manner (e.g. without relying on a ubiquitous representation of the scene). In this work the achievement of the task goals require the efficient control of the robot locomotion based on visual and proprioceptive information. The research on robot navigation and the action selection problem are relevant to this study, so they are discussed in more details in the next sections.

Autonomous navigation

The appearance of mobile robots in the late 1960s initiated the research domain of autonomous navigation. According to the historical review by Siciliano & Khatib [167], early navigation systems were based on fruitful ideas that influenced latter development of motion planning algorithms. Some examples are grid-based environment exploration (Nilsson [135]), and search-trees for the optimal path to a goal (Thompson [177]). Latter, studies in robot manipulation popularized the notion of the configuration space of a mechanical system (Lozano-Pérez [112]). Thus, motion planning was reduced to finding a path for a point in the configuration space.

Other important contribution came from the problem of car parking, which motivated the interest for non-holonomic motion planning (see Li & Canny [109]). According to Laumond [105], non-holonomic systems are characterized by constraint equations involving the time derivatives of the system configuration variables. These equations are non integrable and typically model the case where the system has less controls than configuration variables. For instance, a car-like robot has two controls (i.e. the linear and angular velocities) though it moves in a 3D configuration space. Consequently, a path in the configuration space does not necessarily correspond to a feasible path for the system. This is basically why the purely geometric techniques developed for holonomic motion planning do not apply directly to non-holonomic systems.

More recently, the focus of the research in the field has turned to the problem of autonomous outdoor navigation. That is, under inaccurate localization conditions, such that, uncertain and incomplete models of the world (e.g. navigation maps), and unexpected disturbance (e.g. moving obstacles). These topics definitely present a challenge to the cognitivist approach in AI, by emphasizing the gap between planning a path and

executing the motion. This is discussed in more detail in review of the action selection problem in Sec. 2.5.

Robot localization

According to Thrun et al. [178], localization can be seen as a problem of coordinate transformation. Maps are described in a global coordinate system, which is independent of the robot's pose. Thus, localization would be the problem of establishing correspondence between the map coordinate system and the robot's local coordinate system. Knowing this transformation enables the robot to express the location of objects of interests in space within its own reference frame. Conforming to Murphy [131], localization can be relative to a local environment (e.g., the robot is in the center of the room), to a topology (e.g., in Room 311), or to absolute coordinates (e.g., latitude, longitude, altitude).

Unfortunately, it is often the case where the localization cannot be directly sensed, but has to be inferred from data. The process of observation is subject to sensory noise, such that a single sensor measurement is usually insufficient to determine the localization. Instead, the robot has to integrate data over time to determine its pose. Depending on the circumstances of the task (e.g. the resources available) research topics may be of different levels of difficulty. A taxonomy for classifying research problems is given in Tab. 2.4. The categories are presented in increasing order of difficulty.

In this work the problem of localization considered is the observation of a desired configuration with respect to a know object fixed in the environment. Other elements in the scene (e.g. walls and furniture) are unknown. A global representation of the task, in the form of a navigation map, is assumed to be unavailable. Under this scenario, from the taxonomy given in Tab. 2.4, the research problems can fit on the categories: single-robot moving in a static environment, passive localization, and kidnapped robot. However, not only external references are studied in the walk task. As discussed in Chapter 6, from the egocentric perspective the robot avoids obstacles and learns motion primitives. This distinction is better established in the next section, when discussing multidisciplinary research in spatial cognition.

Category	Types
Local vs. global localization. (knowledge available initially and at runtime to the agent)	**Position tracking.** Known initial pose. Local problem. Uncertainty in the form of sensory noise (assumed to be small) is confined to a region near the true pose. Noise is accommodated to motion usually through unimodal distributions (e.g., a Gaussian). **Global localization.** Unknown initial pose. No boundedness of pose error can be assumed, thus unimodal probability distributions are usually inappropriate. **Kidnapped robot.** During operation the robot can get kidnapped and teleported to some other location. Thus, it might believe it knows where it is while it does not. The ability to recover from failures is essential for truly autonomous robots.

Static vs. Dynamic Environments	**Static.** The only variable quantity (state) is the robot's pose. Objects in the environment remain at the same location forever. **Dynamic.** Objects and the robot can move. Of particular interest are changes that persist over time, and that impact more than a single sensor reading. Changes that affect only a single measurement are best treated as noise.
Passive vs. Active Approaches	**Passive.** The localization module only observes the robot operating. The robot is controlled through some other means. Motion is not aimed at facilitating localization. **Active.** The robot is controlled so as to minimize the localization error and/or the costs arising from moving a poorly localized robot into a hazardous place.
Single- vs. Multi-Robot	**Single-robot.** All data is collected at a single robot platform, and there are no communication issues. **Multi-robot.** A team of robots is considered. One approach is to allow each robot to localize itself, such that robots are able to detect each other. There is also the possibility to use one robot's belief to bias other's when knowledge on the relative location between them is available. This is a non-trivial problem involving team communication.

Table 2.4 – Localization problems taxonomy (Thrun et al. [178]).

Spatial cognition in the brain

In the study of spatial cognition a primary distinction is established between *egocentric* and *allocentric* reference frames. According to Klatzky [98], in an egocentric reference system entities are represented with respect to the particular perspective of the agent. The allocentric reference frame would conform to the previously discussed definition by Thrun et al. [178] of the robot localization problem, so locations are expressed within a fixed framework that is external to the holder of the representation, thus, it is independent of the agent's position. Other important methodological distinction is established for the experimental study of spatial cognition. According to Freksa and Mark [132], the frame for *measurement* of the motion event may be different from the frame for the *representation* of motion.

Burgess [30] has reviewed the advances in the understanding of spatial cognition. Accordingly, spatial memory appears to include multiple representations for the tasks of both egocentric and allocentric types. Thereby, spatial memory and imagery are described as a mechanistic process, where different brain regions intervene. Thus, the hippocampus and medial temporal lobe would provide allocentric environmental representation, the parietal lobe would provide egocentric representation, and the retrosplenial cortex and parieto-occipital sulcus would allow both types of representations to interact. The way how representations are combined is still a matter of debate, that may also deppend on the nature of the task. In this sense, Mou et al. [126] have suggested that individuals use allocentric representations to learn spatial relations of objects for locomotion and reorientation, though egocentric representations are used when allocentric representations are not high fidelity. According to Graziano [76], in occular-manual reaching distinct egocentric representations may be employed, including eye- and body-centered.

Klatzky [98] has proposed a useful terminology to describe studies in spacial cognition, that is going to be adopted in this work. The terminology is based on the following assumptions:

a) Objects have and intrinsic front side.

b) Objects (including the agent) normally move forward with respect to their front (unlike crabs, that regularly move laterally).

c) Motion is performed on a plane surface.

Figure 2.11 illustrates some spatial relations of interest to describe 2D motion. *Spatial parameters* are values that can be assigned to individual points (e.g., the location of a point) or multiple points (e.g., distance between two points). *Primitive* parameters are spatial representations directly conveyed for the entities included in the representation. *Derived* parameters are computed from primitives, possibly in several computational steps.

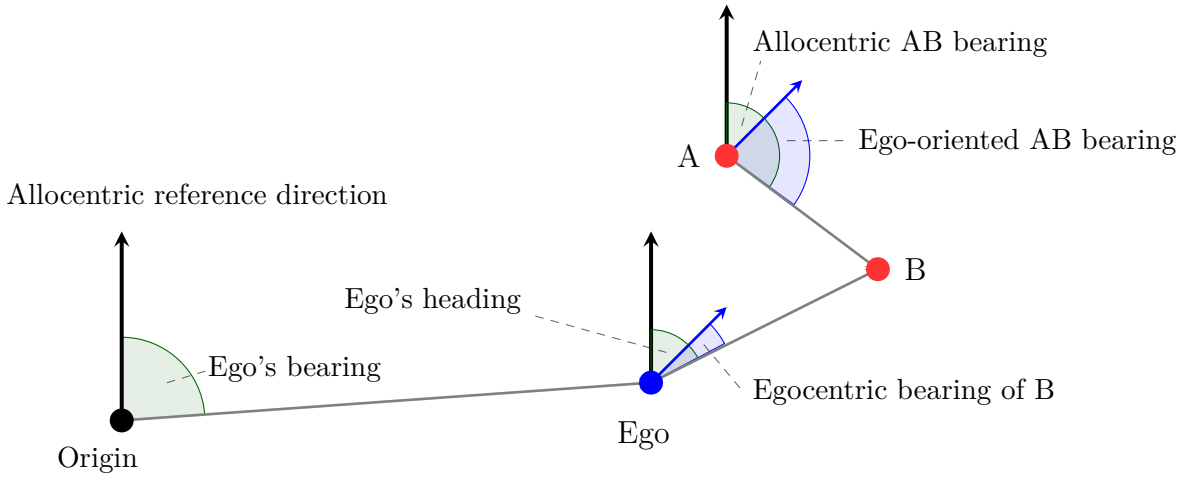


Figure 2.11 – Spatial representations. A fixed allocentric reference is represented as a black dot and a vector direction. In red two objects are represented, in blue the agent is represented with the heading direction. The angles in green correspond to allocentric description, whereas the angles in blue correspond to egocentric descriptions.

Points are spatial locations for which the values of the primitive parameters are known. An *object* comprises multiple points that are organized into a coherent entity. The *axis of orientation* of an object is a line between points on the object that defines a canonical direction in space. Not all objects have an axis of orientation; for example, an object that is radially symmetrical has none. The axis of orientation of a person within a space is aligned with the sagittal plane (as shown in Fig. 2.12). A distinction can be established between the axis of orientation of the head and the body.

The *heading* in space is the angle between the object's axis of orientation and some reference direction external to the object. The heading of a moving object can be differentiated from its course, or direction of travel as defined over the past few locations that were occupied. Because the reference direction is external to the object (a heading that was defined relative to its own axis of orientation would always be zero), heading will sometimes be referred to as allocentric heading.

The *bearing* between two points is defined with respect to a reference direction. The bearing from point A to point B is the angle between the reference direction and a line

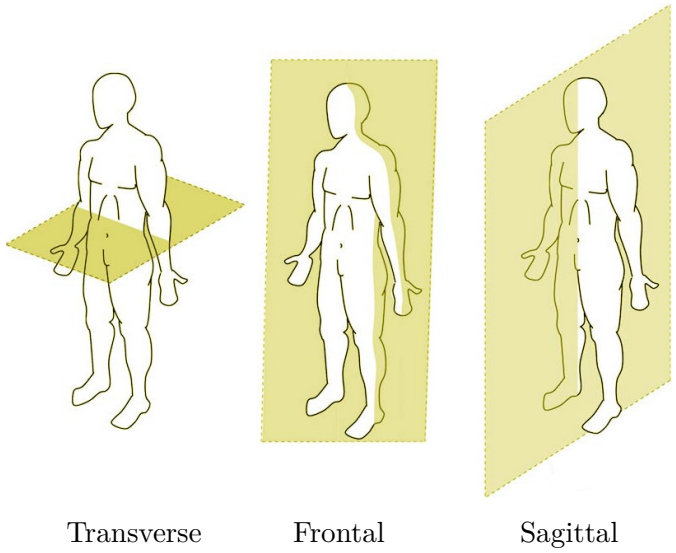


Figure 2.12 – Illustration of human planes of motion.

from A to B. If the reference direction is aligned with the axis of orientation of an "ego" (i.e., an oriented organism in the space), the bearing from A to B will be called ego-oriented. If any other reference direction is used, the bearing from A to B will be called allocentric. The egocentric bearing of a point B, is equivalent to a bearing from ego to B, using ego's axis of orientation as the reference direction. Thus, the egocentric bearing is a special case of the ego-oriented bearing in which ego's location is the source point. The egocentric bearing of B is numerically (but not conceptually) equivalent to the difference between B's allocentric bearing from ego and ego's allocentric heading, when both are defined with respect to a common reference direction.

The action selection problem

Robot navigation tasks involve several sources of uncertainties. One is related to inaccuracies in the observation of the system state, given the measurement noise and incomplete knowledge about the environment. Other is the stochasticity in the actuation system, so the outcome of motion may differ from what is expected. Lastly, disturbances independent of the agent, in the form of environment changes (e.g. slippery floor, uneven illumination, windy weather conditions, and so on), may also affect the task.

In the field of autonomous robotic systems (hereafter agents), a fundamental issue is to decide what to do next. An agent should maximize its expected utility, which is a function of its goals and priorities. Though, due to important constraints, such that environmental complexity, unpredictability, limited response time and resources; the selection may not be optimal. According to Pirjanian [145], this is denoted in the literature as the *action selection problem* (ASP), that is, the problem of resolving conflicts between competing behavioral alternatives.

As pointed out by Prescott [149], initial works in AI viewed the ASP as the execution of the steps of a plan that would lead the way from the current state to the desired goals. This plan was thought to be optimal, obtained from a formal process of search, acquired from imitation learning, or derived from a set of social norms. However, an influential study by Chapman [41] suggested that even refined planning techniques would ultimately

turn out to be unusable in any time-constrained system. In Zhao's [191] opinion, Chapman exerted a profound impact on subsequent planning research, and put into question the whole symbolic AI paradigm.

Brooks' research in the MIT (Brooks [29]) strongly contributed to a shift on the emphasis towards hand-coded systems with minimal on-board search. Thus, the ASP was studied from a modular and hierarchical decomposition of the task for obtaining tractable solutions. Therefore, instead of focusing on planning, the central issue was the integration of disparate, distributed, and parallel functionalities, in order to obtain coherent behavior.

Several authors (e.g. Zhao [191], Pirjanian [145], Murphy [131]) have pointed out three different approaches in the study of the ASP. Despite named differently, as illustrated in Fig. 2.13, available models are more commonly classified into *deliberative*, *reactive* and *hybrid*. However, there is no agreement in this classification. Thus, Mataric [116] has distinguished between hybrid and *behavior-based* architectures. Next the main characteristics of these approaches are discussed.

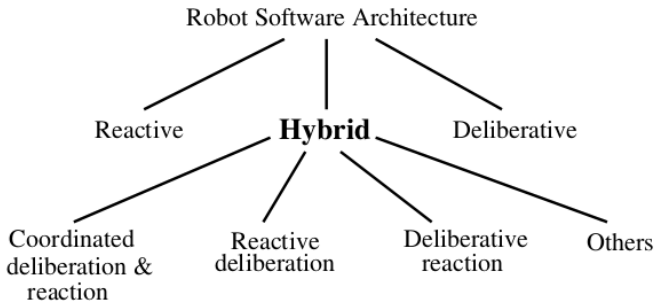


Figure 2.13 – Architectures for mobile robot control. Classification of mobile robot control architectures in Pirjanian [145].

Deliberative models

Deliberative models are related to the cognitivist AI research. The emphasis is put on a global world representation. As shown in Fig. 2.14, the model is based on three sequential operations. Action in the environment is derived from a plan, this means that the robot first senses, then thinks, and then acts. Thus ASP is treated as a centralized process. Since action is planned before execution, an advantage of this approach is that it can produce optimal behavior (e.g. the most efficient route to a goal). Though, the computational pipeline usually requires a significant amount of time, so becoming an architectural bottleneck. Moreover, the task representation must be precise enough as to provide planners with sufficient knowledge to choose optimal actions, which can be difficult to obtain. Another important disadvantage is the difficulty for the architecture to handle uncertainty in the task model (due to sensor noise, environment changes, etc.).

According to Murphy [131], as robotists began to study biological intelligence, they realized that the deliberative logic-based approach was inadequate for navigational tasks requiring a rapid response time to an open world. Thus, solutions based on deliberative models have been employed under controlled environmental conditions. Perhaps two best known deliberative models are the *nested hierarchical controller* (NHC) by Meystel [120], and the *NIST realtime control system* (RCS) by Albus & Proctor [3].

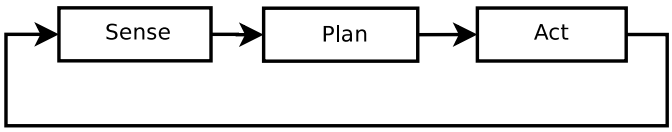


Figure 2.14 – Deliberative models rely on a central representation of the world so the solution can be planned (Murphy [131]).

Reactive models

Reactive models were originally proposed by Brooks [27]. The fundamental attribute of the reactive paradigm is that all actions are accomplished through behaviors, which are a direct mapping of sensory inputs to a pattern of motor action. Thus, behaviors would be equivalent to a transfer function, that transforms sensory inputs into actuator commands. As shown in Fig. 2.15, in reactive models *sense* and *act* are tightly coupled processes, so the overall behavior of the agent emerges as the result of their conjoint operation. Thus, sensing is local to each behavior so there is no global representation of the task (see Fig. 2.16 for a comparison with the deliberative approach).

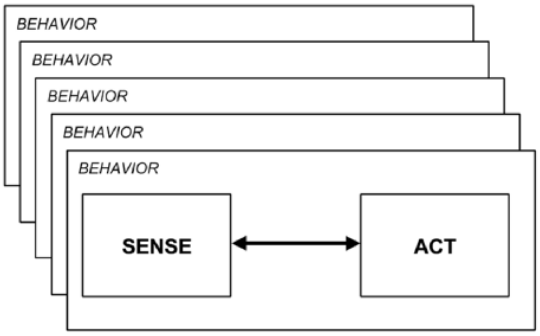


Figure 2.15 – The reactive principle (Murphy [131]). Models are characterized by a close coupling between perception and action. The observable behavior emerges from the concurrent execution of specific tasks.

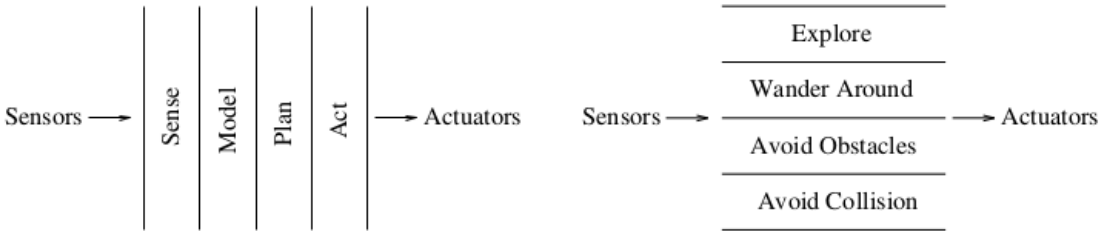


Figure 2.16 – Comparison between deliberative and reactive models (Pirjanian [145]). On the left, the deliberative pipeline between the sensory input and the actuation on the environment. On the right, the concurrent execution of specialized programs so behavior emerges out of their conjoint actions.

Several advantages are associated to these models. Behaviors are inherently modular and easy to test in isolation from the system, which complies with software engineering principles. The approach also conduces to the development of fast response systems (that is convenient to robot navigation problems), since the tight coupling between sensing and acting allows agents to operate in real-time. Behaviors can be implemented directly

in low-cost commercially available hardware, or in low computational complexity algorithms. Nevertheless, there are some disadvantages. Given the simplicity of behaviors and the fact that reactive systems have no memory, they are limited to the level of abstraction of stimulus-response reflexes, which may result in local minima and inefficient motion in complex task scenarios. Also, according to Brooks [29], an important challenge encountered in the study of emergent behavior is to find efficient ways to fuse multiple sources of perceptual information when needed (e.g. when sensing more elaborated events). The complexity of the model can augment significantly as the tasks becomes more complex. Perhaps the more restrictive limitation for service robotics applications is that reactive models cannot be directly commanded to achieve a goal in a particular manner, since there is not a global representation of the task available.

Hybrid models

According to Orebäck & Christensen [138], neither the purely reactive nor deliberative models can perform well when solving complex tasks. Although reactive models were successful in producing robots operating in real-time (which is a limitation of deliberative models), that came at the cost of preventing planning or other functionalities related to the optimal solution (e.g. remembering, reasoning about the global state, etc.). Therefore, hybrid architectures have been increasingly used since they share both desirable properties. By the one hand they are reactive, so they can respond in real-time to changes in dynamic environments. By the other hand they provide deliberation, so actions can be planed ahead in time.

As shown in Fig. 2.17, hybrid models are characterized by the *plan* and *sense-act* principle. The plan component includes all deliberation and global world modeling. Thus, the robot plans how to accomplish a mission and activates at each time the set of behaviors (i.e. sense-act) related to the execution of specific subgoals. The selected behaviors would remain active until completion of the subgoal, then the planner would activate a new set of behaviors according to the subsequent objectives of the plan, and so on. Hybrid models employ asynchronous processing techniques (e.g. multi-tasking, threads, etc.), thus deliberative functions are executed independently of reactive behaviors. For example, the planner computes the next goal for a robot to navigate to, while it is reactively navigating toward its current goal.

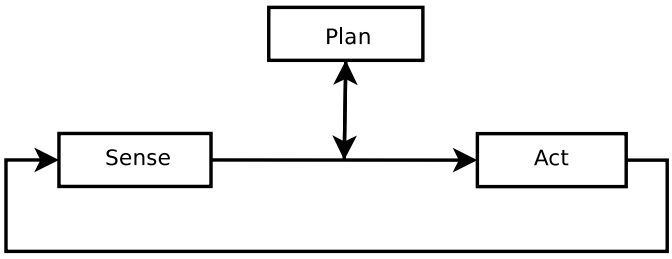


Figure 2.17 – The hybrid model principle (Morphy [131]). The sequence implies that, the robot first plans how to accomplish a mission or a task based on a global world model, and then activates a set of behaviors to fulfill the plan that is executed until it is completed.

According to Murphy [131] there are three main types of hybrid architectures: *managerial*, *state-hierarchy*, and *model-based*. The first type (e.g. AuRA by Arkin [8]) presents a bottom-up organization. At the top are agents which do high level planning, then pass off the plan to subordinates who refine it, gather resources, and then transfer those down

to the lowest level workers which are reactive behaviors. For example, let us suppose a driving task where the robot is off-road at the moment, so it has to advance through a small portion of irregular terrain to get back to the road, and then to follow the road until the next intersection. There are two subgoals that require of specific skills (off-road and path following navigation). Notice that, despite a map may be available, the trajectory is not explicitly given to the agent, so the overall behavior would emerge as a function of the two behavioral modes selected in the bottom layer. From the point of view of the task representation, managerial types would be the closest to pure reactive models, with deliberative functionalities added on the top of the architecture.

The state-hierarchy type (e.g. 3T by Bonasso et al. [22]) distinguishes between deliberation and reaction by the state or scope of knowledge. Reactive behaviors are viewed as having no state, no self-awareness, and operate in the present. Deliberative functions are categorized into those that require knowledge about the past (e.g. the previous localization) and the future (e.g. the mission, path planning, etc.). This type of models is more complex than the previous one, since the task representations are more elaborated, and the possibility of learning is considered. Sequences of behaviors can be managed for instance by remembering what the robot has already done and the success obtained. The planner layer can also process state information to predict the future. Therefore, the overall behavior of the agent in these models would emerge from the sequencing of behaviors, rather than the pure concurrency.

The model-based type (e.g. Saphira by Konolige & Myers[101]) is characterized by a top-down organization, focusing on the creation and maintenance of a global task model. Both specific-sensing and virtual behaviors can be defined. Goal coordination between behaviors is also ensured (which is not considered by the reactive paradigm). The communication with the user or other robots is based on absolute references since the global representation is shared. Thus, this type is conceptually close to pure deliberative architectures, though the task representation is generally less ambitious, and deliberation activities are usually implemented distributed among independent software agents. This provides a high degree of flexibility and computational efficiency. In fact, the programs do not have to run on-board so the processing bottleneck is mitigated.

In the hybrid types reviewed, managerial and state-hierarchy models seem to have evolved from the reactive models, whereas model-based is more close to the deliberative models. In this work global representations of the task are not studied, so the model-based type is of less relevance. The principle of the managerial type is used in Chapter 5 to define a supervised architecture, to obtain reliable approach to an object of interest under saliency ambiguity.

Behavior-based models

According to Mataric [116], behavior-based architectures are derived from the philosophy behind reactive models, though computations are not restricted to look-up or simple functional mappings. Thus, a behavior has a different meaning from reactive models, where it is given the connotation of a purely reflexive action. Here the term "behavior" is more consistent with the ethological use and includes reflexive, innate, and learned behaviors (i.e. close to the notion of a skill). For this, the behavior can implement various types of state representations providing local persistence. Since behavior-based models do not necessarily require of deliberative processes (which is essential to the hybrid model), their scope may include reactive models with internal state representation, and for the

case of deliberative functionalities, the types managerial and state-hierarchy according to Morphy's categorization of hybrid architectures. This flexibility is a desirable property of the behavior-based models, so different task can be studied. In Chapter 6 the *integrated behavior-based control* (iB2C) framework by Proetzsch et al. [152] is selected for studying various topics of visually-guided walk tasks with Nao.

Conclusions

This chapter has started by presenting an overview on the state of the art of the research in humanoid robotics. The historical review has pointed out to a research field that has been developing in the last decades. Initial contributions started in Japan and the USA, and more recently spread to many other countries. Funding has come from the public sector in the form of research projects, for the development of solutions in diverse domains (e.g. health-care, military, assistance, space exploration, and service). From the Honda's experience with the Asimo series, investments in the private sector have been cautious due to non immediate commercial profit. At present, the humanoid robot has not reached the level of massive consumption as for instance mobile and personal computers did. In practice, humanoid industry has mostly produced robots for research labs, followed by the ludic and the entertainment domains.

The exploration of diverse challenges encountered in humanoid research has shown that, compared to the human being, the current capabilities of these robots are limited. By the one hand, although designed with an anthropomorphic body, there are important physical differences related to the kinematic properties, the sense organs, and the actuation system; that impose restrictions to humanoids. By the other hand, the cognitivist AI approach has not been able to provide the robot with adaptability, given the stochasticity of unstructured environments. Thus, despite the many advances obtained in the control of locomotion, manipulation, or adaptation; the field is still waiting for technological and scientific breakthroughs, to reach the maturity required for reliable operation under unstructured scenarios. However, more recently some progresses have been achieved under the embodied cognition perspective. Embodiment is considered as of central interest for obtaining adaptation, autonomy, and learning; among other desirable qualities. These aspect are of great concern for this work. In Chapter 5, inspired by embodied cognition research, a methodology is proposed for obtaining reliable visual object approaching.

The literature on robot navigation was reviewed so the problems of localization and task representation were discussed. A taxonomy for classifying robot localization topics was provided. This work explores the topics of single-robot navigation in a static environment, passive localization, and robot kidnapping; through several study cases in Chapters 4 - 6. The review on studies in spatial cognition has suggested that multiple representations (both egocentric and allocentric) may coexists in the same task. The computational complexity and reliability of the task parameters are related to the definition of the measurement and the representation frames of reference. This work focuses on ego-centric representations obtained from visual information. For this, in Chapter 3 machine vision algorithms are studied for extracting features from images. The definition of a perceptive ego-cylinder for localization is presented in Chapter 4, so different placements for the measurement and the representation frames of reference are studied.

The topic of navigation in unstructured situations has also motivated the research of different solutions for the action selection problem. This is a fundamental aspect for an

agent that consists in deciding what to do next. Given the stochasticity of the task, deliberative, reactive, hybrid, and behavior-base approaches have been proposed. This work is interested in the behavior-based type (though depending on the terminology adopted, it may also include reactive models, and the managerial and state-hierarchy types of hybrid models). Therefore, the tasks under study are modeled as a distributed system, where there is no global representation available (since centralization of computation has been reported as an important weakness of deliberative models). The principle of managerial type is used in Chapter 5, to define a supervised architecture for reliably approaching a known object under saliency ambiguity. In Chapter 6, by adopting the behavior-based formalism iB2C, several topics (e.g. emergent behavior, obstacle avoidance, and learning) are investigated.

3

Visual attention

Contents

3.1 Introduction 33
3.2 Theories of attention 34
3.2.1 The filter theory 34
3.2.2 The spotlight theory 36
3.2.3 The FIT and GS theories 37
3.2.4 Inspiring robotics solutions 37
3.3 Machine vision 38
3.3.1 The camera sensor 39
3.3.2 The human eye 40
3.3.3 Perspective projection 41
3.3.4 Visual feature extraction 42
3.4 Case studies 48
3.4.1 Materials and resources 48
3.4.2 CS-I: Semi-automatic color-based segmentation 48
3.4.3 CS-II: Top-down color-based segmentation 50
3.4.4 CS-III: Bottom-up segmentation based on optical flow 55
3.5 Conclusions 62

Introduction

In the study of autonomous behavior, an aspect of crucial importance is the efficient processing of information, since robotic agents have limited computational and storage capacity. Different human-inspired sensory modalities (e.g. vision, touch, audition, etc.) and supra-human modalities (e.g. laser range, sonar, etc.) can provide humanoids with data about the environment. Here the focus is placed on vision. Thus, from visual attention processing, the robot should be able to detect feature saliency from images of

the scene captured on-board. For this, the literature in the field in reviewed, including both proposals derived from research in cognitive science (i.e. bio-inspired solutions), and from the field of machine vision (e.g. solutions based on mathematical modeling).

Thereby, this chapter starts by introducing relevant theories of human attention that can be (or have already been) used for robotics tasks. Next benchmark visual sensor technologies are reviewed and contrasted to the human eye, in order to illustrate potential challenges for robot vision. A review on the state of the art in machine vision follows, where the whole scene segmentation and the feature tracking approaches are discussed. Three case studies are developed so potential methods for obtaining top-down (i.e. supervised) and bottom-up (i.e. unsupervised) visual selection are prototyped and evaluated. The first two consider whole scene segmentation based on color information (one is top-down, relying on a color model of the object, and the other is bottom-up, based on heuristic clustering). The third study considers feature tracking from dense optical flow, to obtain an unsupervised estimation of the scene structure. The possibilities of employing these approaches for on-board visually-guided walk tasks are evaluated.

Theories of attention

According to Quinlan & Dyson [153] attention is the process whereby the individual can select from among the many competing stimuli present in the environment, facilitating processing of some while inhibiting others. As pointed out by Pinto et al. [144], this selection can be driven endogenously by goals (also called *top-down* or *goal-driven*), or exogenously by a salient or novel stimulus that captures attention away from the task at hand (also called *bottom-up* or *stimulus-driven*). These two attentional systems seem to operate independently. Thus, the balance between endogenous and exogenous factors not only allows the accomplishment of the goals (e.g. finding a particular object of interest on a supermarket's shelve), but also to be sensitive to important external information (e.g attending to a fire alarm or to the sound of a crashing glass).

Because there is too much information at any given moment for the individual to cope with, the attentional mechanism ensures that relevant or important information obtain further processing. In this sense, conforming to Smith & Kosslyn [168], in view that the human being is capable of processing a limited quantity of information in both space and time, the attentional process ensures that the selection occurs in a convenient and not a random fashion. A number of different information-processing theories have attempted to explain the human attentional process. Although none of them can cope with all the aspects related to the attentional phenomena, they certainly handle particular aspects of attention. Some of the most important theories are: the *filter*, the *spotlight*, the *feature integration*, and the *guided search* theory.

The filter theory

Broadbent [26] viewed the attentional system as containing a limited-capacity channel through which only a certain amount of information can pass. Accordingly, as illustrated in Fig. 3.1, the many sensory inputs entering the cognitive system at a particular moment are filtered out, such that only the most important information gain access to semantic processing. In other words, information would be pre-processed in a pre-attentive sensory store, and only sensory events characterized by relevant physical properties would pass

trough the limited capacity processing system. The items blocked by the filter would vanish from the store within a matter of seconds. Thus, the theory suggests that an information bottleneck would occur immediately after the sensory store.

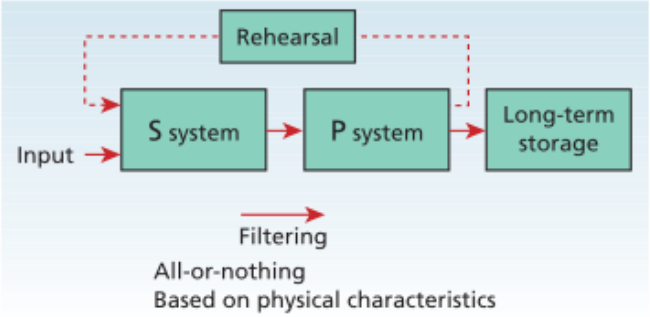


Figure 3.1 – Illustration of the filter model (Quinlan & Dyson [153]). The S (or sensory) system comprises many input channels that deliver information continuously (and in parallel) as stimulation from the outside world impinges on the body. Items that are selected for read-out by the filter are passed on to the limited capacity channel, shown as the P (or perceptual) system in the figure. The P system is assumed to operate serially (items are processed one at a time). Since items decay from the store within a matter of seconds if they are not read out by the filter, the Rehearsal system passes back information from the P system and rehearsed such that it is recirculated into the short-term store in case where the system is in danger of information overload. Only items that have been selected and have exited from the P system stand any chance of entering the long-term memory system.

Broadbent's ideas have received experimental support (see Cherry [44]). In studies of *dichotic listening* the subject is exposed to different messages at each ear and required to attend to one in particular. Generally, subjects are unable to remember the unattended message, even tough it was systematically repeated throughout the trials. However, broaden research on the so-called *cocktail party effect* (see Arons [12]), which is the ability to focus one's listening attention on a single talker among a cacophony of conversations and background noise; has pointed out to some limitations in the filter theory. That is, certain subjects are able to recognize unattended information (e.g. the person's name, words as "fire", etc.) even though the speaker voices are kept constant in the experiment (i.e. no physical novelty), which is considered to be evidence against perceptual processing occurring only before the information bottleneck (also called late-selection). In other words, both physical characteristics and semantic content seem to account for how unattended but high-priority information can still be detected.

Despite criticism, an important contribution of the theory is to promote the debate itself over whether attention operates at an early or late stage, which has highlighted two important aspects of attention. The first one, is that attention can have an effect on the very earliest levels of perceptual processing by reducing the amount of information that enters into the cognitive system. The second one, is that some unattended information reaches very late stages of processing, which shows that the information is not entirely filtered out. Contents related to the context of the current goal or likely to be of extreme importance, can penetrate the attentional filter.

The spotlight theory

A study by Posner et al. [147] has shown that the subject's knowledge about where in space a stimulus will occur affects the efficiency of detection. Consequently, spatial attention would act like a spotlight by highlighting information within the beam region. Information within such circumscribed region of space is selectively brought to awareness, and outside such region it is more likely to be ignored.

Olshausen et al. [137] have proposed an implementation of a biologically plausible model of an attentional mechanism based on the spotlight metaphor, to represent position- and scale-invariant information of visual objects. As ilustrated in Fig. 3.2, in the model control neurons dynamically modify the synaptic strengths of intracortical connections, so that information from a windowed region of primary visual cortex (VI) is selectively routed to higher cortical areas. The selection mechanism provides a computational advantage, because most processing is limited to the small selected region, which considerably simplifies the connection circuitry that would be necessary to cope with the entire visual field at once.

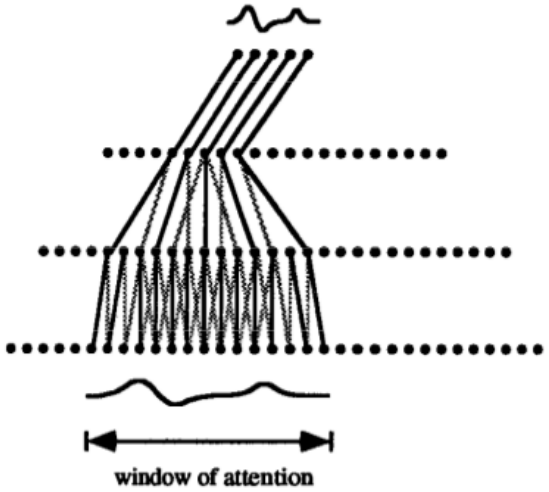


Figure 3.2 – A simple, one-dimensional dynamic routing circuit (see Olshausen et al. [137]). The model relies on a set of control neurons to dynamically modify the synaptic strengths of intracortical connections, thus information from a windowed region of primary visual cortex (VI) is selectively routed to higher cortical areas.

An important contribution of the spotlight metaphor for attention is the idea that space is a powerful coordinate system for the perceptual systems, and that attention may directly operate on these sensory systems. For example, turning toward the spatial source of a noise might result in the incidental selection of other objects that otherwise would have been failed to be noticed.

Nevertheless, there are some criticism to the model. According to Cave & Bichot [34], despite the spotlight metaphor has been inspiring much of the research in visual attention, it is no longer able to account for the level of complexity of recent theories and models of visual selection. Furthermore, according to Smith & Kosslyn [168], a major problem of the model is to explain results from studies suggesting that attention can be directed to a single object, even when superimposed on another object. This contradicts the idea that attention indiscriminately highlights information in a particular spatial region, since if that were the case all objects would have been selected together.

The FIT and GS theories

The feature integration theory (FIT) is mostly concerned with the role attention plays in selecting and binding complex information. Consequently, it takes some distance from the ideas of bottleneck, filtering, and the spotlight metaphor. According to Treisman & Gelade [179], the perceptual system is divided into separate maps, each of which registers the presence of a different visual feature (e.g., color, edges, shapes, etc.). That is, the image is decomposed into low level attributes across several spatial scales, which are combined to form a master saliency map. When the searched object is defined by a single feature (e.g. by its shape) only such map would be consulted to detect the object, thereby, a *disjunctive* search is produced. In case where the searched object would combines features, a joint consultation of corresponding maps would be required, such that, a *conjunctive* search is produced. According to the theory, disjunctive search is pre-attentive, whereas conjunctive search does involve attention.

Koch & Ullman [99] have proposed a biologically plausible architecture within the FIT conceptualization. As illustrated in Fig. 3.3, an implementation of the model has been provided in Itti et al. [89], and has been used for visually-guided autonomous navigation (e.g. Siagian et al. [166], and in García et al. [72] for the Robocup, among others). Some neuroimaging studies have provided evidence for the distinction between disjunctive and conjunctive processes (see Smith & Kosslyn [168]), that is, different types of features appear to be handled by partially distinct neural mechanisms. However, evidence from hemispatial neglect patient research has challenged the FIT assumption that disjunctive search does not engage attention. Moreover, behavioral studies with neurologically unimpaired participants have found that some conjunctions are easier to detect, contrarily to a purely serial search as predicted by the model.

The guided search (GS) theory evolved out of the FIT architecture, and it is currently in its fourth revision (see Wolfe [189]). The idea behind is that the output from a first stage of information processing can guide later search mechanisms. Although the first stage is similar to FIT by including feature maps, it differs in that items that cannot possibly be the target are eliminated in parallel in the feature maps. Thus, by the time information reaches the second attentive stage, the number of candidate targets is already much reduced, when compared to the total number of items possessing a particular feature of the target. In general, major contributions of FIT and GS are the description of the mechanism of information integration, and a more complex model of attention involving early pre-attentive and later attentive stages of processing.

Inspiring robotics solutions

The models of attention reviewed have inspired the current study in several ways. As it is discussed in Chapters 4 - 6, in agreement with the filter theory data is pre-processed for obtaining more efficiency, so only relevant information gains access to more complex processing stages (i.e. early selection). Based on the idea that space is a powerful coordinate system for perception, so attention may directly operate on the sensory native space; in Sec. 5.3.3 the spotlight metaphor is adopted to propose an embodied mechanism (i.e. the Embodied Filtering task) that is in charge of selecting the retinal data related to the object of interest, under top-down saliency ambiguity. Moreover, inspired by the models FIT and GS, the idea of combining multiple layers of image features is adopted in Chapter 6, so top-down and bottom-up saliency features are used to control

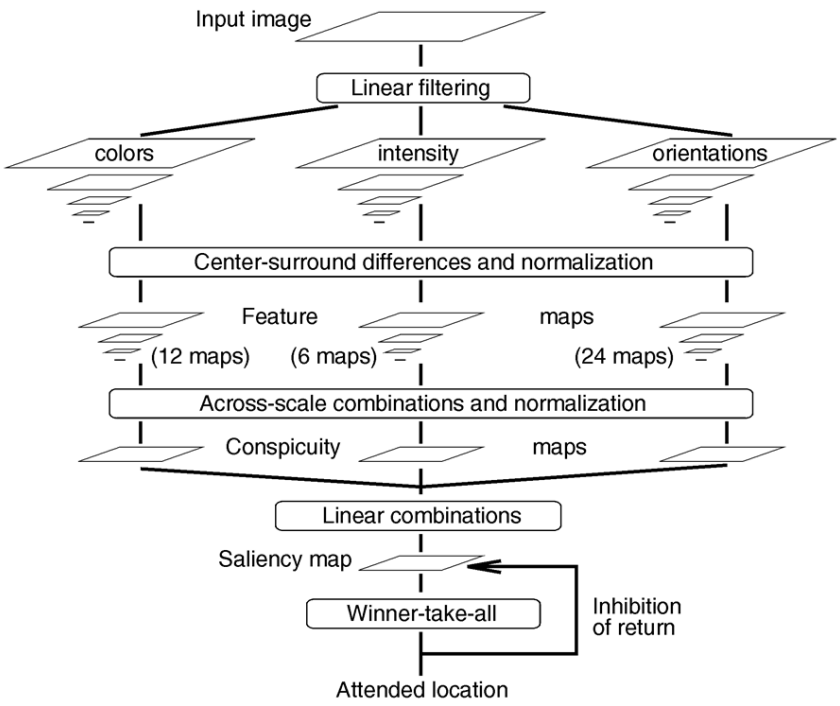


Figure 3.3 – Implementation of the Koch & Ullman architecture by Itti et al. [89]. Visual input is first decomposed into a set of topographic feature maps. Different spatial locations then compete for saliency within each map, such that only locations which locally stand out from their surround can persist. All the feature maps feed, in a purely bottom-up manner, into a master saliency map, which topographically codes for local conspicuity over the entire visual scene. Finally, the model's saliency map is endowed with internal dynamics which generates attentional shifts.

the robot walk to reactively approaching an object or avoiding obstacles. However, differently from these models that appear to consider attention as a synchronous process (this is a key aspect of information processing models); in this work attention is also studied as a distributed and asynchronous process, so parallel sensory schemes observe specific features, by considering one or more acquisitions (of visual, proprioceptive, and inertial sensory modalities). In the next section the literature on machine vision is reviewed to explore available techniques for extracting information from digital images. As it will be discussed the artificial sensor operates in a much different way that human vision works. This imposes several constraints to the development of robotic solutions based on vision.

Machine vision

Computer vision is the application of a computer system for receiving and processing visual information. There are excellent books available in the field of image processing (e.g. Stockman & Shapiro[172], Jähne [90], and Gonzalez & Wood [74]) and robot control based on vision (e.g. Siciliano & Khatib [167] and Hyungsuck [46]), though one that clearly relates both fields in a theoretical and practical way, and certainly is a reference on the domain, is the work by Corke [52]. According to Corke, there are two main research areas in the field: *image processing* and *image interpretation*. The former includes techniques for enhancing the quality of the image for visualization (e.g. motion blurs removal in

Pretto et al. [150]), examples of applications are remote sensing and medical imagery. Alternatively, image interpretation - also known by *scene understanding* or *machine vision* - is the problem of describing physical objects in a scene, given an image (or several images) of that scene. Machine vision techniques are based on the definition of numeric or image features, which reduce the dimensionality of the sensory space in order to derive simpler datasets, from which manageable solutions can be obtained by a computer system. In order to illustrate potential challenges encountered in image interpretation, benchmark visual sensor technologies are firstly reviewed and contrasted to the human eye, then the perspective projective model (typical of conventional cameras) is presented, and the extraction of visual features including techniques for *whole scene segmentation* and *object tracking*, are reviewed.

The camera sensor

According to Corke most of the research in vision-based control has employed some form of solid-state imaging sensor of the type CMOS, NMOS, CCD or CID. These sensors comprise a number of discrete photosites (or pixels), where each site accumulates a charge proportional to the illumination of the photosite integrated over the exposure period (see Fig. 3.4). The charge coupled device (CCD) sensors are considered to produce better-looking images with less visual noise and distortion, but they consume more energy and provide slower data-throughput speed.

The most significant difference between CCDs and other types is that all photosites are sampled simultaneously, when the photosite charge is transferred to the transport registers. For other modalities sensor pixels are exposed over the field-time during the reading (which may result in the effect known by *rolling shutter*, where the image is skewed depending on the direction of camera or object motion). The robot Nao is equipped with complementary metal-oxide semiconductor (CMOS) sensors, which are also widely used in mobile devices, given their less manufacture cost compared to CCDs.

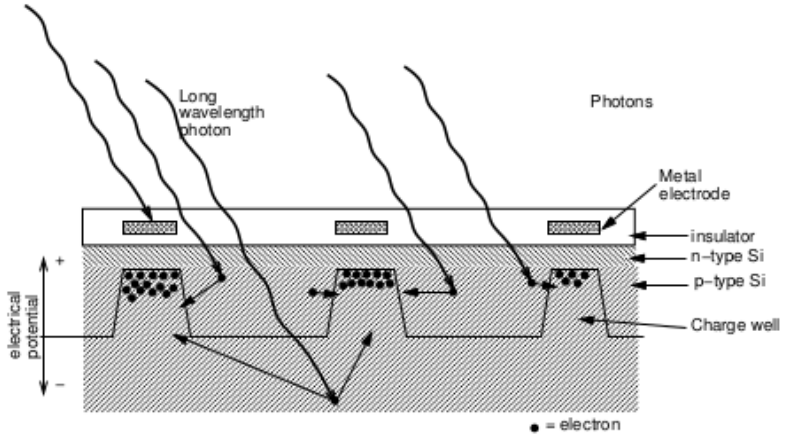


Figure 3.4 – Photosite charge wells and incident photons (Corke [52]). Silicon is more transparent at long wavelengths such that photons may generate electrons deeper within the substrate. This introduces cross-talk between pixels, where the pixel values are not truly independent spatial samples of incident illumination.

An important property associated to a signal is the *dynamic range*, which describes the range of the input levels that can be reliably measured simultaneously, that is, the ability to accurately measure small signals in the presence of the large signals (Halámek

et al. [77]). The most commonly used unit for measuring dynamic range in photography is the *f-stop*, which describes the ratio between the lightest and darkest recordable regions of a scene in powers of two (e.g. a scene with a dynamic range of 3 f-stops has a white that is 8X as bright as its black, since $2^3 = 8$). The largest signal at saturation is directly related to the capacity of the charge well. At very low illumination levels the response of the sensor is totally overwhelmed by the dark current and noise effects described in Fig. 3.4. The smallest discernible output is thus the output noise level. Commercial sensors normally have between 8-14 f-stops.

High-speed relative motion between the camera and scene results in a blurred image, since the photosites respond to the integral of illumination over the exposure period. As a consequence, a blurred object will appear elongated in the direction of motion. In the case where an object moves more than its width during the exposure interval, the illumination will be spread over a greater number of pixels and each will receive less light. That is, as the image blurs, it elongates and becomes more attenuated. Next, the human visual system is briefly described so a comparison with the artificial sensor can be established.

The human eye

The human eye presents a nearly spheric morphology with an average diameter of approximately 20 mm (Gonzalez & Wood [74]). A simplified depiction of the eye is presented in Fig. 3.5. Three membranes enclose the eye: the *cornea* and *sclera* outer cover, the *choroid*, and the *retina*. The cornea is a tough transparent tissue that covers the anterior surface of the eye. Continuous with the cornea, the sclera is an opaque membrane that encloses the remainder of the optic globe. The choroid lies directly below the sclera and provides nutrition to the eye trough a network of blood vessels. It is heavily pigmented and hence it helps to reduce the amount of extraneous light entering the eye and the backscatter within the optical globe. The innermost membrane of the eye is the retina, which lines the inside of the wall's entire posterior portion. When the eye is properly focused, light from an object outside the eye is imaged on the retina.

According to Corke [52], the human eye is different in several ways when compared to a artificial sensors. Light is sensed in the eye by two types of photoreceptors located in the retina: the *cones* and the *rods*. Cones are color sensitive activated in normal daylight conditions. Proportionally, they are distributed such that 65% sense red, 33% sense green, and only 2% sense blue color. The biggest concentration of cones (around 34,000) is located in the *fovea* region, so their density in the rest of the retina is considerably lower. Due to this, the eye presents high resolution of a few degrees only over the foveal field of view, but subconscious fixation point shifts (i.e *saccadic* eye motions) directs the fovea over the entire field of view. Rod sensors are activated at very low light levels. They are monochromatic and their density in the fovea is only 7% of the cones', but increases in the peripheral region. The distance between the lens and retina is approximately constant at 15 mm, so focusing is achieved by muscles which change the shape of the lens.

Cone photoreceptors have a dynamic range of 9 f-stops. Likewise the iris of a lens, the pupil varies in diameter from 2 to 8 mm which provides for a factor of 4 f-stops (3 f-stops in older people). Rods provide another factor of 5 f-stops. Rod sensitivity is chemically adapted with a time constant of tens of minutes. The overall dynamic range of the eye is thus approximately 18 f-stops. The eye has three degrees of rotational motion. The muscles that actuate the human eye are the fastest acting in the body, allowing it to rotate at up to 600 deg/s and 35,000 deg/s^2 for saccadic motion. Smooth pursuit eye motions

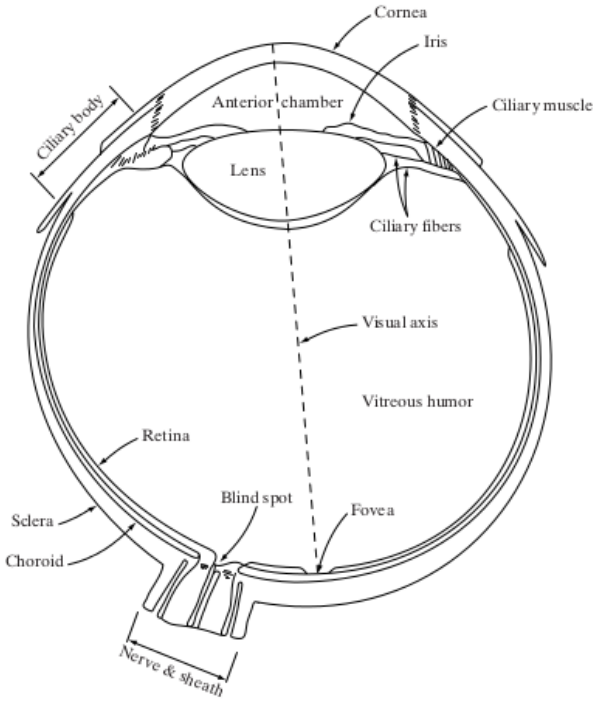


Figure 3.5 – Simplified cross section of the human eye (Gonzalez & Wood [74]).

involved in tracking a moving object operate at up to 100 deg/s. Rotation about the viewing axis (i.e. cyclotorsion), is limited and the maximum varies between individuals (ranging from 5 to 20 deg).

To summarize, the human eye presents several advantages with respect to artificial sensors. Since the retina is curved along the back surface of the eyeball, the edges of the retina would be about the same distance from the lens as the center (differently from 2D area sensors), so better sharpness at the corners of the image is obtained (this will be better understood when reviewing the perspective projective model used by conventional sensors in Sec. 3.3.3). However, punctual comparisons like these may be misleading, since the eye is a living organ and human vision is actually a dynamic process that takes place in several steps, so it would be comparable to a video and not to a photography. That is, the resulting mental image is more a reconstruction of the scene based on different sort of inputs provided by the eyes, that the mere registry of the actual light received by the sensor. This is extremely advantageous since the eye can compensate as it focuses on regions of varying brightness (that is why human night vision is much better than in artificial sensors, and the dynamic range is higher, see Fig. 3.6). It can also look around to encompass a broader angle of view, or focus on objects at a variety of distances. The dual eye overlap field of view region is around 130°(nearly as wide as a fish-eye lens). However, for evolutionary reasons the peripheral vision is used for sensing motion and large-scale objects, and not for high resolution vision. Thereby, the human eye is specialized in detecting different sort of events.

Perspective projection

Conventional cameras use perspective projection. As illustrated in Fig. 3.7, in the perspective transform the 3D space is mapped to the 2D image plane. A non-inverted image is formed on the image plane at $Z = 0$ from a viewpoint at $Z = -f$. Let a world

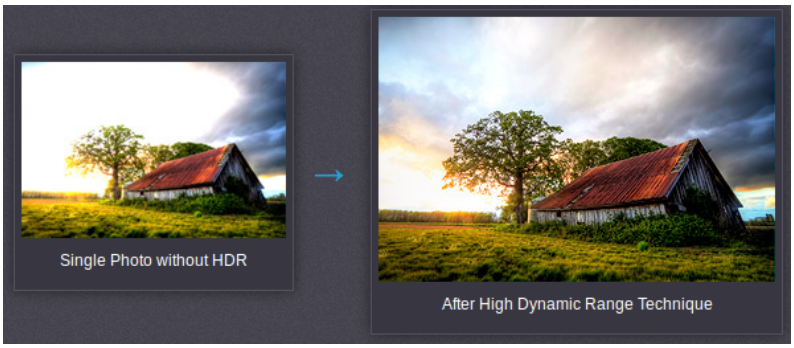


Figure 3.6 – The photographer chooses to take many pictures of the scene at a given exposure by changing either the shutter speed or the aperture. The images are processed in software that determines dynamic information such that the shutter speed and aperture, to calculate how much light actually came from each image region. On the left a single image of the scene, on the right the enhanced image[1] .

1. Available at http://www.cambridgeincolour.com/tutorials/high-dynamic-range.htm (accessed on 10/12/2015)

point in the 3D Cartesian space be denoted by the coordinates (X, Y, Z). Using similar triangles, it can be shown that the 2D coordinates (x, y) of its projection on the image plane is defined by

$$(x, y) = \left(\frac{fX}{f - Z} , \frac{fY}{f - Z} \right). \tag{3.1}$$

The projective-perspective transform has the following characteristics:

- World lines are mapped to lines on the image plane.
- Parallel world lines, not in the plane orthogonal to the optical axis, are projected to lines that intersect at a vanishing point.
- Conics in world space are projected to conics on the image plane, for example, a circle is projected as a circle or an ellipse.
- The mapping is not one-to-one, and a unique inverse does not exist. In general the location of an object point cannot be determined uniquely by its image. All that can be said is that the point lies somewhere along the projecting ray shown in Fig. 3.7. Other information, such that a different view, or knowledge of some physical constraint on the object point (i.e. it is known that the object is lying on the floor) is required in order to fully determine the object's location in 3D space.

Visual feature extraction

Image features are measurable relations in an image. According to Jang [91], these functionals are defined by

$$f = \int \int_{Image} h(x, y, I(x, y))dxdy, \tag{3.2}$$

where $I(x, y)$ is the pixel intensity at location (x, y). The function $h(., ., .)$ is a linear or non-linear mapping depending on the feature, and may also include Dirac delta functions.

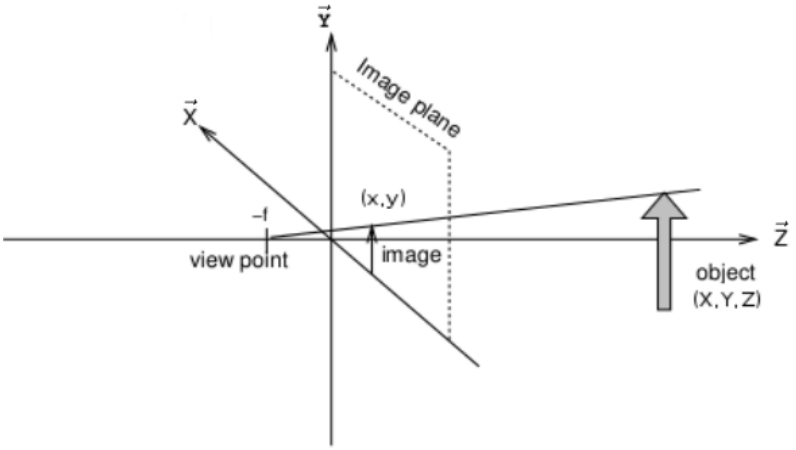


Figure 3.7 – Central perspective geometry (Corke [52]).

Many image features can be defined. For example, the lengths or orientation of line segments connecting distinct objects in the scene (e.g. holes and corners), and template matching for distinctive pixel patterns. *Moments* are easy to compute and very useful features. The $(p+q)^{th}$ order moments is defined by

$$m_{pq} = \int \int_{Image} x^p y^q I(x,y) dx dy. \tag{3.3}$$

The $(p+q)^{th}$ order moment for a digitized image is

$$m_{pq} = \sum_i \sum_j x_i^p y_j^q I(x_i, y_j). \tag{3.4}$$

For a binary image the function $I(x,y)$ is either 0 or 1, so the moments describe a set of locations and not the grey-level of those points. According to Hyungsuck [46], moments can be given a physical interpretation by considering the image function as a mass distribution. Thus m_{00} would be the total mass of the region, the centroid of the region would be given by

$$(x_c, y_c) = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right). \tag{3.5}$$

However, as illustrated in Fig. 3.8 when the object is not viewed along the surface normal, the centroid of the image does not correspond with the centroid of the object.

Translation-invariant central moments for a region R are computed about the centroid (x_c, y_c), such that

$$\mu_{pq} = \sum_i \sum_j (x_i - x_c)^p (y_j - y_c)^q I(x_i, y_i), \qquad (i,j) \in R. \tag{3.6}$$

Similarly, scale-invariant and orientation-invariant moments can be defined. A closed boundary R_b (i.e. the perimeter of the area) can be characterized by N_b pixels, so the normalized contour central moment is defined by

$$\bar{\mu}_{pq} = \frac{\mu_{pq}}{N_b}. \tag{3.7}$$

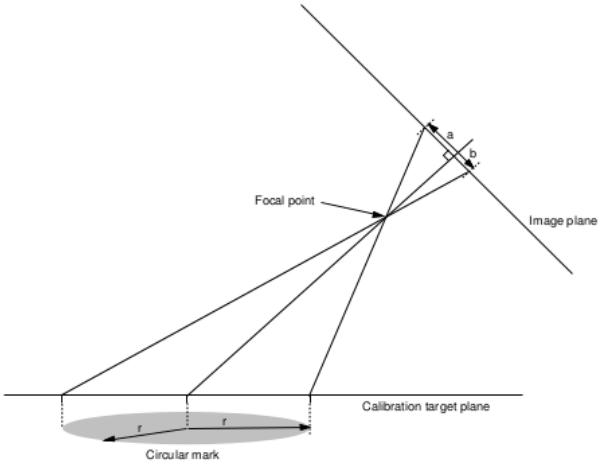


Figure 3.8 – Exaggerated view showing the centroid offset in the image plane (Corke [52]).

Contour moments are computationally less demanding and can be used for calculating the direction (or orientation) of a region. As shown in Fig. 3.9, the region can be represented by an ellipse. The direction of a closed elongated region (it is not defined for a circular region) would correspond to the angle θ between the elongated side and the positive x-axis of the image, such that

$$\theta = \frac{1}{2}\tan^{-1}\left(\frac{2\bar{\mu}_{11}}{\bar{\mu}_{20} - \bar{\mu}_{02}}\right). \tag{3.8}$$

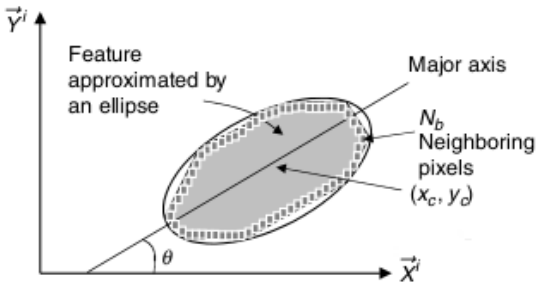


Figure 3.9 – Contour central moment (Hyungsuck [46]).

In general, in vision-based control research the definition and extraction of image features have relied on two main approaches: *whole scene segmentation*, and *feature tracking*. Some relevant techniques within these approaches are reviewed next.

Whole scene segmentation

According to Stockman & Shapiro[172], *image segmentation* is the process of partitioning an image into a set of regions that cover it, with the goal of representing meaningful areas (e.g. objects, people, urban areas, forests of a satellite image, and so on). When the regions of interest do not cover the whole image, the segmentation would partition the foreground from the background to be ignored. Figure 3.10 and Tab. 3.1 illustrate the contribution of image segmentation to a more general task of scene interpretation.

Pixel values may be scalar or vector quantities that can represent luminosity, color, range, velocity, or any other measurable property on the scene. A basic approach to do

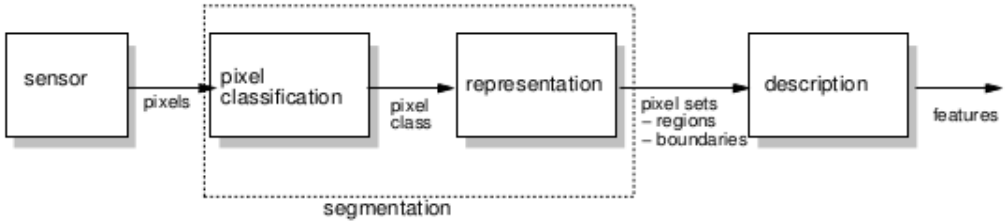


Figure 3.10 – Processing pipeline for scene interpretation (Corke [52]).

Step	Description
Classification	Pixels are classified into spatial sets according to low-level characteristics.
Representation	The spatial sets are represented in a suitable form for further computation (e.g. connected regions, boundaries).
Description	Sets are described in terms of scalar or vector features.

Table 3.1 – Description of the scene interpretation workflow presented in Fig. 3.10.

the classification step is to apply a threshold test T to individual pixels (also known by *thresholding*). Thus, a segmented image is obtained as follows

$$P_{ij} \in \begin{cases} S_\mathrm{b} & \text{if } I(x_i, y_i) < T \\ S_\mathrm{f} & \text{if } I(x_i, y_i) \geq T \end{cases}, \tag{3.9}$$

Where P_{ij} is the pixel, and the sets S_b and S_f contain respectively the pixels in the background and the foreground. The threshold T can be obtained automatically by processing the image histogram (i.e. in Otsu [140]). This technique is mostly employed in the context of the lab where the environment can be controlled (e.g. disposing bright objects over a dark background, using fluorescent lamps, etc.). For less constrained situations such that outdoor tasks, the performance is generally inadequate.

Jähne [90] has classified basic approaches of image segmentation into *pixel-based*, *region-based*, *edge-based* and *model-based*. Pixel-based methods rely exclusively on the value of the pixel to produce the segmentation. The advantage of these methods is that they tend to be simple to implement and computationally efficient. As a drawback, noise can be easily misclassified. In the study developed in Sec. 3.4.2, the clustering algorithm *k-means* by MacQueen [115] is employed to segment the image based on the pixel intensities. Region-based techniques analyze the values in larger areas, so the resulting segmentation is relative to a local vicinity or neighborhood (spacial coherence). Some examples are the *region growing* the *split-and-merge* techniques, and the NCA texture kernel (Ferreira et al. [70]). These methods are computationally more expensive, though the effect of noise can be more efficiently mitigated. The case study developed in Sec. 3.4.3 has employed the technique by Kato et al. [95] to obtain robust segmentation. Edge-based methods exploit the fact that the position of an edge is given by a peak on the first-order derivative of the signal, or a zero crossing in the second-order derivative. Therefore, these methods are conveniently employed to detect the borders of objects in the image (e.g. the edge detector by Canny [32]). Model-based segmentation relies on specific knowledge about the geometrical shape of the objects, which can be compared with the local information available in the image. Depending on the application a detailed model of the object may be required (e.g. the segmentation of magnetic resonance images based on a heart mesh, see Legrand et al. [106]). Though, in less constrained scenarios, heuristics on the shape of the object may also be employed (e.g. the detection of line and curves through the

hough transform, see Duda & Hart [57]).

Feature tracking

Feature tracking can be obtained under the principle of Verification Vision (VV), as proposed by Bolles [21]. VV algorithms assume that prior knowledge about the type, placement, and appearance of the objects is available to the system, such that the goal is to verify and to refine the location of such features in the scene. Determining the initial location of features requires the entire image to be searched, but this need only to be done once. Thus, this approach is analogous to a top-down search. A commonly used criteria to match features between consecutive frames is the least-squares fitting. Features are chosen on the basis of a confidence measure computed in a neighborhood. The tracking technique by Comport et at. [49] is based on this principle. From the knowledge of the geometry of the object, its initial pose, and the estimation of the spatial evolution of the camera; the features are predicted in the image plane, such that local search matches real measurements with the virtual features to track the object.

Less restrictive approaches in terms of modeling have also been proposed. The *continuously adaptive mean shift* (CAMShift) algorithm by Bradski [24] considers a color model of the object to perform heuristic optimization search in a local neighborhood. The technique was originally proposed in the context of user machine interface applications (e.g. to track the face of the user). Thus, in favorable conditions acceptable results are obtained. Though when variations in the point of view are introduced, the color model may no longer be useful to detect the object. An improvement is proposed by Exner et al. [64], and consists in the accumulation of multiple histograms to handle various perspectives of the object.

In the other extreme are model-less approaches. As pointed out by Bradski [23], the idea is to estimate motion between two frames without any prior knowledge but the changes induced by the motion itself. This is in fact the notion of *optical flow*, as originally introduced by Gibson [73] when working with visual perception. The optical flow (also named *optic flow*, or *flow field*), is the visual motion that results from an observer's own movement through the environment. According to Beauchemin & Barron [16], optical flow algorithms are useful for applications such that: recovering 3D motion of the visual sensor (to within a scale factor), recovering 3D structure of surfaces (the shape or relative depth) of the environment, motion detection, object segmentation, time-to-collision calculations, motion compensated encoding, stereo disparity measurement, and perhaps many others.

According to Gonzalez & Woods [74], the mathematical definition of optical flow is based on three important premises: a) the object reflectivity and illumination does not change during the interval $[t_1,\ t_2]$, b) the distances of the object from the camera or light sources does not vary significantly over this interval, and c) each small intensity neighborhood $R_{x,y}$ at time t_1 is observable in some shifted position $R_{x+\delta x,\ y+\delta y}$. Obviously, these assumptions do not hold tight in real imagery, though in some cases they can lead to useful computation of image flows. Using the brightness constancy constraint for the intensity function $f(x, y, t)$, the image flow can be defined by

$$f(x, y, t) = f(x + \delta x,\ y + \delta y,\ t + \delta t). \tag{3.10}$$

A Taylor series representation (including only the linear terms) in a small neighborhood of an arbitrary point (x, y, t) can be considered, such that

$$f(x+\delta x,\ y+\delta y,\ t+\delta t) = f(x,y,t) + \frac{\partial f}{\partial x}\delta x + \frac{\partial f}{\partial y}\delta y + \frac{\partial f}{\partial t}\delta t. \tag{3.11}$$

The image vector to be determined is the velocity $[\delta x\ \delta y]^{\mathrm{t}}$ associated to each pixel. From the previous equations it follows that

$$-\frac{\partial f}{\partial t}\delta t = \frac{\partial f}{\partial x}\delta x + \frac{\partial f}{\partial y}\delta y \ = \ \begin{bmatrix}\frac{\partial f}{\partial x} & \frac{\partial f}{\partial y}\end{bmatrix}^{\mathrm{t}} \begin{bmatrix}\delta x \\ \delta y\end{bmatrix} = \nabla f \begin{bmatrix}\delta x \\ \delta y\end{bmatrix} \tag{3.12}$$

Equation (3.12) does not give a unique solution for the flow vector $[\delta x\ \delta y]^{\mathrm{t}}$, but imposes a linear constraint on it. In fact, a problematic situation described as the *aperture problem* (see Fig. 3.11) may produce multiple possibilities for the vector flow, due to similar pixel intensities.

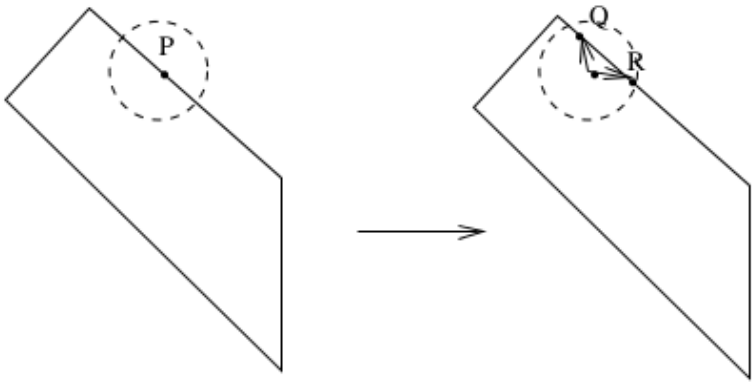


Figure 3.11 – The aperture problem. An intensity edge moves towards the right from time t_1 to time t_2. However, due to the limited size of the neighborhood (i.e., the *aperture* used for matching), the location of the displaced point P could be R or Q, or some other point along the edge segment determined by them (Gonzalez & Woods [74]).

According to Bradski [23], an estimate of the instantaneous velocity can be associated to each pixel of the image, representing the distance the pixel has moved between two successive frames. Such a construction is usually referred to as *dense optical flow*. Dense algorithms usually consider interpolation between points that are more easily distinguishable, so as to solve for points that are more ambiguous. Thus, these algorithms usually present higher computational cost. Some approaches available are: the *Horn-Schunck* method (Horn & Schunck [85]) that computes the velocity field, the *block matching* techniques (e.g. Beauchemin & Barron [16]) where the images are divided into small regions called blocks and motion is computed within each block, and the polynomial expansions approach by Farnebäck [68] which is explored in detail in Sec. 3.4.4.

As alternatives to dense estimations, *sparse optical flow* algorithms have been proposed with much less computational cost. These methods rely on some means of specifying beforehand the subset of points that are to be tracked. The selection may be obtained automatically according to some desirable properties (e.g. corners and edges, see Harris & Stephens [78]). The *Lucas-Kanade* algorithm (Lukas & Kanade [113]) was originally designed to calculate dense flow, but is widely used as a sparse technique since it only relies on local information. A survey on existing bottom-up tracking methods is presented in Ngau et al. [133] (see Tab. 3.2).

Characteristics	Biologically Inspired Models			Purely Computational Models		
	Pixel-based	**Frequency-based**	**Region-based**	**Pixel-based**	**Frequency-based**	**Region-based**
Algorithm complexity	High	Very-high	High	Average	High	Average
Computational speed	Average	Low	Average	High	Average	Average
Memory requirements	Average	Very-high	High	Low	Low	High
Detection performance	High	High	Average	High	High	Average

Table 3.2 – Survey on the performance of bottom-up tracking method according to the feature choice (Ngau et al. [133]).

Case studies

Given the task of interest for this work, which is approaching and positioning with respect to objects of interest on the scene guided by vision, several techniques for whole segmentation and feature tracking were explored. The idea was to verify whether reliable top-down and bottom-up information could be obtained from an inflow of digital images captured on-board. Since the autonomy is a desirable aspect of the solution, only methods relying on soft modeling were considered (the principle of Verification Vision was not included). In the testing conditions random and brusque motion are applied to the camera (producing motion blurs), simulating the robot walk motions. Furthermore, the algorithms are tested under illumination noise (i.e. under artificial and natural light sources). Thereby, the more promising results obtained are discussed in three case studies. In the first study a pixel-based semi-automatic approach relying on the clustering technique k-means is detailed. In the second study a top-down region-based segmentation technique considering a color model of the object under a *Markov random field* framework, is improved for a use in the context of continuous processing of visual inflow. In the third case study a feature tracking technique based on dense optical flow from polynomial expansions is explored.

Materials and resources

A RGB color web camera Logitech model C210, 640×480 (1.3 megapixels) resolution was used. Some images were also downloaded from the Internet for testing. The programs were implemented in the C++ programming language. The OpenCV 2.4.8 library was linked to the project, providing the implementation of *calcOpticalFlowFarneback* and *kmeans*. The algorithms were developed under the Eclipse Juno IDE and run in Ubuntu 12.04.5 LTS (Precise Pangolin). The host plattform was a DELL Vostro 1500 laptop (Intel Core 2 Duo 1.8GHz 800Mhz, 4.0GB DDR2 667MHz RAM, 256MB NVIDIA GeForce 8600M GT).

CS-I: Semi-automatic color-based segmentation

The *k-means* algorithm by MacQueen [115] is a convenient technique that can be used for unsupervised learning. The distinctive aspects of clustering is that it avoids the need for pre-structuring data, so structure is automatically found. According to Ertel [63], k-means can be considered as a deterministic or discrete version of the *expectation maximization* (EM) algorithm, where clusters are represented by points and not by probability

distributions. K-means has been employed for image segmentation (e.g. in Doggaz & Ferjani [56]), such that properties related to the image pixels can be grouped into clusters. In this study case, the problem of segmentation of a sequence of images is tackled.

As its name suggests, k clusters are defined by their average value. The procedure is illustrated in Algorithm 1. Firstly, the k cluster midpoints $C = \{\mu_1, \ ... \ , \mu_k\}$ are randomly or manually initialized. Then, the *classify* and the *recalculate* procedures are systematically applied until convergence. In the former points are assigned to a cluster based on distinct metrics depending on the particular application. Some available criteria are euclidean distance, sum of squared, manhattan distance, among others (see Duda et al. [58]). In the *recalculate* step, the cluster midpoint μ for points $S = \{P_1, \ ..., P_n\}$ are determined such that

$$\mu = \frac{1}{n} \sum_{i=1}^{n} S_i. \tag{3.13}$$

Algorithm 1 K-means

```
1: procedure K-MEANS(S, k)
2:     initialize μ_1, ... , μ_k                    ▷ e.g. randomly
3:     repeat
4:         Š ← classify P ∈ S to each's nearest μ_i ∈ C
5:         recalculate C
6:     until no change in C
7:     ← (Š, C)
```

The outputs of k-means are the midpoints set C and the clustered set $\check{S}$. It is important to mention that convergence is not ensured in the algorithm. Though a partial solution is obtained by restricting to a maximal number of iterations. Normally, the number of iterations is typically much smaller than the cardinality of the data point set. The complexity order of the algorithm is $O(ndkt)$, where n is the total number of points, d is the dimensionality of the feature space, and t the number of iteration steps.

Experiment

The images were captured in the RGB color space and convolved with a 3×3 low-pass Gaussian kernel to reduce noise. Three experiments were designed. In the first experiment the color channels of the image are de-multiplexed (which is equivalent of obtaining 3 gray-scale image matrices $\mathbf{R}$, $\mathbf{G}$, and $\mathbf{B}$; representing respectively the intensities of the red, green, and blue components), so the feature vector $\mathbf{P}_i$ related to the image location $i = (x, y)$ is $\mathbf{P}_i = [\mathbf{R}_{xy} \ \mathbf{G}_{xy} \ \mathbf{B}_{xy}]^{\mathrm{t}}$. In the second experiment the coordinates of the pixels are also included in the feature vector, in order to enforce clusters to have also a topological coherence, such that $P_i = [x \ y \ \mathbf{R}_{xy} \ \mathbf{G}_{xy} \ \mathbf{B}_{xy}]^{\mathrm{t}}$. The dataset was normalized in each dimension to avoid the effect of the scaling factor. For these experiments the clusters are initialized randomly, k is varied between 2 and 10, and a threshold for the iteration number $t \leq 10$ is set. In a last experiment an heuristic criteria for initializing the clusters is examined, such that, based on the assumption of scene constancy, the cluster midpoints obtained from the processing of a frame are provided as initial values for the successive frame.

Results

A comparison of the results for the first two experiments is given in Fig. 3.12. The differences are more distinguishable when the number of clusters is reduced, so the coordinate of the pixel would enforce a topological clustering. For larger values of k the results are more or less equivalent. In general, for the scene shown a value of $4 \leq k \leq 6$ provided the best results. That is, for low values of k distinct objects tended to be merged (a problem known as *under segmentation*), contrarily, with a high number of clusters objects scattered on adjacent regions (i.e. the *over segmentation* problem). As illustrated in the top row of Fig. 3.13, the structure of the black filing cabinet at the back could be consistently recognized, though the red ball and the blue-frame calendar were not. When the clusters were initialized randomly the segmentation varied considerably between successive frames. Thus, the heuristics used in the third experiment produced more consistent clusters among frames.

Discussion

The objective of this study was to explore a segmentation technique from pixel-based clustering, in order to verify whether a physically plausible segmentation can be obtained from the scene without possessing previous knowledge about the characteristics (e.g. the color and shape) of the objects contained. As the results have shown, although some structure can be recovered, the stability of the segmentation for small objects cannot be ensured (the segmentation oscillates given the illumination noise). Another disadvantage of the method is that it is sensitive to the parameter k, which implies that some knowledge about the scene may be available. In case the last cluster midpoints were given for initialization, better results were obtained, though when the first clusters did not produce a correct segmentation, errors were propagated to successive frames. Moreover, the appearance and disappearance of objects in the scene destabilized the clusters, so the heuristics criteria would no longer hold (this is problematic since even though the scene is static, objects can enter and leave the field of vision of the robot when it walks).

The addition of topological information in the feature vector did not produce the expected results, since the number of clusters is much smaller than the number of locations in the image, which is insufficient to capture the local context around neighbor pixels. Increasing the number of clusters would not produce better results since the image would be over segmented. Ming et al. [122] have proposed to impose spatial constraints to the clustering, so considering contour detection for merging regions in order to reduce over segmentation. The algorithm gives good results for individual images, but it is not obvious how to obtain adequate performance for on-line applications over a video sequence, since it is based on the estimation of the adjacency between regions. There are $n = r!/(2(r-2)!)$ possible ways of selecting a pair among r regions. For a natural scene, depending on k, hundreds or regions may be segmented which would be intractable. From the results obtained, it can be concluded that the clustering algorithm detailed is not adequate for processing visual saliency for the image sequence generated by the robot locomotion.

CS-II: Top-down color-based segmentation

In top-down segmentation the objective is to distinguish the regions of interest on the image, based on a supervised color model of the object(s) of interest. Kato et al.

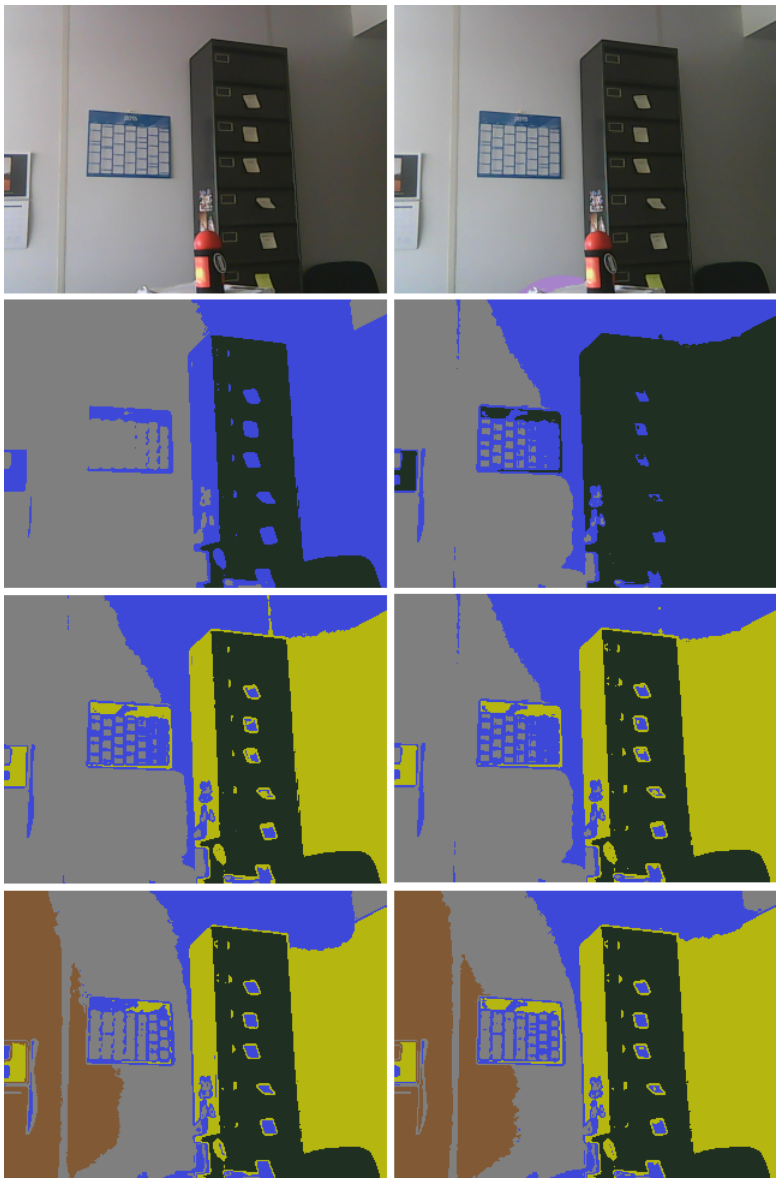


Figure 3.12 – K-means segmentation from color and topology. The first row shows two captured frames of the scene. From the second to the bottom row the results for $k \in \{3, 4, 5\}$. The left column presents the clustering obtained from the first experiment (i.e. the feature vector $\mathbf{P}_i = [\mathbf{R}_{xy}\ \mathbf{G}_{xy}\ \mathbf{B}_{xy}]^{\mathrm{t}}$). The right column presents the clusters generated for the second experiment (i.e. the feature vector $\mathbf{P}_i = [x\ y\ \mathbf{R}_{xy}\ \mathbf{G}_{xy}\ \mathbf{B}_{xy}]^{\mathrm{t}}$).

[95] proposed an image segmentation technique within a *Markov Random Field* (MRF) framework. The approach combines information from individual pixels and a surrounding neighborhood (the spacial coherence), providing a more robust solution under noisy conditions. Let the observed image $F = \{\mathbf{f}_i \mid i \in I\}$ consisting of spectral components values expressed in a certain color-space η, be represented by the vector $\mathbf{f}_i$ at each location i. The label of interest $\hat{\varphi}$ is the one that maximizes the a posteriori probability $p(\varphi \mid F)$, such that

$$\mathrm{argmax}_{\varphi \in \Phi} \prod_{i \in I} p(\mathbf{f}_i \mid \varphi_s) p(\varphi), \tag{3.14}$$

where Φ denotes the set of all possible labellings. Since the goal is to partition the image

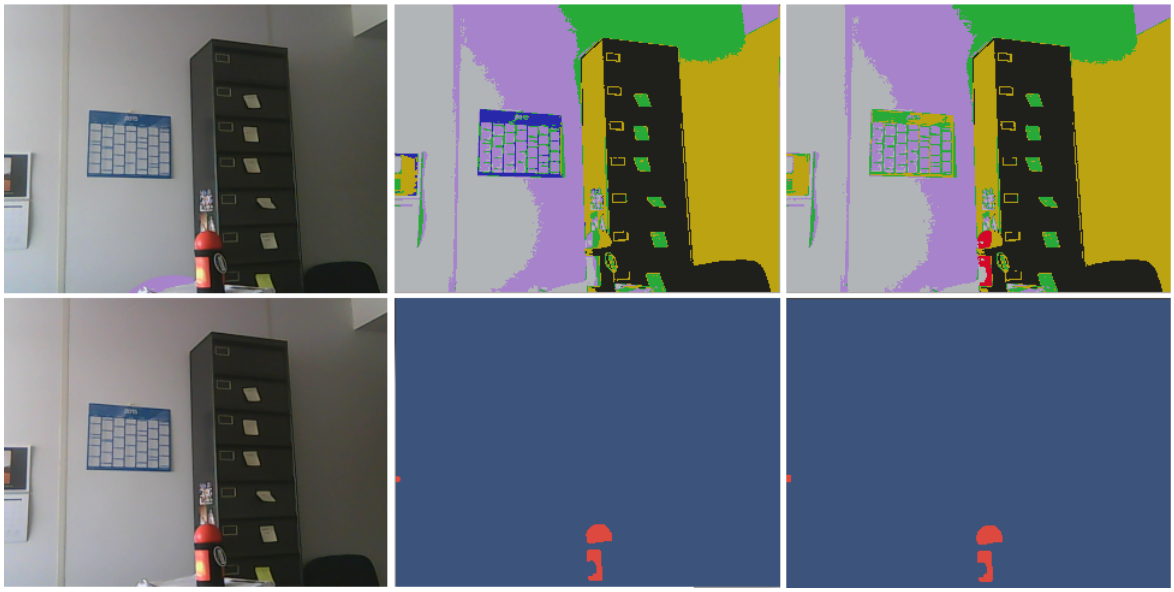


Figure 3.13 – Comparison between k-means and the top-down region-based segmentation method developed in Sec. 3.4.3. In the top row, the results for k-means with 6 clusters. At the bottom row the results for the region-based segmentation algorithm. On the left column the RGB image. At the center and right two successive segmentations within an interval of 50 milliseconds. K-means cannot ensure a consistent detection of the red ball.

into labeled regions, a pixel class λ may represent more than one homogeneous color patches in the input image. Such regularities are modeled by considering additive white noise with covariance Σ_λ, centered around the expected color value μ_λ. Thus, $p(\mathbf{f}_i \mid \varphi_i)$ follows a Gaussian distribution and pixel classes $\lambda \in \Lambda = \{1, 2, \ldots, L\}$ are represented by the mean vectors μ_λ and the covariance matrices $\mathbf{\Sigma}_\lambda$. Furthermore, $p(\varphi)$ corresponds to a MRF with respect to a first order neighborhood system (as shown in Fig. 3.14).

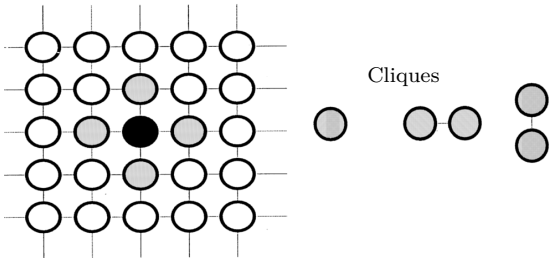


Figure 3.14 – First-order neighborhood system. Single pixel cliques are called singletons, horizontal and vertical cliques are called doubletons (Kato et al. [95]).

According to the *Hammersley-Clifford theorem*, $p(\varphi)$ follows a Gibbs distribution, such that

$$p(\varphi) = \frac{e^{-u(\varphi)}}{m(\gamma)} = \frac{\prod_{c \in C} e^{-v_c(\varphi_c)}}{m(\gamma)}, \tag{3.15}$$

where $u(\varphi)$ is called the *energy function* and $m(\gamma) = \sum_{\varphi \in \Phi} e^{-u(\varphi)}$ is the normalizing constant (or partition function). v_c denotes the *clique potential* of clique $c \in C$ having the label configuration φ_c. The energies from pixel values (i.e. the *singleton* contribution) directly reflect the probabilistic modeling of labels without context, while clique potentials (i.e.

the *doubleton* contribution) express the relationship between neighboring pixel labels. The energy function has the form

$$u(\varphi, F) = \sum_{i \in I} \ln\left(\left(\sqrt{(2\pi)^3 |\mathbf{\Sigma}_{\varphi_i}|}\right) + \frac{1}{2}(\mathbf{f}_i - \mu_{\varphi_i})\mathbf{\Sigma}_{\varphi_i}^{-1}(\mathbf{f}_i - \mu_{\varphi_i})^{\mathrm{t}}\right) + \beta \sum_{\{i,r\} \in C} \delta(\varphi_i, \varphi_r), \tag{3.16}$$

where $\delta(\varphi_i, \varphi_r)$ is the Kronecker delta function. At the right side of the equation, the left term corresponds to the singleton contribution and the right term to the doubleton contribution. The parameter $\beta > 0$ controls the homogeneity of the regions. As β increases the regions become more homogeneous. The function $u(\varphi, F)$ is non-convex, so the convergence to the global optimum cannot be ensured, since the calculation of $m(\gamma)$ in (3.15) is intractable. In practice, combinatorial optimization techniques (e.g. the *iterated conditional modes* (ICM) by Besag [19]) are employed to achieve the segmentation. The next state $\hat{\varphi}_i^{k+1}$ is determined by

$$\hat{\varphi}_i^{k+1} \leftarrow \mathrm{argmin}_{\varphi_{i \in \{1,\ldots,L\}}} u(\hat{\varphi}^k, F). \tag{3.17}$$

The stop condition is attained when

$$\hat{\varphi}_i^{k+1} = \hat{\varphi}_i^k, \forall i \in I. \tag{3.18}$$

To summarize, the parameters of the system are $(\mu_\lambda, \mathbf{\Sigma}_\lambda, \beta)$. In case when they are provided by the user, a supervised segmentation is obtained. Otherwise, they must be estimated simultaneously to φ (e.g. the unsupervised algorithm by Deng & Clausi [55]).

The segmentation algorithm

The application of interest for the algorithm is to distinguish a particular object (the foreground) from other elements on the scene (the background), that is, to obtain a binary mask of the scene (i.e. $|\Lambda| = 2$) at each frame. For this, the sample implementation of the technique [1] for the segmentation of single images was improved for an efficient use in continuous video inflow. The resulting routine considered the ICM optimization (see Algorithm 2). The user specifies the color model of the object by enclosing a region on the image. In order to assist this procedure, the GrabCut technique by Rother et al. [156] is employed. The images are converted from the RGB to the YUV color-space (see Stockman & Shapiro[172] for an in depth review on image color spaces), since with YUV typically compression artifacts are more efficiently masked, both are 3D color spaces. Consequently, a probabilistic distribution of color intensity under Gaussian noise is obtained (i.e. the parameters μ_λ and $\mathbf{\Sigma}_\lambda$ of the model), and given to the segmentation algorithm. The parameter β is set to 1. The resulting computational complexity is $O(tn^{|\Lambda|})$, where n is the number of pixels in the image, and t is the maximal number of iterations allowed (in case it is specified). The *localEnergy* function corresponds to Eq. (3.16). The energy cost for each pixel is evaluated only with the model of the object, that is, no assumptions are made about the colors at the background. This is crucial since the same model is used to process successive frames, so in case a model would be taken for the background, unseen colors eventually entering the scene would introduce ambiguity in the labeling.

1. Available at http://www.inf.u-szeged.hu/~kato/software/mrfdemo.html

The *initialize* step sets the initial segmentation $\hat{\varphi}$ by minimizing the singleton term (i.e. a labeling without context, see Eq. (3.16)).

Algorithm 2 Segmentation

```
1: procedure doSegmentation
2:     φ̂ ← initialize                                  ▷ Singleton initialization
3:     e_Old ← 0
4:     repeat
5:         e ← 0
6:         for y = 0 → y < height do
7:             min_e ← localEnergy(x, y, φ̂)
8:             for x = 0 → x < width do
9:                 for λ = 0 → λ < |Λ| do
10:                    c_e ← localEnergy(x, y, λ)          ▷ current energy
11:                    if c_e < min_e then
12:                        φ̂_{y,x} ← λ
13:                        min_e ← c_e
14:            e ← e + min_e
15:        Δe ← abs(e_Old − e)
16:        e_Old ← e                                   ▷ stop when the change is too small
17:    until Δe > ϵ
18:    ← φ̂
```

Experiment

Two applications of the algorithm are evaluated: one with single images, and the other with on-line processing of captured sequences from a moving camera. For reducing noise in the second condition, the 10 initial acquisitions with the static camera are averaged before building the color model.

Results

As shown in Figs. 3.15 and 3.16, the technique provides robust segmentation for natural scenes and colored objects. This was also the case for the condition of camera motions. As illustrated in Fig. 3.17, despite motion blurs (see Sec. 3.3.1) were produced, so the morphology of the salient blob was slightly elongated and deformed; the object could be fully segmented. In relation to the number of iterations for convergence, it was observed that most of the final segmentation is accomplished in $t \leq 5$ iterations. In general, good results are obtained for diffuse, non-reflective textures. Less satisfactory segmentations were obtained for metallic textures that reflected specular illumination. It was also observed that when artificial light was present more samplings were required to build the color model (since conventional light bulbs add oscillatory noise to the scene).

Discussion

The evaluation of the segmentation algorithm has shown that it is a plausible approach for unstructured scenes. The information provided by the local neighborhood allows a more robust handling of illumination noise, which is not possible from the pixel-based approach (see Fig. 3.13). However, it is important to mention that the segmentation does not provide good results for certain materials (e.g. polished and reflective surfaces),

Figure 3.15 – Segmentation of a natural scene. On the left the original image. On the right the segmentation of the backboard.

Figure 3.16 – Segmentation of colored objects. On the left the original image. On the right the segmentation of the zebras.

or when artificial illumination is excessive (e.g. the incidence of low-frequency lights on the scene, specially during the night). Besides, as a top-down technique, it presents the disadvantage of requiring an explicit model of the color of the object, which was provided by demonstration. This technique is used in the study cases of Chapters 4 - 6 as a means to obtain top-down saliency processing, for approaching a known object in the scene from images captured on-board.

CS-III: Bottom-up segmentation based on optical flow

The interest of bottom-up segmentation in this research is to recognize the spatial structure of the scene, without possessing a model of the objects or the arrangement between them in the environment. This can be done through the processing of the dense optic flow. In the method proposed by Farnebäck [68], the central idea is to predict the signal at a pixel location $\mathbf{x}$ based on a polynomial approximation of its local neighborhood (i.e. a *polynomial expansion*), and to look for a similar pattern in the next image. For this, a quadratic polynomial is used to capture information about the signal. The DC level, the odd, and the even part of the signal, are respectively modeled by the constant, the linear, and the quadratic term. Thus, the signal can be expressed in the coordinate system

$$f(\mathbf{x}) \sim \mathbf{x}^{\mathrm{t}}\mathbf{A}\mathbf{x} + \mathbf{b}^{\mathrm{t}}\mathbf{x} + c, \tag{3.19}$$

Figure 3.17 – Segmentation under camera motions. On the left column the still image of the scene, on the right column a strong lateral motion was applied to the camera. The row at the bottom shows the segmentation obtained for red regions.

from the quadratic basis $\{1,\ x,\ y,\ x^2,\ y^2,\ xy\}$, such that

$$\mathbf{A} = \begin{bmatrix} x^2 & \frac{xy}{2} \\ \frac{xy}{2} & y^2 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} x \\ y \end{bmatrix}, \quad c = 1. \tag{3.20}$$

The model can be generalized so the pixels of the image are associated to a $n \times 1$ polynomial parameter vector $\mathbf{r}$ that captures the structure of the signal, with n the size of the neighborhood. These parameters can be obtained by $\mathbf{r} = (\mathbf{B}\mathbf{W}_\mathrm{a}\mathbf{W}_\mathrm{c}\mathbf{B})^{-1}\mathbf{B}\mathbf{W}_\mathrm{a}\mathbf{W}_\mathrm{c}\mathbf{f}$, where the $n \times n$ matrices $\mathbf{W}_\mathrm{a} = \mathrm{diag}(\mathbf{a})$ and $\mathbf{W}_\mathrm{c} = \mathrm{diag}(\mathbf{c})$. The non negative $n \times 1$ applicability vector $\mathbf{a}$ indicates the significance or importance of each point in the neighborhood (i.e. the locations m on a 2D region centered on the pixel are represented by a column vector). The non negative $n \times 1$ certainty vector $\mathbf{c}$ is a measure of the confidence in the signal values at each point. Possible causes for uncertainty are: defective sensor elements, varying confidence in the results from previous processing, and locations outside the image bounds, among others. The measured signal in the neighborhood is denoted by $\mathbf{f}$. Each basis function is an element of a finite dimensional vector space Q^n represented by a $n \times 1$ column vectors $\mathbf{b}_i$. The set $\{\mathbf{b}_i\}_1^m$ of the basis functions, are stored in the $n \times m$ matrix $\mathbf{B} = [\mathbf{b}_1\ \mathbf{b}_2\ \cdots\ \mathbf{b}_m]$ (an example of a basis function of size $n = 9$ is given in Eq. (3.39)).

According to Farnebäck [67], a polynomial expansion is performed in both images which provides the coefficients of Eq. (3.19) $\mathbf{A}_1$, $\mathbf{b}_1$, and c_1 for the first image; and $\mathbf{A}_2$, $\mathbf{b}_2$, and c_2 for the second image. From the analysis of polynomial changes under an ideal global translation between the two images (in practice more sophisticated motion model of pixels are used, e.g. in Eq. (3.30)), the following identities hold

$$\mathbf{A}_2 = \mathbf{A}_1, \tag{3.21}$$

$$\mathbf{b}_2 = \mathbf{b}_1 - 2\mathbf{A}_1\mathbf{d}, \tag{3.22}$$

$$c_2 = \mathbf{d}^{\mathrm{t}}\mathbf{A}_1\mathbf{d} - \mathbf{b}_1^{\mathrm{t}}\mathbf{d} + c_1, \tag{3.23}$$

where the first and the second frames are denoted by 1 and 2 respectively, and $\mathbf{d}$ is the translation between the polynomial locations. A solution can be obtained from Eq. (3.22) if $\mathbf{A}_1$ is non-singular, such that

$$\mathbf{d} = -\frac{1}{2}\mathbf{A}_1^{-1}(\mathbf{b}_2 - \mathbf{b}_1). \tag{3.24}$$

However, since noise affects the measurements, Eq. 3.19 is approximated with local polynomials at each pixel neighborhood. Thus, a new notation is introduced to denote the fact that the global displacement has been replaced by the spatially varying displacement field $\mathbf{d}(\mathbf{x})$. In practice

$$\mathbf{A}(\mathbf{x}) = \frac{\mathbf{A}_1(\mathbf{x}) + \mathbf{A}_2(\mathbf{x})}{2} \tag{3.25}$$

and

$$\Delta\mathbf{b}(\mathbf{x}) = -\frac{1}{2}(\mathbf{b}_2(\mathbf{x}) - \mathbf{b}_1(\mathbf{x})) \tag{3.26}$$

are used. Thus, the primary constraint of Eq. (3.24) would become

$$\mathbf{A}(\mathbf{x})\mathbf{d}(\mathbf{x}) = \Delta\mathbf{b}(\mathbf{x}), \tag{3.27}$$

Despite Eq. (3.27) can be solved point-wise, the results would be probably affected by noise. Thereby, by assuming a slow variation of the displacement field, information can be integrated over a neighborhood I around each pixel. Thus, a solution is obtained by minimizing the term

$$\sum_{\Delta\mathbf{x}\in I} w(\Delta\mathbf{x})\|\mathbf{A}(\mathbf{x}+\Delta\mathbf{x})\mathbf{d}(\mathbf{x}) - \Delta\mathbf{b}(\mathbf{x}+\Delta\mathbf{x})\|^2, \tag{3.28}$$

where $w(\Delta\mathbf{x})$ is a weight function affecting the contribution of the points in the neighborhood. The minimum is obtained from

$$\mathbf{d}(\mathbf{x}) = \left(\sum w\mathbf{A}^{\mathrm{t}}\mathbf{A}\right)^{-1}\sum w\mathbf{A}^{\mathrm{t}}\Delta\mathbf{b}. \tag{3.29}$$

The notation has been simplified to make the expression more readable. According to Farnebäck [68], a solution exist and is unique unless the whole neighborhood is exposed to the aperture problem (see Fig. 3.11).

The displacement field can be parameterized according to a more sophisticated pixel motion model, so both the affine transform (i.e. translation and rotation under orthographic projection), and the perspective projection, are taken into account. The model is defined by eight parameters as follows

$$\begin{aligned} t_{\mathrm{x}}(x,y) &= a_1 + a_2 x + a_3 y + a_7 x^2 + a_8 xy, \\ t_{\mathrm{y}}(x,y) &= a_4 + a_5 x + a_6 y + a_7 xy + a_8 y^2. \end{aligned} \tag{3.30}$$

Thus, the displacement from Eq. (3.29) is defined such that

$$\mathbf{d}(\mathbf{x}) = \mathbf{S}\mathbf{p} \tag{3.31}$$

$$\mathbf{S} = \begin{bmatrix} 1 & x & y & 0 & 0 & 0 & x^2 & xy \\ 0 & 0 & 0 & 1 & x & y & xy & y^2 \end{bmatrix} \tag{3.32}$$

$$\mathbf{p} = [a_1 \; a_2 \; a_3 \; a_4 \; a_5 \; a_6 \; a_7 \; a_8]^{\mathrm{t}}, \tag{3.33}$$

The weighted least squares problem in Eq. (3.28) is reformulated to consider the motion model, so

$$\sum_i w_i \|\mathbf{A}_i \mathbf{S}_i \mathbf{p} - \Delta\mathbf{b}_i\|^2, \tag{3.34}$$

with i indexing the coordinates in a neighborhood. The solution for the motion model parameters is

$$\mathbf{p} = \left(\sum_i w_i \mathbf{S}_i^{\mathrm{t}} \mathbf{A}_i^{\mathrm{t}} \mathbf{A}_i \mathbf{S}_i \right)^{-1} \sum_i w_i \mathbf{S}_i^{\mathrm{t}} \mathbf{A}_i^{\mathrm{t}} \Delta\mathbf{b}_i. \tag{3.35}$$

A priori knowledge about the displacement field can be heuristically used to compare the polynomial at $\mathbf{x}$ in the first signal to the polynomial at $\mathbf{x} + \tilde{\mathbf{d}}(\mathbf{x})$ in the second signal, where $\tilde{\mathbf{d}}(\mathbf{x})$ is the initial displacement field rounded to integer values (since in the image measurements are taken in discrete pixels). The relative displacement between the real value and the rounded a priori estimate can be obtained by replacing Eqs. (3.25) and (3.26) by

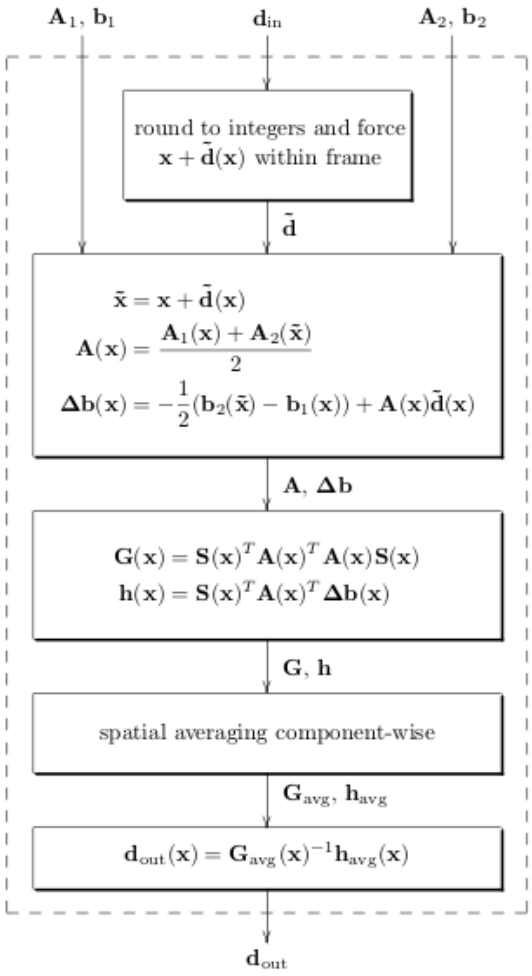
$$\mathbf{A}(\mathbf{x}) = \frac{\mathbf{A}_1(\mathbf{x}) + \mathbf{A}_2(\tilde{\mathbf{x}})}{2}. \tag{3.36}$$

$$\Delta\mathbf{b}(\mathbf{x}) = -\frac{1}{2}(\mathbf{b}_2(\tilde{\mathbf{x}}) - \mathbf{b}_1(\mathbf{x})) + \mathbf{A}(\mathbf{x})\tilde{\mathbf{d}}(\mathbf{x}), \tag{3.37}$$

where

$$\tilde{\mathbf{x}} = \mathbf{x} + \tilde{\mathbf{d}}(\mathbf{x}). \tag{3.38}$$

The structure of the iterative solution to the flow estimation is presented in Fig. 3.18. Two different approaches are considered: iterative and multi-scale estimation. In the former, the output of one iteration is used as a priori displacement for the next step. The system can be initialized with a priori zero displacement, unless actual knowledge about the displacement field is available. The multi-scale approach is suited for handling the cases of too large displacements between successive frames. The idea is to start by a coarse scale to get a rough displacement estimate, and to propagate this through finer scales to obtain increasingly more accurate estimates. Compared to the iterative



Eq. (3.38)

Eqs. (3.36), (3.37), (3.38)

$\mathbf{G}(\mathbf{x})$ and $\mathbf{h}(\mathbf{x})$ are components of Eq. (3.35)

The motion model parameters $\mathbf{p}$ of Eq. (3.33) are estimated

Figure 3.18 – Algorithm for displacement estimation (Farnebäck [68]).

estimation, this approach requires new polynomial expansion coefficients to be computed for each scale.

The computational complexity of the displacement estimation is dominated by two steps: the polynomial expansion and the spatial averaging step. The complexity of polynomial expansion depends on a number of factors, including the dimensionality u of the signal space, the size n of the applicability per dimension, whether the certainty is assumed to be constant, and whether the applicability is separable and sufficiently symmetric. For a 2D image, assuming constant certainty in symmetric kernels, a complexity $O(2u^2)$ is obtained. The averaging operation can be assumed to be implemented by separable filtering. Let s and z be respectively the length and dimensionality of such filters, and j the components of the motion model (for 2D images there are in total 39 components for the eight-parameter motion model). For k iterations of the algorithm the computational complexity per pixel is $O(\frac{zsj^2k}{2})$.

Experiments

Two experiments are designed. In the first one, an single image is processed according to the polynomial expansion technique, in order to verify whether the predicted signal preserves the structure of the scene. The image is firstly converted to gray-scale and convolved with a 3×3 Gaussian low-pass filter for reducing noise. The quadratic basis

functions is employed for the polynomial expansion (see Eq. (3.19)), according to the basis set $\{1,\ x,\ y, x^2,\ y^2,\ xy\}$. For this, a 3×3 neighborhood ($n = 9$) is considered. In the certainty matrix $\mathbf{W}_\mathrm{c}$, a value of 1 was given to all pixels populated with valid data, and 0 otherwise. A Gaussian kernel is used as the applicability criteria $\mathbf{W}_\mathrm{a}$. The resulting matrices are obtained by variating the coordinates x and y (e.g. the azimuth and elevation with respect to the sensor retina) relative to the central pixel, such that

$$\mathbf{g} = \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}, \qquad \mathbf{a} = \begin{bmatrix} 1 \\ 2 \\ 1 \\ 2 \\ 4 \\ 2 \\ 1 \\ 2 \\ 1 \end{bmatrix}, \qquad \mathbf{B} = \begin{bmatrix} 1 & -1 & -1 & 1 & 1 & 1 \\ 1 & -1 & 0 & 1 & 0 & 0 \\ 1 & -1 & 1 & 1 & -1 & 1 \\ 1 & 0 & -1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & -1 & 1 \\ 1 & 1 & -1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & -1 & 1 \end{bmatrix}, \tag{3.39}$$

where $\mathbf{g}$ is the Gaussian kernel, and $\mathbf{a}$ is the column vector of $\mathbf{g}$. The local coordinate system for obtaining $\mathbf{B}$ is defined from a second order neighborhood (i.e. a grid topology containing the central element surrounded by 8 neighbors).

The second experiment considered the segmentation of the scene based on the optical flow induced by camera motions. The magnitude of the flow in the image is partitioned into 5 sets, to verify whether the segmentation is physically plausible. That is, if the magnitude of the flow would provide information about the depth of the parts of the object to the camera sensor. Let the optic flow vector associated to each pixel $i = (x, y)$ of the image I be denoted by $\mathbf{o}_i = [\delta x\ \delta y]^\mathrm{t}$ (see Eq. (3.12)). The flow magnitude image $\bar{f}$ is defined such that

$$\bar{f} = \|\mathbf{o}_i\|,\ \forall i \in I. \tag{3.40}$$

The segmentation is obtained by a threshold test for each cluster conforming to Eq. (3.9).

Results

The results for the first experiment are presented in Fig. 3.19. As seen, the structure of the scene was approximately reconstructed by the polynomial expansion, though some contours were lost (e.g. the top border of the filing cabinet). Figure 3.20 gives the results for the second experiment. As the images show, it is possible to segment the objects from the background and detecting the morphology, without possessing any prior information about the objects (e.g., the color, the geometrical properties, etc.).

Discussion

The first experiment (i.e. the polynomial expansion of the image) showed that information related to the structure of the scene can be captured with relative precision. As expected, the best results are obtained for contrasting textured regions, where more distinguishable features can be detected. This is the case of edges (e.g. the calendar fixed at the wall, see Fig. 3.19). Regions of low variations on pixel intensities are flattered (e.g. the irregular illumination in the walls is practically undetected). Moreover, given

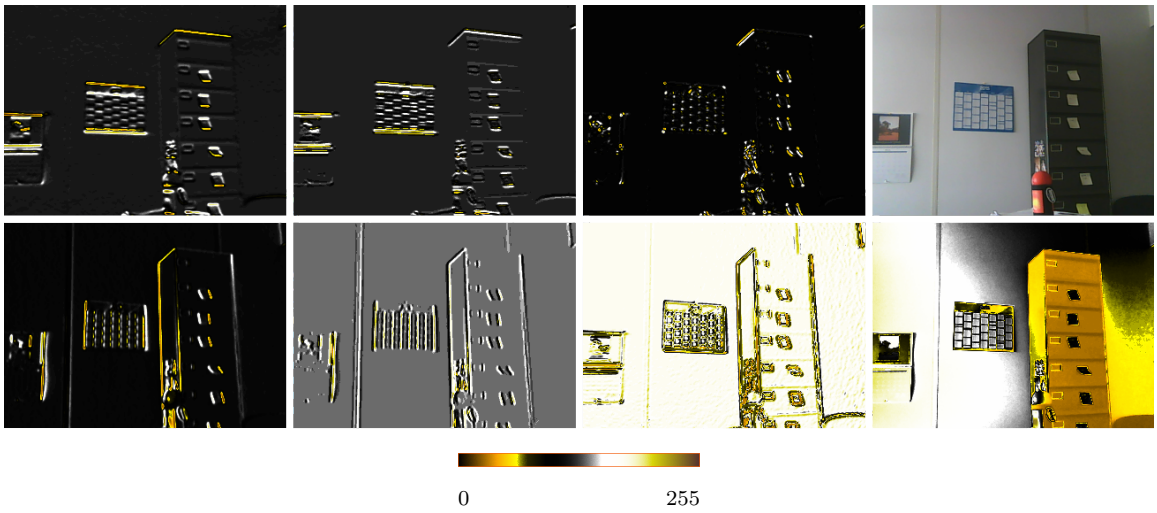


Figure 3.19 – Polynomial basis and signal reconstruction. In relation to the quadratic basis set $\{1,\ x,\ y, x^2,\ y^2,\ xy\}$, the top row, includes from left to right the images corresponding to x, x^2, xy, and the image I. Likewise, in the bottom row the images correspond to y, y^2, the reconstructed signal $f(\mathbf{x})$ (see Eq. (3.19)), and the original scene mapped to a similar color gradient scale (shown at the bottom) for comparison.

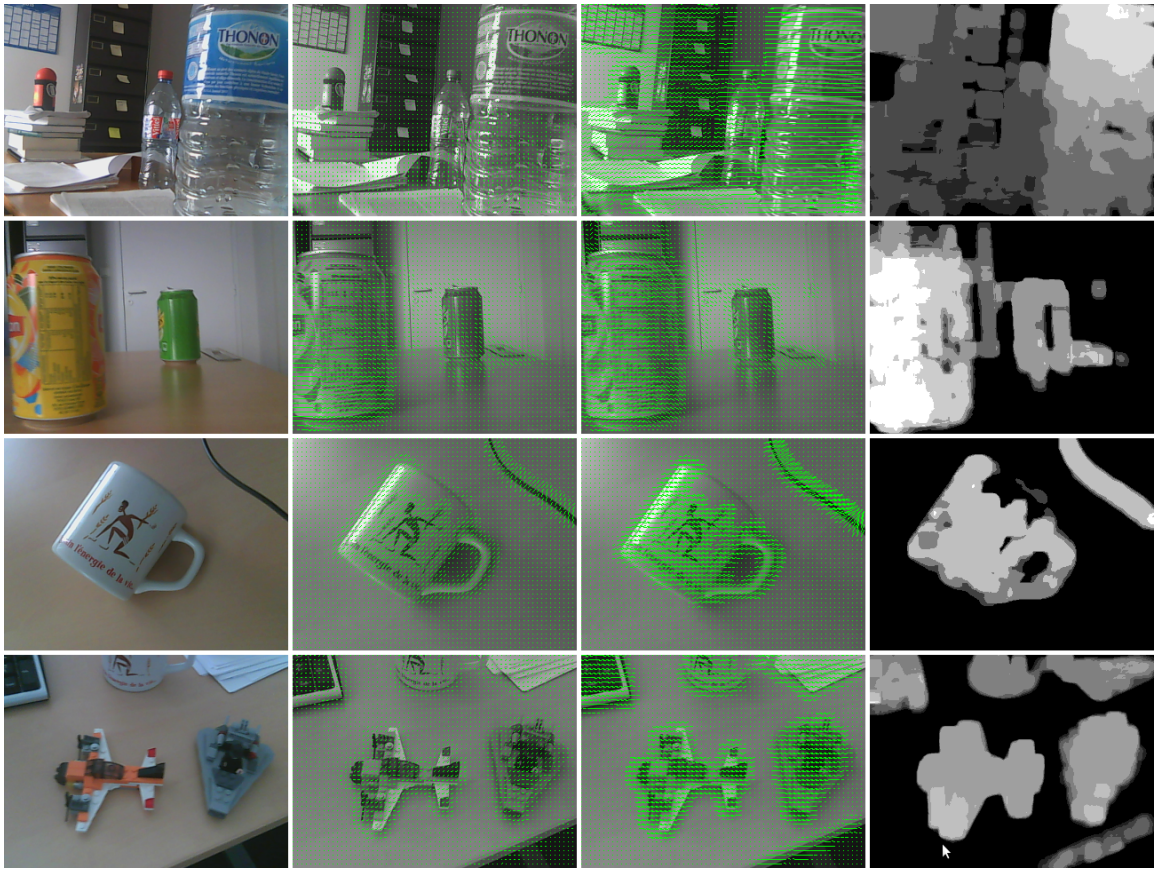


Figure 3.20 – Optical flow segmentation. The images illustrate the segmentation of the scene based on the magnitude of the optical flow. From left to right, columns correspond to: a captured frame, the dense flow estimated at to two consecutive frames, and the segmentation obtained. In the segmentations brighter regions presented bigger flow magnitude.

the aperture problem (illustrated in Fig. 3.11), it is possible that the contour between adjacent homogeneous regions is undetected.

The results for the second experiment suggested that it is possible to obtain an unsupervised estimation of the structure of objects from the optical flow. The magnitude of the flow can be related to the spatial depth relative to the sensor for a static object. Problems occur when the camera motions are too brusque. When motion blurs are produced the local contrast is reduced, so the prediction based on polynomial expansion is less precise. Furthermore, the greater the displacement between features, the less precise the estimation of motion would be, thus a relatively high frame-rate must be available for applications in humanoid locomotion. As it is discussed in the study cases of Chapter 6, this approach is used for inducing reactive motion in a static scene. The more general problem of segmentation under joint motion (from the agent and the objects) is investigated in Sekkati & Mitiche [163], where the *level sets* approach is proposed.

Conclusions

In the context of autonomous behavior, this chapter has focused on the study of visual attention. As discussed, the selection of information can be driven endogenously (by goals or top-down), or exogenously (by novelty or bottom-up). From the multidisciplinary perspective, important models from cognitive science research were reviewed. In this sense, the filter theory has pointed out the effect attention exerts on reducing the amount of information that enters the cognitive system. The spotlight metaphor has pointed out how space can constitute a powerful coordinate system for perceptual systems where attention may directly operate. The FIT and GS models have provided a more detailed description of mechanisms for information integration, and a more complex treatment for the stages of processing (i.e. the pre- and post-selection of information). As it is discussed in Chapters 4 - 6, these models have inspired the current study in several ways. In agreement with the filter theory data is pre-processed for obtaining more efficiency, so only relevant information gains access to more complex processing stages (i.e. early selection). In Sec. 5.3.3 the spotlight metaphor is employed to propose an embodied mechanism (i.e. the Embodied Filtering task) that is in charge of selecting the retinal data related to the object of interest, under top-down saliency ambiguity. Inspired by the models FIT and GS, the idea of combining multiple layers of image features is adopted. For this, in Chapter 6 top-down and bottom-up saliency features are used to control the robot walk, so it can reactively approaching an object or avoiding obstacles.

A review on the sensor technologies was also presented. Some problems related to CCD sensors (e.g. motion blurs) were discussed, and an overview of the structure of the human eye was included. Among the several differences with respect to conventional cameras are: the decoupling of information of illumination and color in dedicated photoreceptors in the eye vs. coupling in the camera photosites, the higher dynamic range of human vision, the non-uniform disposition of receptors in the retina vs. a grid-like arrangement of pixels, and the fact that curved geometry of the retina vs. the planar geometry of the camera retina may provide better resolution at the image borders. Perhaps the most importance difference is that human vision is actually a dynamic process that takes place in several phases, so it would be comparable to a video inflow and not to a photography. That is, the resulting mental image is a reconstruction of the scene based on different sort of inputs that the eyes actively gathers in different phases, and not the

mere registry of the actual light received by the sensor. Thereby, human vision is much less affected by noise.

The review on the literature of machine vision has revealed two main research branches for feature extraction. In the whole scene segmentation branch, available methods were classified into pixel-, region-, edge-, and model-based. In the feature tracking branch, the principle of verification vision was described and contrasted to model-less approaches, such that dense and sparse optical flow. Based on this review three study cases were conducted. The first study considered semi-automatic pixel-based segmentation, by employing the k-means clustering technique. The results suggested that although some structure is recovered, the segmentation may not be physically plausible for continuous imagery, so the approach is not suited for goals of this work. In the second study, a top-down region-based technique for image segmentation within a MRF was improved for operating in real-time in the case of continuous inflow. The evaluation showed that it is a plausible approach for unstructured scenes, though the performance is degraded under artificial illumination, and the detection of metallic or reflective objects. This technique is used in the study cases of Chapters 4 - 6 as a means to obtain top-down saliency processing, for approaching a known object in the scene from images captured on-board. Finally, the third study considered a tracking method for the estimation of dense optical flow based on polynomial expansion. The results suggested that, despite some limitations in the reconstruction of the signal from the neighborhood of a polynomial expansion, it is possible to obtain an unsupervised estimation of the structure of objects from the optical flow. As it is discussed in the study cases of Chapter 6, this approach is used for producing reactive motion under unstructured but static scenarios.

4

Visually-guided locomotion

Contents

4.1 Introduction 65
4.2 Related work 66
4.3 Visual servoing 67
4.3.1 The interaction matrix 68
4.3.2 Controlling the robot effectors in joint space 69
4.4 Task definitions 70
4.4.1 The Walk task 70
4.4.2 The Look-at task 71
4.5 Egocentric localization 72
4.5.1 Sensory ego-cylinder 72
4.5.2 Vision-based localization 73
4.5.3 Object models 74
4.6 Case studies 76
4.6.1 Materials 76
4.6.2 CS-I: Simulation of the approach to a salient object 77
4.6.3 CS-II: Placement for the spatial reference system 79
4.6.4 CS-III: Approaching a real object 89
4.7 Conclusions 91

Introduction

Humanoid robots are designed to resemble the body and comportment of human beings. Among the behavior repertoire, the possibility of visually positioning in relation to stimuli is crucial for individual adaptation and relies on the on-board sensory system. Vision-based locomotion is a challenging task for walking robots. As discussed in the precedent chapter, unlike human beings which possess a extremely sophisticated visual

sense, the majority of the research in humanoid vision has employed general purpose cameras. Given the low quality of the images captured on-board, several studies have fixed the sensors on the environment for obtaining reliable localization. The main disadvantages of this approach are the need for adaptations on the scene, and the disregard of the corporal metaphor. Some studies have considered on-board solutions, though they have relied on extensive knowledge about the environment, consequently, the results have been obtained under controlled conditions.

In this chapter the problem of egocentric on-board localization for autonomous walk is investigated. The top-down segmentation technique discussed in Sec. 3.4.3 is integrated to a behavior model of the task, so visual servoing schemes are used to control independently the walk and the head motion of the robot. Given that the knowledge available to the agent is distributed, embodiment in the form of eye-centered and body-centered placements for the sensory ego-cylinder is investigated. Thus, the chapter starts by presenting relevant works in the field, and discussing the visual servoing framework. From the proposal of the behavior scheme three case studies are conducted in Nao. The possibility of obtaining autonomous and robust visually-guided walk is assessed in both simulations and real experiments.

Related work

Early research on humanoid localization have resorted to external cameras to extract information (e.g., Lewis & Simo [108], and Michel et al. [121]), due to difficulties in the processing of images captured on-board. Despite the fact of obtaining higher quality images for the task, the use of extra-corporeal sensors present several disadvantages. One is the fact that the robot may occlude the sensors, thus compromising the visual feedback. The approach is inflexible since the environment must be adapted to the task. This is in practice a form of rigorous control over extraneous variables, that conditions the autonomy of the solution. Extra-corporeal sensors also don't comply to the humanoid metaphor.

On-board solutions have been proposed under the visual servoing (VS) framework (which is detailed in the next section). A study by Dune et al. [59] has considered a monocular vision task with the robot HRP2. Given the walk style of the robot, the solution involved the cancellation of the oscillatory contribution to the control signal (also called the *sway motion*). In order to handle the image noise, the feature tracking technique based on the VV principle (see Sec. 3.3.4) by Comport et al. [49] was employed. A similar strategy was followed by Moughlbay et al. [128] when studying service tasks with the robot Nao. In general, some limitations of this approach could be mentioned. Since a realistic model of the object is required (e.g. a 3D model of the door or the drawer), the reusability of the solution to other stimuli is prevented. Also, the evolution on the task depends on the quality of the initial estimate of the object's pose (this information must be obtained at each trial). Moreover, since the sensor is considered to be dismembered, accurate estimates on the spatial evolution of the camera (visual odometry) is required. Thus, a relatively high frequency acquisition must be available, which may not be the case for some platforms.

Allocentric model-based navigation has been explored in the simultaneous localization and mapping (SLAM) research (see Thrun et al. [178]). Examples of contributions in the field are numerous. Just to mention a few, in the work by Hornung et al. [86] starting from a volumetric map of the environment, precise indoor localization is ob-

tained by adapting a range sensor to the robot's head. A work by Oriolo et al. [139] considered building the map on-line by fusing proprioceptive, inertial, and visual information, within an extended Kalman filter. In general, map-based navigation has produced impressive results, but it has also received some criticism. According to Shapiro [164], researchers in the field of embodied cognition disagree on the premise that organisms must firstly represent the environment for then transversing it. Indeed, this would not be adequate to unstructured or reactive situations. Moreover, from the practical point of view, map-based solutions present as a drawback requiring maintenance, where environmental changes must be systematically acknowledged.

Visual servoing

In visual servo control computer vision data is used to control the motion of a robot (see Chaumette & Hutchinson [42]). The approach relies on techniques from image processing, computer vision, and control theory. The goal of a vision-based control scheme is to minimize an error $\mathbf{e}(t)$, which is typically defined by

$$\mathbf{e}(t) = \mathbf{s}(m(t), a) - \mathbf{s}^*. \tag{4.1}$$

The vector of k visual features $\mathbf{s}(m(t), a)$ is defined from the image measurements $m(t)$ (e.g. point coordinates of the target, image coordinates of the centroid of an object, among others), and the set of parameters a that represent additional knowledge about the system (e.g. the camera intrinsic parameters, or a 3-D model of the target). The vector $\mathbf{s}^*$ contains the desired values of the features.

Depending on the characteristics of the task, a fixed goal can be considered where changes in $\mathbf{s}$ depend only on the camera motion. A more general situation can also be modeled, where the target is moving and the resulting image depends both on the camera and the target motion. In any case, visual servoing schemes mainly differ in the way $\mathbf{s}$ is designed. For image-based visual servo control (IBVS), $\mathbf{s}$ consists of a set of features that are immediately available in the image data. For position-based visual servo control (PBVS), $\mathbf{s}$ consists of a set of 3-D parameters that must be estimated from image measurements.

The relationship between the time variation of $\mathbf{s}$ and the camera velocity is given by

$$\dot{\mathbf{s}} = \mathbf{L}_\mathrm{s}\mathbf{v} \tag{4.2}$$

The spatial velocity of the camera is denoted by $\mathbf{v} = (\mathbf{v}_\mathrm{c}, \omega_\mathrm{c})$, with $\mathbf{v}_\mathrm{c}$ the instantaneous linear velocity of the origin of the camera frame, and ω_c the instantaneous angular velocity. The matrix $\mathbf{L}_\mathrm{s} \in \Re^{6\times k}$ is named the interaction matrix related to $\mathbf{s}$.

By combining Eq. (4.1) and Eq. (4.2), the relationship between $\mathbf{v}$ and the time variation of the error can be defined by

$$\dot{\mathbf{e}} = \mathbf{L}_\mathrm{e}\mathbf{e}, \tag{4.3}$$

An exponential decrease of the error e can be obtained by taking $\mathbf{v}$ as the input to the robot controller, so the velocity of the camera can is define by

$$\mathbf{v} = -\lambda \mathbf{L}_{\mathrm{e}}^{+} \mathbf{e}, \tag{4.4}$$

where λ is a proportional gain, ${\mathbf{L}_{\mathrm{e}}}^{+} \in \Re^{6\times k}$ is chosen as the Moore-Penrose pseudoinverse of $\mathbf{L}_{\mathrm{e}}$, that is $\mathbf{L}_{\mathrm{e}}^{+} = (\mathbf{L}_{\mathrm{e}}^{\mathrm{t}}\mathbf{L}_{\mathrm{e}})^{-1}\mathbf{L}_{\mathrm{e}}^{\mathrm{t}}$ when $\mathbf{L}_{\mathrm{e}}$ is of full rank 6. When $k = 6$ and $\det \mathbf{L}_{\mathrm{e}} \neq 0$ it is possible to invert $\mathbf{L}_{\mathrm{e}}$, obtaining the control $\mathbf{v} = -\lambda \mathbf{L}_{\mathrm{e}}^{-1}\mathbf{e}$.

For real visual servo systems it is not possible to know perfectly either $\mathbf{L}_{\mathrm{e}}$ or $\mathbf{L}_{\mathrm{e}}^{+}$, so an approximation or estimation of one of these two matrices must be done. So the control law is in fact

$$\mathbf{v} = -\lambda \widehat{\mathbf{L}_{\mathrm{e}}^{+}} \mathbf{e} \tag{4.5}$$

The interaction matrix

The analytical form of the interaction matrix depends on the type of sensor (e.g. 2D, 3D, omni-directional camera, etc.), the projection model used, and the features $\mathbf{s}$ selected. Conventional cameras are 2D sensors that employ perspective projection (see Sec. 3.3.3), thus, the definition of the iteration matrix to be discussed is based on the perspective projective geometry.

Image-Based Visual Servoing (IBVS)

In IBVS the definition of features $\mathbf{s}$ includes the camera intrinsic parameters to go from image measurements (expressed in pixels) to features. Commonly used features are points. Let $s_i = (x, y)$ be the pixel coordinates of an image point i related to the world coordinates (X, Y, Z); the interaction matrix $\mathbf{L}_{\mathrm{si}}$ is defined by

$$\mathbf{L}_{\mathrm{si}} = \begin{bmatrix} -1/Z & 0 & x/Z & xy & -(1+x^2) & y \\ 0 & -1/Z & y/Z & (1+y^2) & -xy & -x \end{bmatrix}. \tag{4.6}$$

Notice that the depth Z of point i relative to the camera frame is required. Thus, any control scheme that adopts this form of interaction matrix must be provided with an estimation of Z. Each point i allows the control of 2 degrees of freedom (DOF), in order to control more DOFs additional points are needed (e.g. 6 DOF would require of 3 points). The interaction matrix $\mathbf{L}_{\mathrm{s}}$ is obtained from the $\mathbf{L}_{\mathrm{si}}$ related to each feature point, such that

$$\mathbf{L}_{\mathrm{s}} = \begin{bmatrix} \mathbf{L}_{\mathrm{s}1} \\ \vdots \\ \mathbf{L}_{\mathrm{s}n} \end{bmatrix} \tag{4.7}$$

Position-Based Visual Servoing (PBVS)

In PBVS the features $\mathbf{s}$ are defined from the pose of the camera expressed in a reference frame. Thus, it requires of the intrinsic and extrinsic parameters of the camera, and the 3-D model of the observed object.

Three coordinate frames are defined: the actual camera frame C, the desired camera frame C^*, and a reference frame O attached to the object. The feature vector $\mathbf{s} = [\mathbf{p}\ \theta\mathbf{u}]^{\mathrm{t}}$

depends on the translation $\mathbf{p}$ and the angle/axis parameterization for the rotation $\theta\mathbf{u}$. A convenient choice is to define $\mathbf{p} = {}^{C^*}\mathbf{p}_C$, such that $\mathbf{s}^* = 0$ and $\mathbf{e} = \mathbf{s}$.

The iteration matrix (Chaumette & Hutchinson [42]) is given by

$$\mathbf{L}_e = \begin{bmatrix} \mathbf{R} & 0 \\ 0 & \mathbf{L}_{\theta u} \end{bmatrix} \tag{4.8}$$

where $\mathbf{R} = {}^{C}\mathbf{R}_{C^*}$ is the rotation matrix that expresses the orientation of the current camera frame relative to the desired frame. The rotational component $\mathbf{L}_{\theta u}$ is defined by

$$\mathbf{L}_{\theta u} = \mathbf{I}_3 - \frac{\theta}{2}[\mathbf{u}]_\times + \left(1 - \frac{\text{sinc}(\theta)}{\text{sinc}^2\left(\frac{\theta}{2}\right)}\right)[\mathbf{u}]_\times^2, \tag{4.9}$$

I_3 is the 3×3 identity matrix, $\text{sinc}(\theta)$ is the sinus cardinal (i.e., $\theta\text{sinc}(\theta) = \sin(\theta)$ and $\text{sinc}(0) = 1$), and $[\mathbf{u}]_\times$ is the skew symmetric matrix of the axis vector $\mathbf{u}$. Thereby, $\mathbf{L}_{\theta u}$ is define such that $\mathbf{L}_{\theta u} = \mathbf{L}_{\theta u}^{-1}\theta\mathbf{u} = \theta\mathbf{u}$.

The control law is given by

$$\begin{bmatrix} \mathbf{v}_c \\ \omega_c \end{bmatrix} = \begin{bmatrix} -\lambda \mathbf{R}^t \mathbf{p} \\ -\lambda\theta\mathbf{u} \end{bmatrix}. \tag{4.10}$$

Controlling the robot effectors in joint space

A control law based on VS can be defined in the joint space in order to ensure the operation of the robot end effectors. For the *eye-to-hand* configuration (Chaumette & Hutchinson [43]), that is, when vision data is acquired from a pan-tilt camera mounted on the humanoid's head, the control law has the form

$$\dot{\mathbf{s}} = \mathbf{J}_s\dot{\mathbf{q}} + \frac{\partial \mathbf{s}}{\partial t}, \tag{4.11}$$

where the derivative term is the time variation of $\mathbf{s}$ due to potential motion of the object, $\mathbf{J}_s \in \Re^{k\times 6}$ is the feature Jacobian matrix. It is related to the interaction matrix $\mathbf{L}_s$, so

$$\mathbf{J}_s = \mathbf{L}_s {}^{C}\mathbb{V}_B {}^{B}\mathbf{J}(\mathbf{q}), \tag{4.12}$$

where the matrix ${}^{B}\mathbf{J}(q)$ is the robot Jacobian expressed in the end-effector reference frame B. The matrix ${}^{C}\mathbb{V}_B$ is the spatial transformation of velocities expressed in frame B. Let the rigid body transformation from the camera frame C to the base frame B be denoted by ${}^{C}\mathbf{R}_B \in \text{SE}(3)$, and the translation be ${}^{C}\mathbf{p}_B$, the general form of ${}^{C}\mathbb{V}_B$ is given by

$${}^{C}\mathbb{V}_B = \begin{bmatrix} {}^{C}\mathbf{R}_B & [{}^{C}\mathbf{p}_B]_\times {}^{C}\mathbf{R}_B \\ 0 & {}^{C}\mathbf{R}_B \end{bmatrix}. \tag{4.13}$$

An exponential decoupled decrease of $\mathbf{e} = \mathbf{s} - \mathbf{s}^*$ can be obtained in the joint space, trough the control scheme

$$\dot{\mathbf{q}} = -\lambda\widehat{\mathbf{J}_e}^{+}\mathbf{e} - \widehat{\mathbf{J}_e}^{+}\widehat{\frac{\partial \mathbf{e}}{\partial t}}. \tag{4.14}$$

The contribution of the second term of the control law anticipates the variation of $\mathbf{s}^*$ and removes the tracking error that it would produce (i.e. it is null when the object is static).

Task definitions

The behavior under study is the approach to a given face of a static object, by walking on a plane in a scene without obstacles. Figure 4.1 shows the definition of the task frames. The desired behavior is obtained from the simultaneous execution of the Walk and the Look-at motor tasks.

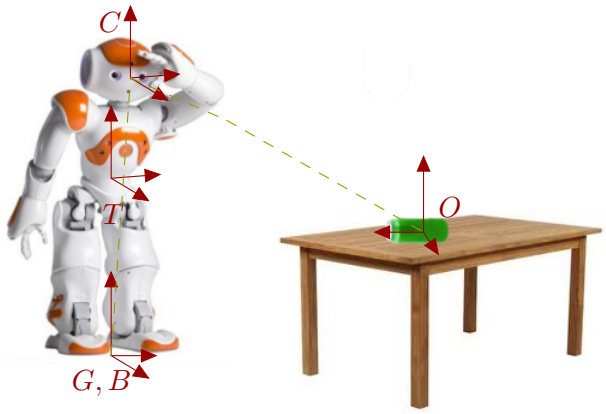


Figure 4.1 – Definition of the reference frames to solve the localization task. G corresponds to the walk primitive frame, B is the movable reference frame for the Walk task (in the figure it coincides with G, though in Sec. 4.6.3 different locations for B are studied), C is the camera frame, T is the torso frame, and O is the object frame.

The Walk task

The localization ${}^B\zeta$ of the object is represented by the four parameters

$$ {}^B\zeta = \begin{bmatrix} \rho & \theta & \iota & \phi \end{bmatrix}^{\mathrm{t}}, \tag{4.15} $$

where ρ, θ, and ι are position components, respectively the distance, the bearing, and the height of the center of the object. The parameter ϕ corresponds to the heading of the object. It is estimated by the difference between the projection on the motion plane of the mean normal direction to the tracked face of the object, and the projection of the robot Saggital plane (see Fig. 2.12).

Therefore, starting from the knowledge of a desired ego-centric perception of the object ${}^B\zeta^*$, the agent has to autonomously return as close as possible to such state once disturbed. The behavior can be viewed as a PBVS regulation task where the control parameters include a 2D pose (i.e. the object height ι is assumed to be constant). Formally, the approach error ${}^B\mathbf{e}_1$ (see Eq. (4.1)) expresses the desired configuration τ^* of the body (see Fig. 4.2) in the actual egocentric perspective, such that

$$ {}^B\mathbf{e}_1 = \begin{bmatrix} \tau_{\rho^*} & \tau_{\theta^*} & \tau_{\phi^*} \end{bmatrix}^{\mathrm{t}}. \tag{4.16} $$

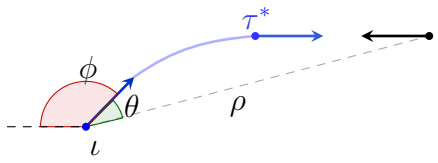


Figure 4.2 – Top view of the localization parameters observed by the agent. The object center and the heading direction is represented in black. The agent position and Saggital projection direction is represented in blue. A desired configuration in relation to the object is represented by τ^*, a possible trajectory to approach the object is illustrated in light blue.

Notice that ${}^B\mathbf{e}_1 = 0$ once the agent is at the desired location.

The walk task also produces a prediction ${}^B\tilde{\zeta}$ for the next observation of the object, based on the assumption of deterministic motion ${}^B\bar{\mathbf{m}} = [\bar{\rho}\ \bar{\theta}\ \bar{\phi}]^{\mathrm{t}}$ (the motion request ${}^B\bar{\mathbf{m}}$ is defined in Eq. (4.30)), that is, an ideal noise-free robot moving at constant velocity (see Fig. 4.3). More specifically, ${}^B\tilde{\zeta}$ is defined by

$$ {}^B\tilde{\zeta} = \begin{bmatrix} \sqrt{\rho^2 - 2c\rho\bar{\rho} + \bar{\rho}^2} \\ \mathrm{atan2}(a,b) - \bar{\phi} \\ \iota \\ \phi - \bar{\phi} \end{bmatrix}, \tag{4.17} $$

where $c = \cos(\theta - \bar{\theta})$, $a = \sin(\theta)\rho - \sin(\bar{\theta})\bar{\rho}$, $b = \cos(\theta)\rho - \cos(\bar{\theta})\bar{\rho}$, and $(\ ^-\ )$ denoting the elements of $\bar{\mathbf{m}}$.

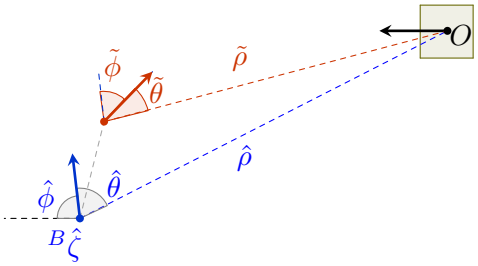


Figure 4.3 – Top view of the localization prediction. The parameter ι is not shown since it is assumed do be constant. The heading direction of the object is shown in black. The motion direction of the agent is shown in blue. The prediction ${}^B\tilde{\zeta}$ from Eq. (4.17) is shown in orange.

The Look-at task

The goal of the Look-at task is to maintain the object centered in the field of view by controlling the articulated neck of the robot. Motion is expressed with respect to the reference frame T, which is attached to the trunk (see Fig. 4.1). Two internal subtasks are executed in sequence. One is the open-loop direction of the gaze toward the predicted location of the object, under deterministic motion assumption. The other is the regulation in close-loop of the view direction to maintain the object centered in the field of view.

Let $\mathbf{q} = [\alpha\ \beta]^{\mathrm{t}}$ be the posture of the pitch α and the yaw β of the Nao's neck. In the predictive subtask, the head orientation error $\mathbf{e}_2$ is expressed in the joint space such that

$$ \mathbf{e}_2 = \mathbf{q} - \mathbf{q}^*. \tag{4.18} $$

The desired posture $\mathbf{q}^*$ is obtained from the prediction of the pose of the robot in relation to the object, by assuming that the configuration of the legs and the trunk is constant after the motion, thus

$$\mathbf{q}^* = \begin{bmatrix} \text{atan2}(\tilde{\iota}, \cos(\tilde{\theta})\tilde{\rho}) \\ \tilde{\theta} \end{bmatrix}. \tag{4.19}$$

A proportional regulation of the neck posture with gain λ is obtained by

$$\dot{\mathbf{q}} = \lambda \mathbf{e_2}. \tag{4.20}$$

The second subtask corresponds to a close-loop IBVS scheme (see Eq. (4.14)). As illustrated in Fig. 4.4, the desired retinal motion to maintain the object centered in the field of view is received by visual feedback. Thereby, the feature $\mathbf{s}$ is the image point corresponding to the centroid of the salient blob. The task error ${}^{\mathrm{T}}\mathbf{e}_3$ is defined such that

$${}^{\mathrm{T}}\mathbf{e}_3 = \begin{bmatrix} \mathbf{s}_\mathrm{x} - \mathrm{i}_\mathrm{x} \\ \mathbf{s}_\mathrm{y} - \mathrm{i}_\mathrm{y} \end{bmatrix}, \tag{4.21}$$

where $(\mathrm{i}_\mathrm{x}, \mathrm{i}_\mathrm{y})$ is the coordinate of the center of the image.

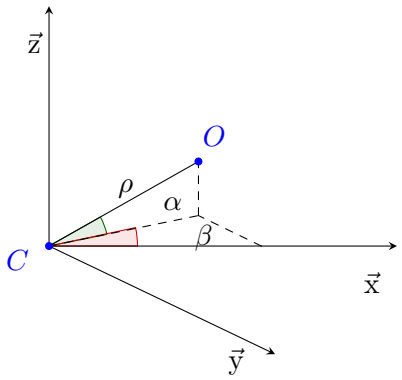


Figure 4.4 – Illustration of the Look-at task. The center of the object is denoted by O. After the head correction, the x-axis of the camera frame C will be aligned with the direction $\overline{CO}$.

Egocentric localization

Sensory ego-cylinder

Inspired by studies on mammalian neural systems, Peters et al. [142] have proposed the concept of Sensory Ego-Sphere (SES) for humanoids, which is a computational structure in charge of integrating different sensory modalities. The SES would exert the role of a short-term episodic memory, providing the location and orientation of stimuli with respect to the agent. Though less general, cylindrical geometry is more appealing to represent positions on a plane. It is also easier to implement and computationally more efficient to query. Thereby, an ego-cylinder principle for localization is adopted. As shown in Fig. 4.5, the four parameters defined in Eq. (4.15) are persisted in the structure. The first three parameters represent the position of the center of the stimulus in cylindrical coordinates, and the fourth parameter represents the heading direction of the object.

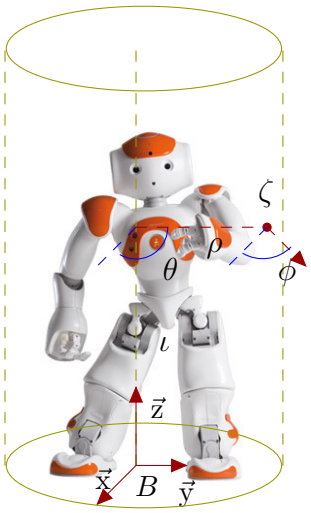


Figure 4.5 – Representation of the ego-cylinder localization. In the image, B corresponds to the base frame, and ζ represents the localization of an object in the environment. The heading direction ϕ is represented emerging from the cylinder's surface.

The origin of the ego-cylinder can be fixed to different parts of the body. There is no agreement in the literature on the placement for this structure. A work by Bodiroza et al. [20] has for instance fixed the SES on the robot's neck, whereas in Ruesch et al. [157] it was centered at the head, fixed with respect to the orientation of the torso. In Sec. 4.6.3 different placements for the ego-cylinder are studied. For the moment, in other to illustrate the concept, and taking into account that the walk primitive of the robot uses the reference frame G to express motion (see Fig. 4.1), the origin B is placed at the same location of G.

Vision-based localization

The localization of the object is obtained from the task frames described in Fig. 4.1. Let the homogeneous transformation ${}^{B}\mathbf{T}_{\mathrm{O}}$ between the base frame and the object frame be defined by

$$ {}^{B}\mathbf{T}_{\mathrm{O}} = {}^{B}\mathbf{T}_{\mathrm{C}}(\mathbf{q}){}^{\mathrm{C}}\mathbf{T}_{\mathrm{O}}, \tag{4.22} $$

so the transformation ${}^{B}\mathbf{T}_{\mathrm{C}}(\mathbf{q})$ expresses the camera frame C in the base frame B, and depends on the actual joint configuration $\mathbf{q}$ of the robot. Similarly, the transformation ${}^{\mathrm{C}}\mathbf{T}_{\mathrm{O}}$ expresses the object frame O in frame C, and is determined from the 3D pose

$$ {}^{\mathrm{C}}\mathbf{o} = \begin{bmatrix} \xi & \vartheta \end{bmatrix}^{\mathrm{t}} = \begin{bmatrix} \begin{bmatrix} X & Y & Z \end{bmatrix} & \begin{bmatrix} \gamma & \beta & \phi \end{bmatrix} \end{bmatrix}^{\mathrm{t}}, \tag{4.23} $$

where ξ is the position component and ϑ is the orientation component. The calculation of ${}^{\mathrm{C}}\mathbf{o}$ is obtained by computer vision processing so a rough 3D container encompassing the object is fit to the segmented region on the image. This is discussed in the next section.

The localization of the object in the ego-cylinder is obtained from ${}^{B}\mathbf{T}_{\mathrm{O}}$ by expressing the position of the center of frame O in cylindrical coordinates, and adding the heading direction ϕ, as defined in Eq. (4.15). The transformation ${}^{B}\mathbf{T}_{\mathrm{B}^*}$ between the current placement of B and the desired placement B^* in relation to the object, is given by

$$ {}^{B}\mathbf{T}_{\mathrm{B}^*} = {}^{B}\mathbf{T}_{\mathrm{O}}\,{}^{\mathrm{O}}\mathbf{T}_{\mathrm{B}^*}, \tag{4.24} $$

where ${}^{\mathrm{O}}\mathbf{T}_{\mathrm{B}^*}$ is defined by kinesthetic demonstration. That is, by positioning the robot at the desired configuration in relation to the object. Thereby, the estimation of the localization error in Eq. (4.16) is obtained from ${}^{B}\mathbf{T}_{\mathrm{B}^*}$.

Object models

The estimation of the object 3D pose in the camera frame relies on the region-based segmentation technique described in Sec. 3.4.3. Thus, the pose is observed by establishing a correspondence between the 3D model of the object and its perspective projection on the image plane. The quality of the estimation hardly depends on the segmentation available. Relatively good segmentations can be obtained with the MRF algorithm, so the model of the object is approximated by a rough 3D container that can virtually encompass or be attached to the surface of the object. This provides flexibility and is a reasonable assumption for convex objects. The dimension of the object are known, and the color model is obtained by supervised demonstration. Next, the modeling of two of these containers and how to estimate the pose from the image blob is given as examples.

Cylindrical Wrapper

The frame O is attached to the center of mass of the model as shown in Fig. 4.6. Given the bilateral symmetry of the shape, the projection of the object in the image plane is not affected by the rotation β around O_{y}, so it is assumed to be constant.

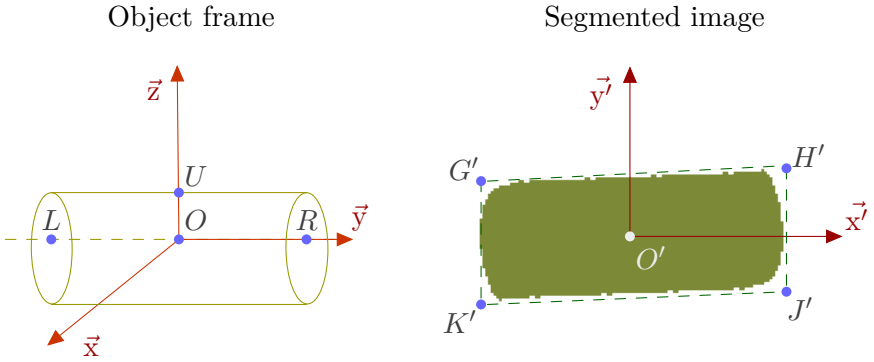


Figure 4.6 – Cylindrical object model. On the left the 3D model of the object, the placement of frame O, and the definition of four points of interest. On the right the segmented blob and the definition of image features from the oriented bounding box.

Depth estimation

The blob is assumed to be approximately centered on the image due to the action of the Look-at task. Thus, calculations over a clipped projection of the object are avoided. As illustrated in Fig. 4.7, in order to estimate the position ${}^{\mathrm{C}}\mathbf{o}_{\xi}$ of frame O, the observation of the depth $\hat{Z}$ for ${}^{\mathrm{C}}L$ and ${}^{\mathrm{C}}R$ is obtained by

$$\hat{Z} = \frac{r}{r'}\sqrt{r'^2 + f^2}, \tag{4.25}$$

where r is the radius of the cylinder, r' is its projection on the image plane, and f is the focal length of the camera. The projection of the radius r'_{l} and r'_{r} for ${}^{\mathrm{C}}L$ and ${}^{\mathrm{C}}R$ is respectively taken such that $r'_{\mathrm{l}} = \|G' - K'\|/2$ and $r'_{\mathrm{r}} = \|H' - J'\|/2$. This model produces the best results when the orientation component ${}^{\mathrm{C}}\mathbf{o}_{\phi} = \pi$ (see Eq. (4.23)).

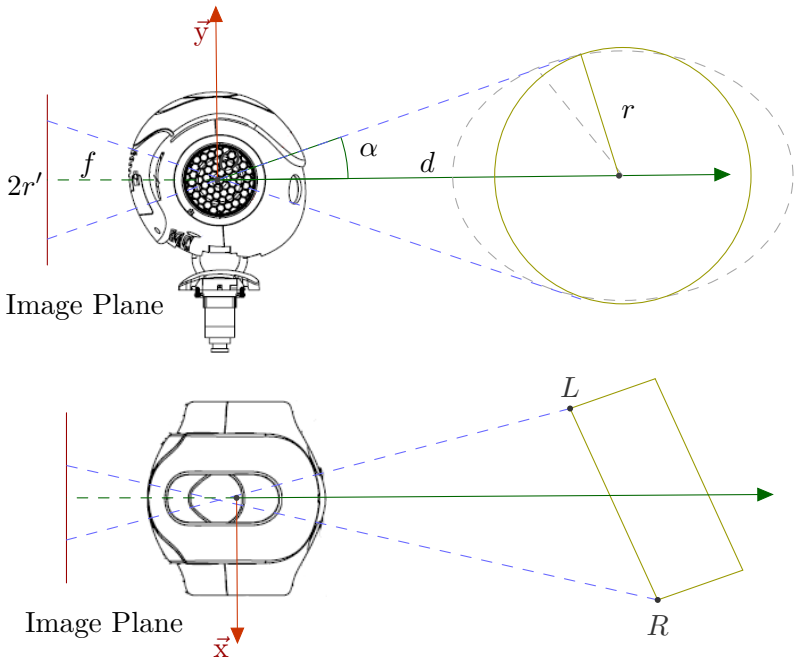


Figure 4.7 – Estimation of the object's depth. On the top, the model assumes a view perspective ${}^{\mathrm{C}}\mathbf{o}_{\phi} = \pi$. Bellow, the $\vec{x}\vec{z}$ visualization of the scenario, where the circumference corresponds to an ellipse and the distance from the projective ray and the center of frame O is larger than r.

Position estimation

The position of a point P in 3D can be recovered from the image projection P' by applying Eq. 3.1, such that

$$P = \left(\frac{((P'_{\mathrm{x}} - C'_{\mathrm{x}})P_{\mathrm{Z}})}{f}, \frac{((P'_{\mathrm{y}} - C'_{\mathrm{y}})P_{\mathrm{Z}})}{f}, Z \right), \tag{4.26}$$

where C' is the image center, and Z is the point's depth. Thereby, the observation of points ${}^{\mathrm{C}}L$, ${}^{\mathrm{C}}R$ are obtained through Eq. (4.26). ${}^{\mathrm{C}}\hat{O} = ({}^{\mathrm{C}}\hat{L} + {}^{\mathrm{C}}\hat{R})/2$, and ${}^{\mathrm{C}}\hat{U} = {}^{\mathrm{C}}\hat{O} + (0, 0, r)$. The position component ${}^{\mathrm{C}}\mathbf{o}_{\xi}$ in Eq. (4.23) is such that

$${}^{\mathrm{C}}\mathbf{o}_{\xi} = {}^{\mathrm{C}}\hat{O}. \tag{4.27}$$

Orientation estimation

The orientation component ${}^{\mathrm{C}}\mathbf{o}_{\vartheta}$ is obtained from the rotation matrix ${}^{\mathrm{C}}\mathbf{W}$, thus

$$^{\mathrm{C}}\mathbf{W} = \begin{bmatrix} \mathbf{s} & \mathbf{n} & \mathbf{a} \end{bmatrix} = \begin{bmatrix} H & V & (H \times V) \end{bmatrix}, \tag{4.28}$$

with $H = (^{\mathrm{C}}R - {}^{\mathrm{C}}L)/|^{\mathrm{C}}R - {}^{\mathrm{C}}L|$, and $V = (^{\mathrm{C}}U - {}^{\mathrm{C}}O)/|^{\mathrm{C}}U - {}^{\mathrm{C}}O|$.

Rectangular Surface

Rectangles are useful geometric models for tracking surfaces in walls, doors and furnitures. The model is simpler than the previous case since it is a 2D shape. The points defining O correspond to those of Fig. 4.6. The features tracked in the image are the same of the previous case. The calculation for the depth of O is given by

$$d(h, h', f) = \frac{hf}{h'}, \tag{4.29}$$

where f is the focal distance of the camera, h is the height of the rectangle, and h' is the image projection of h. The relation between the image features and the location of $^{\mathrm{C}}R$, $^{\mathrm{C}}L$, $^{\mathrm{C}}O = \mathrm{mean}(^{\mathrm{C}}L, ^{\mathrm{C}}R)$, and $^{\mathrm{C}}U$ is similar to the previous case.

Case studies

Three studies are conducted in order to evaluate distinct aspects of the task model. In the first study a simulated scene is designed so a single object is salient, and the agent's task is to approach the object by doing holonomic walk. The objective is to verify the plausibility of the model. The second study focuses on the aspect of embodiment. Thus, different placements (i.e. body- and eye-centered) for the origin of the sensory ego-cylinder are studied. In the comparison many aspects are analyzed, such that the computational cost, the precision, and the robustness to noise. In the last study a real task with the robot Nao is evaluated so it approaches a yellow card in the scene.

Materials

The platform is the humanoid robot Nao by Aldebaran Robotics. The control program is implemented in the C++ programming language. The images are captured at 320×240 pixels resolution. The vision processing is obtained with the support of the OpenCV 2.4.8 library. The robot functionalities are accessed through the naoqi 1.14 library. The algorithms are developed in the Eclipse Juno IDE under Ubuntu 12.04.5 LTS (Precise Pangolin). The simulations are performed in the Webots robot simulator 7.4.0 by Cyberbotics. The results are processed in Gnu Octave 3.2.4 and the KNIME data analytics, reporting and integration platform 2.10.4. The on-board calculations relied on an ATOM Z530 1.6GHz CPU, with 1 GB RAM, 2 GB flash memory, and 4 flash memory dedicated to user purposes. The study also included a DELL Vostro 1500 laptop (Intel Core 2 Duo 1.8GHz 800Mhz, 4.0GB DDR2 667MHz RAM, 256MB NVIDIA GeForce 8600M GT).

CS-I: Simulation of the approach to a salient object

This study considered holonomic correction of the agent's posture in relation to a salient stimulus. The observed localization error ${}^{B}\hat{\mathbf{e}}_1$ (see Eq. (4.16)) between the current and the desired configuration is obtained from the matrix ${}^{B}\mathbf{T}_{B^*}$ (see Eq. (4.24)). Given the sources of uncertainties in the observation (e.g., the 3D model imprecisions), the study aims to verify whether the Walk task would steer the agent toward the desired state. Thereby, ${}^{B}\overline{\mathbf{e}_1}$ is defined by saturating the observed error. The bounds corresponded to the radial distance $\overline{\rho} = 0.1$ meters (m), the bearing $\overline{\theta} = \pi$ radians (rad), and the heading $\overline{\phi} = \pi/12$ rad. A proportional correction ${}^{B}\overline{\mathbf{m}}$ according to the magnitude of the individual components of ${}^{B}\overline{\mathbf{e}_1}$ is applied, such that

$$ {}^{B}\overline{\mathbf{m}} = \lambda {}^{B}\overline{\mathbf{e}_1}, \tag{4.30} $$

where the normalized vector λ is defined from the motion bounds as follows

$$ \lambda_i = \frac{|\overline{\mathbf{e}_{1i}}|}{(|\overline{\mathbf{e}_{1\rho}}|/\overline{\rho} + |\overline{\mathbf{e}_{1\theta}}|/\overline{\theta} + |\overline{\mathbf{e}_{1\phi}}|/\overline{\phi})}. \tag{4.31} $$

The motor tasks are implemented with the help of the naoqi environment, which provides a Program Aplication Interface (API) for sending commands to the robot through parameterizable routine calls. In the Walk task the agent is stopped once all the components of the observed localization error ${}^{B}\hat{\mathbf{e}}_1$ are smaller than a given threshold ϵ. The tolerance considered is a radial distance $\epsilon_\rho = 0.05$ m, the bearing $\epsilon_\theta = 0.04$ rad, and the heading $\epsilon_\phi = 0.1$ rad. The walk primitive can be controlled both in position or velocity. The position version is used since it showed a more accurate performance. The primitive receives position commands in Cartesian coordinates (so the cylindrical coordinates are accordingly converted). The mean walk velocity set for the robot is around $\mathbf{v} = [0.022 \text{ m/s } 0.04 \text{ m/s } 0.106 \text{ rad/s}]^{\mathrm{t}}$.

The Look-at task is also controlled in position. The correction of the head posture is obtained by assuming constant velocity along the time interval. A tolerance $\epsilon = 0.03$ rad is admitted for convergence of $\mathbf{e}_2$ (see Eq. (4.18)), and a tolerance for 10 pixels is accepted for ${}^{\mathrm{T}}\mathbf{e}_3$ (see Eq. (4.21)). The head posture is regulated independently from the walk (i.e. the tasks run in parallel), which means that the motion induced by the Walk task can affect the convergence of the Look-at task, notably, at slow turning of the head. Thereby, a velocity profile of 4 rad/s is employed so convergence for the Look-at task is obtained.

Experiments

In order to assess the performance under modeling imprecisions, a simulated environment was designed in Webots. As illustrated in Fig. 4.8, the object of interest is a red soda can placed over the table. The texture of the can is clearly distinguishable from the rest of the room to avoid multiple saliency detection. The desired configuration ${}^{B^*}\mathbf{T}_{\mathrm{O}}$ is specified by positioning the robot in front of the can. Two experiments are designed. In the first one, the robot is moved away from the desired pose such that it has to return autonomously to the desired configuration. In the second experiment disturbances are introduced in the task. Thus, the robot is moved while approaching the object (i.e. the robot kidnapped problem as described in Sec. 2.4.1).

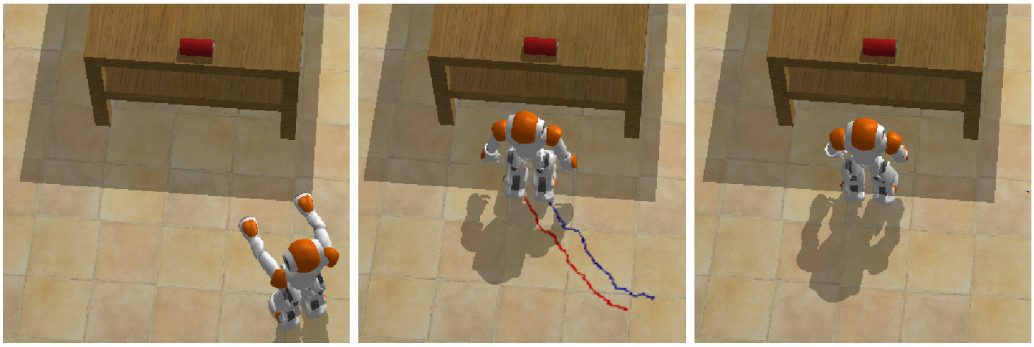


Figure 4.8 – The approach task modeled in Webots. On the left the robot's original pose. In the center the followed trajectory. On the right the desired pose with respect to the red can. As noticed, despite the modeling errors the robot was able to converge to a location very similar to the demonstration.

Results

The evolution of the localization along the followed trajectory is shown in Fig. 4.9. A comparison on the precision of the localization is also given in Fig. 4.10, where the difference between the on-board estimations and the ground truth measurements provided by Webots is shown. As expected, the more distant from the object the less precise the estimations are. This is due to the effect of the perspective projection (i.e. distant objects are perceived as smaller, so the 3D model of the object is fit to smaller segmentations, producing imprecise estimates). The most affected component is the observation of the heading direction of the object, which depends on the quality of the blob contours. Though, as the robot approached the target, the precision improved enough as to allow it to convergence to a location very similar to the demonstration. In relation to the second experiment, it was observed that despite the disturbances applied, whenever the object remained within the field of view, the robot was able to approach it.

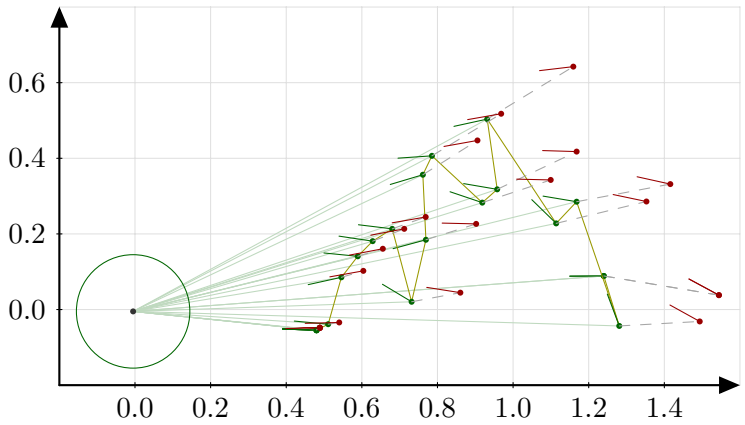


Figure 4.9 – Egocentric visualization of the localization as perceived in B. The circumference represents the ego-cylinder. In red the real values, in green the estimations. Distances are expressed in m.

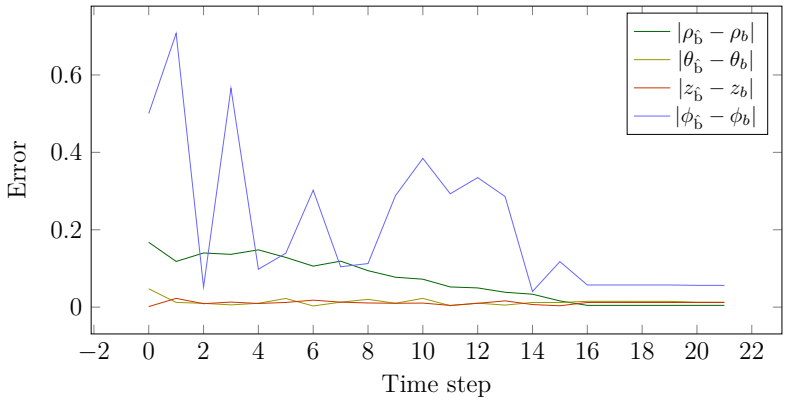


Figure 4.10 – Evolution of the localization error between the estimations $\hat{b}$ and the measurements b.

Discussion

This study has explored a behavior scheme based on vision. The modeling of the task followed the visual servoing framework. That is, the PBVS and IBVS control techniques were employed simultaneously to maintain the object of interest in the field of view and to steer the robot to the desired pose in relation to the object. The solution considered the motion primitives of walking and directing the head. The processing of the localization was based on the design of a sensory ego-cylinder, where the 3D position of the center of the object and its heading direction on the plane were represented. This information was obtained from a binary image and a 3D model of the object. A region-based whole segmentation technique was employed for the binarization. The monocular vision mapping from the salient region on the image to the 3D space relied on a rough model encompassing the object. In this study a fairly simple situation was simulated where a single object was salient, and there were no obstacles between the robot and the object. Under these conditions, it was observed that the farther the robot was, the less precise is the localization obtained. The most affected component was the estimation of the object's heading. Though, the agent was able to accomplish the task when the object was visible, even when being kidnapped.

CS-II: Placement for the spatial reference system

Five placements were studied for the base frame B. As listed in Tab. 4.1 and illustrated in Fig. 4.11, three locations considered the spatial constraint imposed by the localization model (i.e., taking the z-axis perpendicular to the motion plane), whereas the others did not. Moreover, among the locations some are body-centered whereas others are eye-centered.

B	**Description**	**Type**	**Constraint**
G	Ground	body-centered	Yes
T_g	Torso	body-centered	Yes
E_g	Eye_g	sensory-centered	Yes
N	Neck	body-centered	No
E	Eye	sensory-centered	No

Table 4.1 – The studied placements for the reference frame.

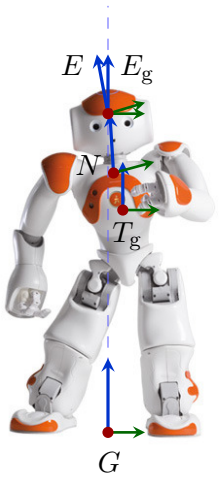


Figure 4.11 – Evaluated placements for the base frame B. For each frame the z-axis is represented in blue, the y-axis in green, and the x-axis goes towards the reader's direction so is represented as a red dot.

The definition of the placements relied on the direct geometrical model of the robot. For this, the body of Nao was modeled as a set of interconnected serial structures that depart from the common reference frame T, which is placed at the center of the torso. The body kinematics is defined according to the modified Denavit and Hartenberg notation (see Khalil & Kleinfinger [97]). Table 4.2 presents the geometric parameters of the robot model. Figure 4.12 illustrates the body structure, excluding the representation of the arms since they are irrelevant to the studied behavior.

j	a_j	σ_j	α_j	d_j	θ_j	r_j
0	T	2	$-3\pi/4$	0	$\pi/2$	0
1	0	0	0	0	q_1	0
2	1	0	$\pi/2$	0	$q_2 - 3\pi/4$	0
3	2	0	$\pi/2$	0	q_3	0
4	3	0	0	$-d_4$	q_4	0
5	4	0	0	$-d_5$	q_5	0
6	5	0	$-\pi/2$	0	$q_6 + \pi$	0
7	T	2	$-\pi/4$	0	$\pi/2$	0
8	7	0	0	0	q_8	0
9	8	0	$\pi/2$	0	$q_9 + 3\pi/4$	0
10	9	0	$\pi/2$	0	q_{10}	0
11	10	0	0	$-d_{11}$	q_{11}	0
12	11	0	0	$-d_{12}$	q_{12}	0
13	12	0	$-\pi/2$	0	$q_{13} + \pi$	0
14	T	2	0	0	0	r_{14}
15	14	0	0	0	q_{15}	0
16	15	0	$-\pi/2$	0	q_{16}	0

Table 4.2 – Modified Denavit & Hartenberg parameters for Nao. For the frame j, a_j is the predecessor, σ_j is the joint type (0 for revolute, 1 for a prismatic joint, and 2 for fixed joint), α_j is the angle between $\vec{z}_{j-1}$ and $\vec{z}_j$ about $\vec{x}_{j-1}$, d_j is the distance between $\vec{z}_{j-1}$ and $\vec{z}_j$ about $\vec{x}_{j-1}$ ($d_4 = d_{11} = 0.1$ m and $d_5 = d_{12} = 0.1029$ m), θ_j is the angle between $\vec{x}_{j-1}$ and $\vec{x}_j$ about $\vec{z}_j$, and r_j is the distance between $\vec{x}_{j-1}$ and $\vec{x}_j$ about $\vec{z}_j$ ($r_{14} = 0.1265$ m).

The transformations from T to the effectors frames (i.e. the left foot frame F_l, the right foot frame F_r, and the camera frame C mounted at the forehead) are obtained by

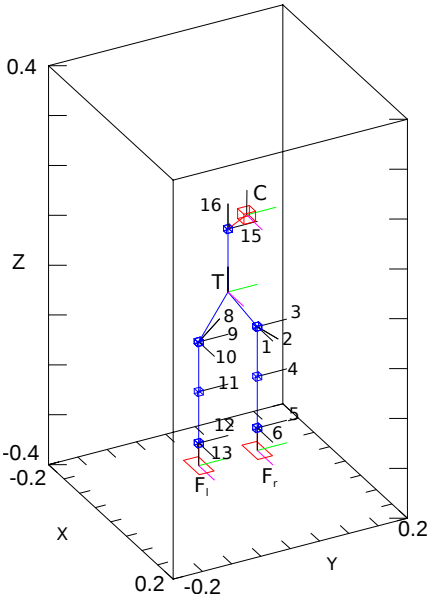


Figure 4.12 – Geometrical model of the robot Nao. Distances are expressed in m. The orientation of the z-axis are projected in black along the structure. For the base frame T and the effector frames C, F_l, and F_r, the x- and y-axis are also plotted in magenta and green respectively. The active joints are numerated and represented by blue boxes. The effectors are represented in red.

$$\begin{aligned} {}^{\mathrm{T}}\mathbf{T}_{\mathrm{F_l}} &= {}^{\mathrm{T}}\mathbf{T}_0\,{}^{0}\mathbf{T}_6(\mathbf{q})\,{}^{6}\mathbf{T}_{\mathrm{F_l}} \\ {}^{\mathrm{T}}\mathbf{T}_{\mathrm{F_r}} &= {}^{\mathrm{T}}\mathbf{T}_7\,{}^{7}\mathbf{T}_{13}(\mathbf{q})\,{}^{13}\mathbf{T}_{\mathrm{F_r}} \\ {}^{\mathrm{T}}\mathbf{T}_{\mathrm{C}} &= {}^{\mathrm{T}}\mathbf{T}_{14}\,{}^{14}\mathbf{T}_{16}(\mathbf{q})\,{}^{16}\mathbf{T}_{\mathrm{C}}, \end{aligned} \tag{4.32}$$

where ${}^{6}\mathbf{T}_{\mathrm{F_l}} = \mathrm{Trans}(r_\mathrm{f}, 0, 0)\mathrm{Rot}(\vec{\mathrm{y}}, -\pi/2)$, ${}^{13}\mathbf{T}_{\mathrm{F_r}} = \mathrm{Trans}(r_\mathrm{f}, 0, 0)\mathrm{Rot}(\vec{\mathrm{y}}, -\pi/2)$, ${}^{16}\mathbf{T}_{\mathrm{C}} = \mathrm{Trans}(r_\mathrm{xc}, -r_\mathrm{yc}, 0)\mathrm{Rot}(\mathrm{x}, \pi/2)$, $r_\mathrm{f} = 0.04519$ m, $r_\mathrm{xc} = 0.05871$ m, and $r_\mathrm{yc} = 0.06364$ m.

The frame Ground

The frame G is placed between both feet by taking the z-axis perpendicular to the motion plane. It is the same frame employed by the walk primitive of the robot (see Fig. 4.13). Two auxiliary frames are defined for obtaining the transformation ${}^{\mathrm{T}}\mathbf{T}_{\mathrm{G}}$: the frame K that is placed at the foot in contact with the ground, and the frame Q that is fixed to the other foot. The ground contact is measured by the force sensitive resistor (FSR) sensors located in the foot sole. In case when both feet are in contact the frame K is placed at the right foot. Therefore, ${}^{\mathrm{T}}\mathbf{T}_{\mathrm{G}}(\mathbf{q})$ depends on the current body posture $\mathbf{q}$, it is obtained from ${}^{\mathrm{K}}\mathbf{T}_{\mathrm{Q}}(\mathbf{q})$, such that

$${}^{\mathrm{T}}\mathbf{T}_{\mathrm{G}} = \begin{bmatrix} \mathbf{R}_{\frac{\Phi}{2}} & [\mathbf{p}_{\frac{\mathrm{x}}{2}}\ \mathbf{p}_{\frac{\mathrm{y}}{2}}\ 0]^{\mathrm{t}} \\ 0 & 1 \end{bmatrix}, \tag{4.33}$$

where $\mathbf{R}$ and $\mathbf{p}$ are the rotation and position component of ${}^{\mathrm{K}}\mathbf{T}_{\mathrm{Q}}(\mathbf{q})$. That is, $\mathbf{R}_{\frac{\Phi}{2}} = \mathrm{Rot}(\vec{\mathrm{z}}, \frac{\Phi}{2})$ considers half of the rotation along the z-axis, whereas $\mathbf{p}_{\frac{\mathrm{x}}{2}}$ and $\mathbf{p}_{\frac{\mathrm{y}}{2}}$ denote half of the translation along the x- and the y-axis respectively.

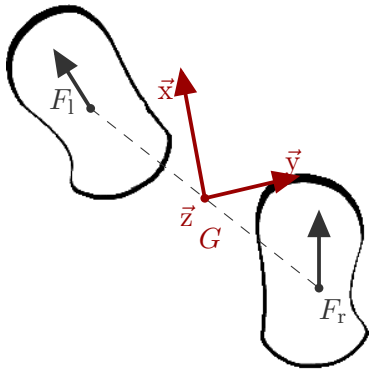


Figure 4.13 – Definition of the ground frame. The frame G is placed at center of the projection of the feet in the ground. The z-axis is taken perpendicular to the ground, the x-axis is the mean direction obtained from the projection of the orientation F_l of the left foot and F_r of the right foot on the motion plane.

The frame $Torso_g$

The frame T_g is placed at the center of the torso, so the z-axis direction is aligned with the gravity vector direction. The orientation components are obtained from the *inertial measurement unit* (IMU) also located at the torso. The transformation ${}^{T}\mathbf{T}_{T_g}$ is defined such that

$$ {}^{T}\mathbf{T}_{T_g} = \text{Rot}(\vec{y}, \gamma)\text{Rot}(\vec{x}, \eta), \tag{4.34} $$

where γ and η correspond respectively to the azimuth and the direction inclination of the torso with respect to the gravity vector.

The frame Eye_g

The idea of this frame is to provide an eye-centered reference aligned with the direction of the gravity vector. For this, the placement of frame E_g coincides with the placement of the camera frame C. Therefore, in the observation of the position of the object the frames for measurement and representation (Freksa and Mark [132], see Sec. 2.4.2) are the same. The orientation components are obtained from the IMU (similarly to T_g,). Thus, ${}^{T}\mathbf{T}_{E_g}$ is defined by

$$ {}^{T}\mathbf{T}_{E_g} = {}^{T}\mathbf{T}_{C} \begin{bmatrix} \mathbf{R}_{\gamma\beta} & 0 \\ 0 & 1 \end{bmatrix}, \tag{4.35} $$

where $\mathbf{R}_{\gamma\beta} = \text{Rot}(\vec{x}, \gamma)\text{Rot}(\vec{y}, \beta)$, with γ and β the rotations around the x- and the y-axis obtained from the transformation ${}^{C}\mathbf{T}_{T} = ({}^{T}\mathbf{T}_{T_g}{}^{T_g}\mathbf{T}_{C})^{-1}$.

The frames Neck and Eye

The orientation of the z-axis for the frames N and E are taken according to the instantaneous posture of the robot. Therefore, frame N is placed at the location of frame 15 (see Tab. 4.2 and Fig. 4.12), so ${}^{T}\mathbf{T}_{N} = {}^{T}\mathbf{T}_{14}{}^{14}\mathbf{T}_{15}$. The frame E actually corresponds to the camera frame C. The new notation is introduced simply to improve readability.

In the definition of E the observation of the object's pose is done such that the frame for measurement and representation are the same.

Experiments

Three experiments were designed under the same scenario considered for the previous study (see. Fig. 4.8). In the first experiment each location B is evaluated at ten distinct initial conditions (some cases are illustrated in Fig. 4.14). In order to avoid simulation bias, each trial is repeated 3 times, so the total number of trials are $10 \times 5 \times 3 = 150$. Since the walk and the Look-at task run simultaneously, the joint positions $\mathbf{q}$ are stored during the image capture time, such that the computations for the localization are based on proprioceptive data related to the image under analysis. The ground truth is obtained from Webots by attaching sensors to the body of the robot (i.e. the type *GPS* for the position and *Compass* for the orientation).

The dependent variables under study are summarized in Tab. 4.3. E_{tim} is the total time in seconds required for convergence. T_{eff} is the ratio between the initial distance to the desired location and the total linear displacement of the agent. L_{pre} is the absolute norm between the final pose of the robot in relation to the object and the pose taught by demonstration. It is a scale-less measurement obtained by adding the angular and the linear differences. P_{pre} is the absolute norm in radians between the demonstrated joint positions (i.e., the body posture) and the final joint positions.

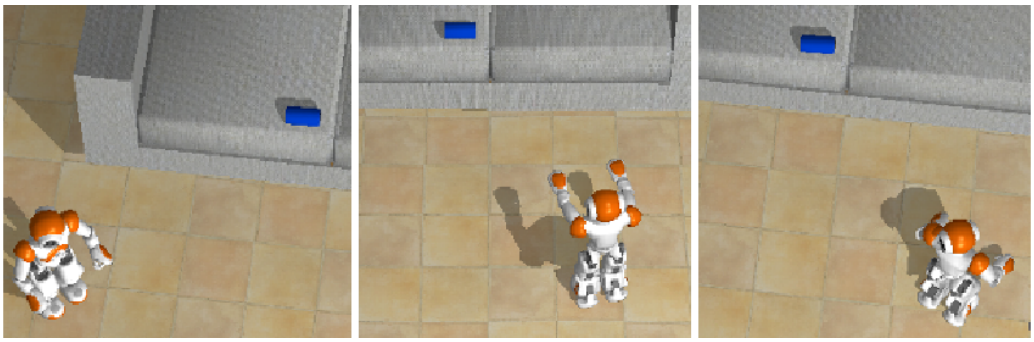


Figure 4.14 – Examples of initial conditions for the frame placement study. The robot must approach the blue object over the sofa as taught by demonstration.

Var	Expression	Description
E_{tim}	$\int_0^\tau dt$	Experiment time in s.
T_{eff}	$\frac{\rho_0}{\left(\int_0^\tau m_\rho dt\right)}$	Trajectory efficiency, with ρ_0 the initial distance and m_ρ the ground truth displacements.
L_{pre}	$\lvert\zeta^\tau - \zeta^*\rvert$	2D pose precision, with ζ^τ the final pose and ζ^* the desired pose.
P_{pre}	$\lvert\mathbf{q}^\tau - \mathbf{q}^*\rvert$	Body posture precision in rad, with $\mathbf{q}^\tau$ the final joint values and $\mathbf{q}^*$ the desired values.

Table 4.3 – Dependent variables under study.

Motion in the walk primitive is expressed with respect to the reference frame G. Thus, motion represented in other reference frames has to be accordingly converted. Let

the 6D pose vector $\zeta = [\xi\ \vartheta]^{\mathrm{t}} = [[X\ Y\ Z]\ [\gamma\ \beta\ \phi]]^{\mathrm{t}}$ express the 3D position ξ and the orientation ϑ of a body in space, a 3D rotation matrix be denoted by $\mathbf{R}$, and the position of frame G be denoted by $\mathbf{g}$. The differential of motion ${}^{\mathrm{B}}\Delta\mathbf{M}$ can be expressed with respect to frame G, such that

$$ {}^{G}\Delta M = \begin{bmatrix} {}^{G}\mathbf{R}_{B}({}^{B}\vartheta \times {}^{B}\mathbf{g} + {}^{B}\xi) \\ {}^{G}\mathbf{R}_{B}{}^{B}\vartheta \end{bmatrix}, \tag{4.36} $$

where ${}^{B}\vartheta = [0\ 0\ \phi]^{\mathrm{t}}$ is the rotation around the z-axis (i.e. the change on the angular motion direction of the robot). Notice that this definition imposes a spatial constraint to frame B, so the z-axis must be taken parallel to the z-axis of G. As it happens with some of the placements evaluated this constraint is not ensured, therefore the other components of the orientation are ignored (i.e. a less effective regulation of the robot heading is obtained). The motion correction defined in Eq. (4.30) can be expressed with respect to G as follows

$$ {}^{G}\overline{\mathbf{m}} = [{}^{G}\Delta M_{\mathrm{X}}\ {}^{G}\Delta M_{\mathrm{Y}}\ {}^{G}\Delta M_{\phi}]^{\mathrm{t}}. \tag{4.37} $$

A second experiment considered a new criteria for the regulation of the angular motion. Based on the heuristic assumption that the z-axis of frame 16 (the neck yaw, see Tab. 4.2 and Fig. 4.12) is approximately aligned with the z-axis of frame G, the idea is to verify whether the body configuration of the robot and its walk style would allow to obtain a correction on the angular motion, by the regulation of the yaw posture α of the neck to the desired state α^{*}, learned by demonstration (i.e. the robot is put in front of the object so the Look-at task centers it on the field of view, and the posture of neck is registered). In other words, the robot changes the orientation of the walk to reach the desired posture of the neck. Thus, the error $\mathbf{e}_1$ in Eq. (4.16) is redefined such that

$$ \mathbf{e}_1 \approx \begin{bmatrix} \mathbf{e}_{1\rho} \\ \mathbf{e}_{1\theta} \\ \alpha - \alpha^{*} \end{bmatrix}. \tag{4.38} $$

A third experiment is designed in order to study the effect of proprioceptive uncertainties over the walk trajectories. For this, Gaussian noise ($\mu = 0$, $\sigma = 2\,\mathrm{deg}$) is added to the joint measurements $\mathbf{q}$ when computing the localization.

Results

The results of the first experiment are illustrated in Fig. 4.15. The final precision L_{pre} was slightly superior for E and E_{g}. This is due to the additional contribution of the Look-at task to the regulation of the localization error. That is, since these placements are centered on the sensor, they are constantly redirected toward the stimulus, which results in a decrease of the localization error. In relation to the trajectory efficiencies the results seemed to be related to the proximity to frame G. In this sense, the closer to G the more efficient appeared to be the trajectories. This may be due to the sway motions as pointed out by Dune et al. [60], which is the contribution to the signal error from the oscillatory evolution of the torso. The placement G also presented more consistent results in the time consumption E_{tim} (lowest standard deviation).

The previous comparisons showed relatively subtle differences among the candidates, but the evaluation of the posture precision revealed that E and E_{g} were significantly

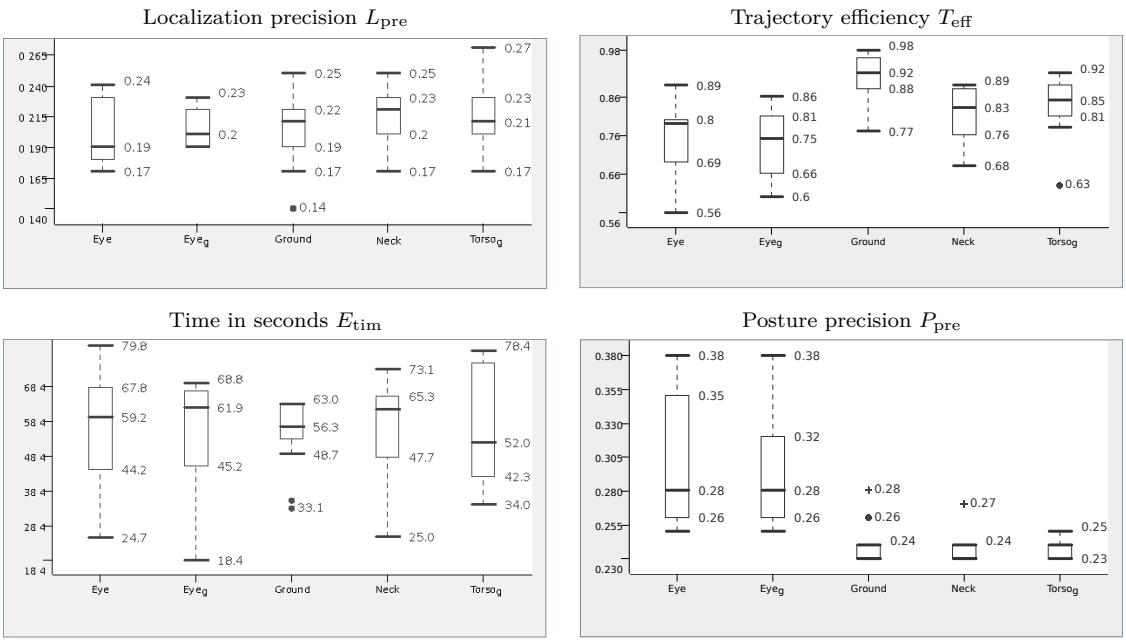


Figure 4.15 – Box plots for experiment 1. In the plots the bottom and top box correspond to the first and third quartiles, the band inside the box is the second quartile (the median), the ends of the whiskers represent the range of the data. Outliers are shown with dots and crosses. In the first row from left to right the results for L_{pre} and T_{eff}, these variables are unit-less. Likewise, in the bottom row the results for E_{tim} in seconds and P_{pre} in radians.

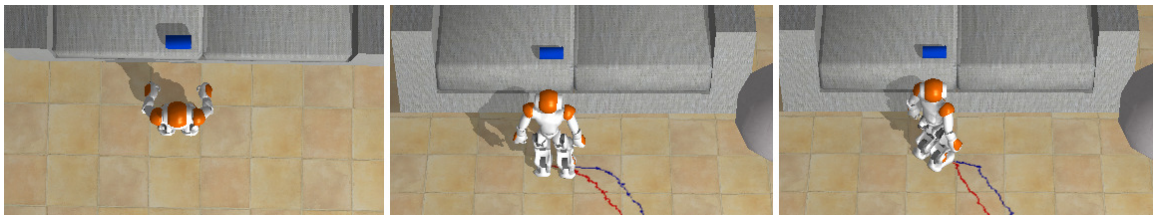


Figure 4.16 – Body posture comparison between E_{g} and G. On the left the demonstrated posture. At the center the convergence obtained for frame G. On the right the convergence obtained for frame E_{g}. In both cases the agent converged to the correct position but the body posture for E_{g} differed.

less adequate choices. As illustrated in Fig. 4.16, the final posture differed significantly from the demonstration provided. The differences can be clearly appreciated in Figs. 4.17 and 4.18, which compare the evolution of the position of the stimulus in the ego-space for distinct choices of the base frame. It is noticeable that eye-centered placements produced almost straight-line paths towards the desired location, given the fact that the head posture in relation to the body is constantly altered by the Look-at task (see Fig. 4.19 for a comparison on the evolution of the neck yaw). Thus, the context of the body posture is lost, which would be equivalent to consider a flying (or dismembered) camera.

The results for the second experiment are presented in Tab. 4.4, where a comparison between the performance of the original and the heuristic version of the Walk task is given. Rows with apparent improvements are highlighted in blue, whereas deterioration is marked in red. In relation to G the new scheme had a positive effect by maintaining the results for P_{pre} and improving the other measurements. The choice T_{g} was penalized in T_{eff} evidencing larger trajectories, contrarily, E_{tim} tended to be shorter. One plausible

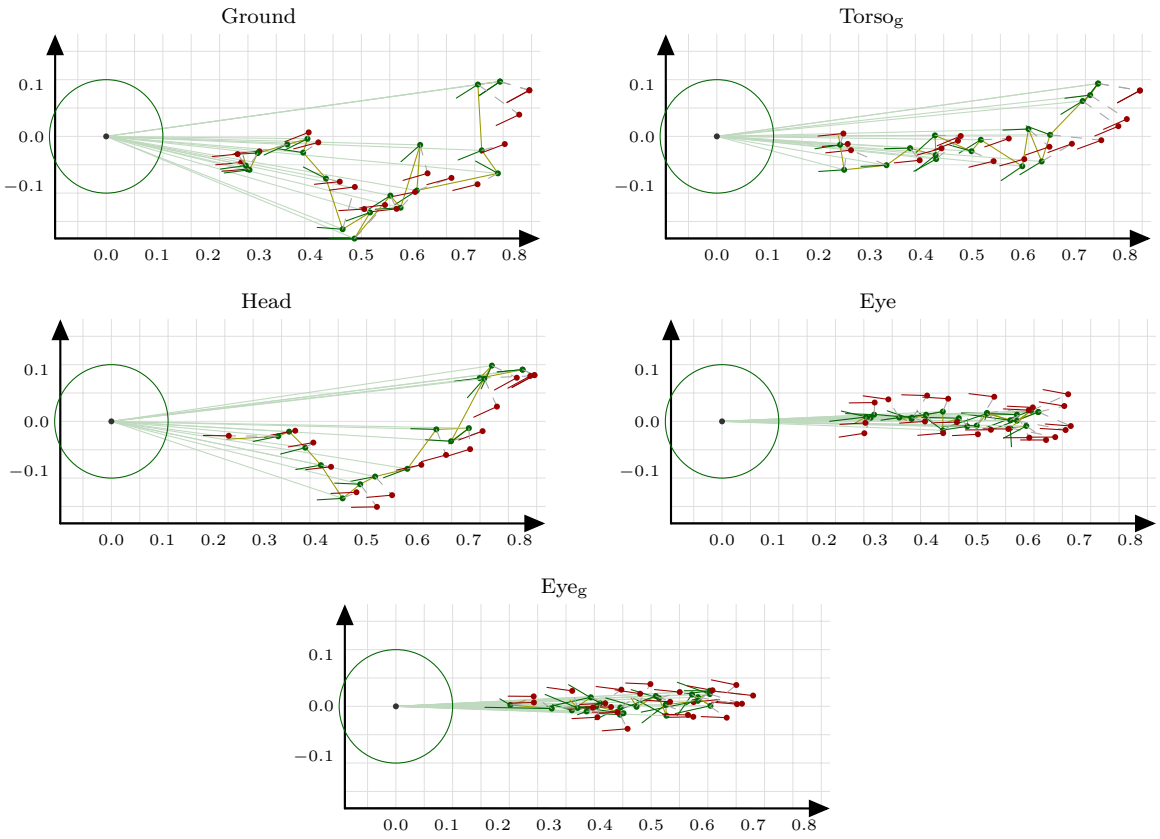


Figure 4.17 – Top view of egocentric localization. The circumference represents the ego-cylinder. In red the ground truth, in green the estimations. Distances are expressed in m.

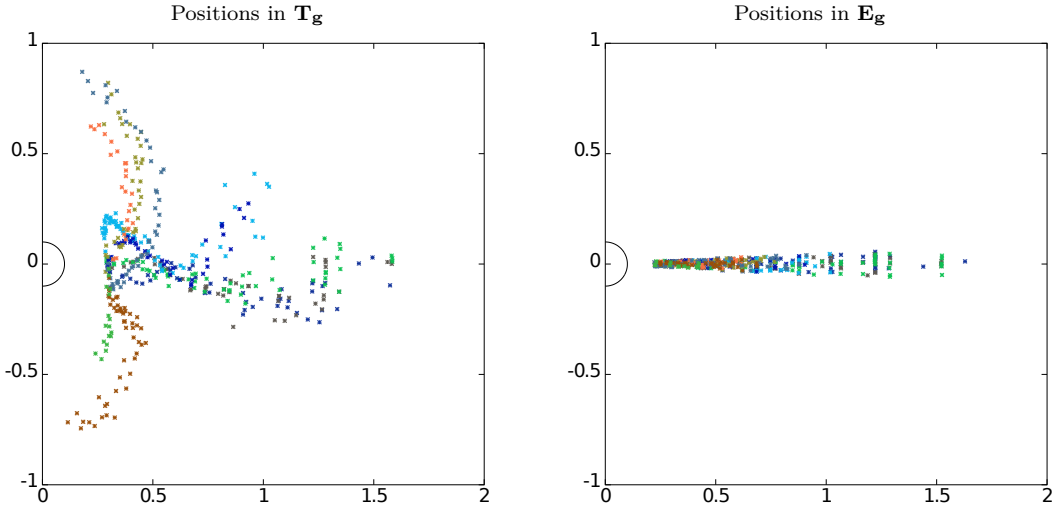


Figure 4.18 – Top view of object positions for T_g and E_g. The plots show the 10 initial conditions evaluated. Each trial is assigned a distinctive color. Distances are expressed in m.

explanation for this effect is that the velocity achievable by the robot in the saggital plane is higher than in the frontal plane, thus the robot may have walked more in the saggital direction. The placement N presented improvements in P_{pre} and L_{pre}, but slightly larger times T_{eff}. In relation to E and E_g, there is a general tendency of improvements. It is noticeable that the final precision of the posture P_{pre} for the eye-centered cases is now comparable to the performance obtained with other placements. Thus, the heuristic

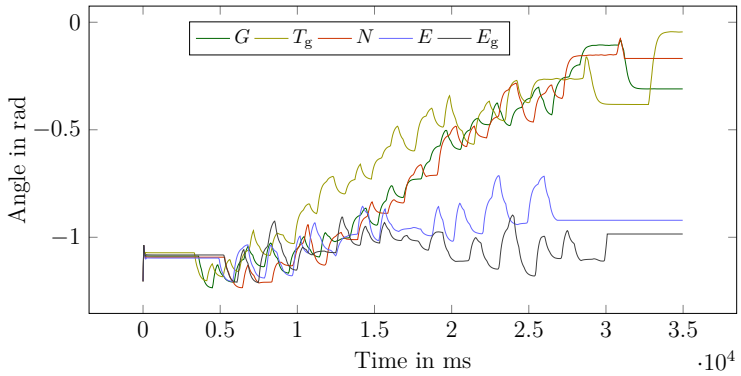


Figure 4.19 – Comparison on the evolution of the neck yaw for the test case 3. E and E_g exhibited faster convergence but the body posture differed largely from the others.

modification appeared to exert a positive or neutral effect over the agent's behavior.

B	Var	μ_1	μ_2	σ_1	σ_2
G	L_{pre}	0.20	0.19	0.03	0.02
	P_{pre}	0.24	0.24	0.02	0.02
	T_{eff}	0.90	0.92	0.07	0.04
	E_{tim}	57.73	53.32	13.31	6.83
T	L_{pre}	0.22	0.21	0.03	0.03
	P_{pre}	0.24	0.23	0.01	3.9e-3
	T_{eff}	0.84	0.81	0.06	0.05
	E_{tim}	56.14	53.17	16.05	9.65
N	L_{pre}	0.22	0.20	0.02	0.03
	P_{pre}	0.24	0.23	0.01	2.2e-3
	T_{eff}	0.82	0.82	0.07	0.06
	E_{tim}	55.55	56.97	12.28	10.51
E	L_{pre}	0.21	0.23	0.04	0.03
	P_{pre}	0.30	0.24	0.05	0.01
	T_{eff}	0.75	0.77	0.08	0.07
	E_{tim}	57.57	53.65	15.14	10.91
E_g	L_{pre}	0.21	0.20	0.02	0.02
	P_{pre}	0.29	0.23	0.04	4.5e-3
	T_{eff}	0.73	0.78	0.09	0.06
	E_{tim}	57.58	58.90	13.13	11.28

Table 4.4 – Comparison on the performance of the original version ($i = 1$) and the heuristic version ($i = 2$) of the Walk task (see Tab. 4.3). The mean μ_i and the standard deviation σ_i are given.

The results obtained for the third experiment, where noise is added to the joint measurements, are illustrated in Fig. 4.20. The performance of eye-centered placements appeared to be less affected by noise and the relative advantages of body-centered placements observed in the first experiment were practically leveled. Figure 4.21 presents a comparison on the evolution of the localization signals for frame G during the task. It is noticeable that noise hampered the convergence so the agent invested more efforts to position in front of the object.

Discussion

Given the lack of consensus in the literature about the placement for the ego-sensory structure, this study has investigated five possibilities: three body-centered and two eye-

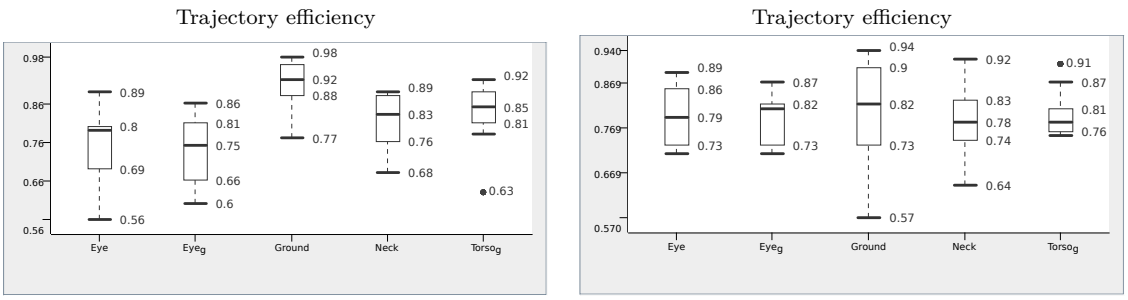


Figure 4.20 – Comparison on T_{eff} for experiments 1 (left) and 3 (right). In the experiment 3 the angular motion of the agent was obtained by the regulation of the yaw position of the neck, and noise was added to the proprioceptive measurements.

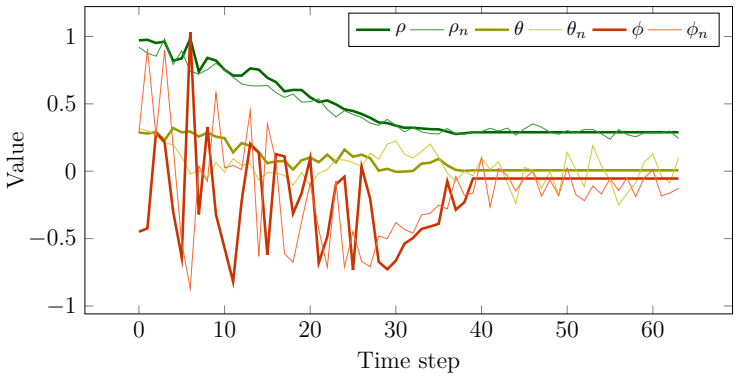


Figure 4.21 – Localization error for frame G with Gaussian noise ($\mu = 0$, $\sigma = 2\,\text{deg}$) added to proprioceptive measurements. Thicker lines correspond to the noise-free condition.

centered locations. The results of the first experiment suggested that from the characteristics of the robot (i.e. the fact that the motion primitive operates in frame G), the more adequate placement for the base frame is G. Though, convergence was obtained for all body-centered choices. It is noticeable that the placement N did not conform to the constraint imposed by the localization model. Thus, the fact that the agent walked on a plane in vertical posture constrained the mobility of the reference system N. Thereby, close results can be obtained for G and N without using the IMU of the robot.

Eye-centered placements did not preserve the context of the body posture during the task (since they were not necessarily aligned to the sagittal plane of the agent, see Fig 2.12). However, as the second experiment has showed, in case heuristic knowledge is employed, such that a body posture regulation task is enforced; a hybrid solution can be obtained where the correction in the position is determined eye-centered but the regulation on the angular motion is calculated body-centered. This combination produced interesting results and was the less affected by the noisy condition of the third experiment, where the more intermediate joints between the measure frame (i.e. the sensor frame) and the task representation frame, the worse the result obtained. The heuristic solution is also computationally more efficient since the operations for the frame transformations between the camera and the base frame are no longer required.

CS-III: Approaching a real object

From the results obtained in simulation, the objective of this study is to verify whether a real task with the robot Nao can be accomplished.

Experiments

The scene considered was indoor (i.e. the robotics lab), closed to an unstructured environment, under uneven illumination (natural and artificial light sources), under constant influx of collaborators in the facilities. The object of interest is a yellow card clearly distinguishable in the scene. In case the segmentation provides more than one salient area, the biggest blob is selected for processing the localization. The desired pose in relation to the object is shown to the robot by pressing the head tactile sensor. Two experiments are designed. In the first one the robot is moved away from the desired configuration and 10 trials are repeated from different initial configurations. In view of the processing limitations of the robot, the images are captured on-board and transferred to the remote station for processing. Likewise, motion commands are sent to the robot trough the wireless link. Given that the performance and accuracy of the motion primitives are affected by several factors (e.g. the heat of the motors, the accumulation of errors in the software platform, sensor inaccuracies, and sliding, among others), the base frame was placed in location G since it seemed to produce more accurate motion. A move-then-stop policy for the walk is employed, where the robot stops completely before processing additional commands. Though, the Look-at task runs continuously. The tolerance for convergence was the same as detailed in Sec. 4.6.2. In the second experiment disturbance is included so the robot is moved to another location while approaching the object (i.e. the robot kidnapped problem as described in Sec. 2.4.1).

Results

Figure 4.22 shows the trajectory followed by the robot in two distinct trials, and Fig. 4.22 presents some on-board views and the corresponding segmentations during the approach. The experiment was successfully accomplished all the times so the robot could autonomously return to the front of the yellow card as learned by demonstration. Though, as it can be noticed in the frames, the robot walked much of the trajectory in the frontal plane direction. In relation to the second experiment the results confirmed those obtained in simulations, thus, it was observed that despite the disturbances applied, whenever the object remained within the field of view, the robot was able to approach it.

Discussion

The objective of this study was to assess whether a real task could be accomplished under the modeling approach adopted. The results of the experiments have shown that from the controlled conditions described (i.e., a move-then-stop policy and a clearly salient object), the robot was able to autonomously accomplish the task. Nevertheless, the trajectories obtained were not esthetically appropriate, and perhaps inefficient since the robot moves faster in the saggital plane than in the frontal plane. Also, the fact of disposing an object clearly distinguishable from the rest of the scene is a bit far from natural situations. In fact, as shown in Fig. 4.24, changing the stimulus by a more

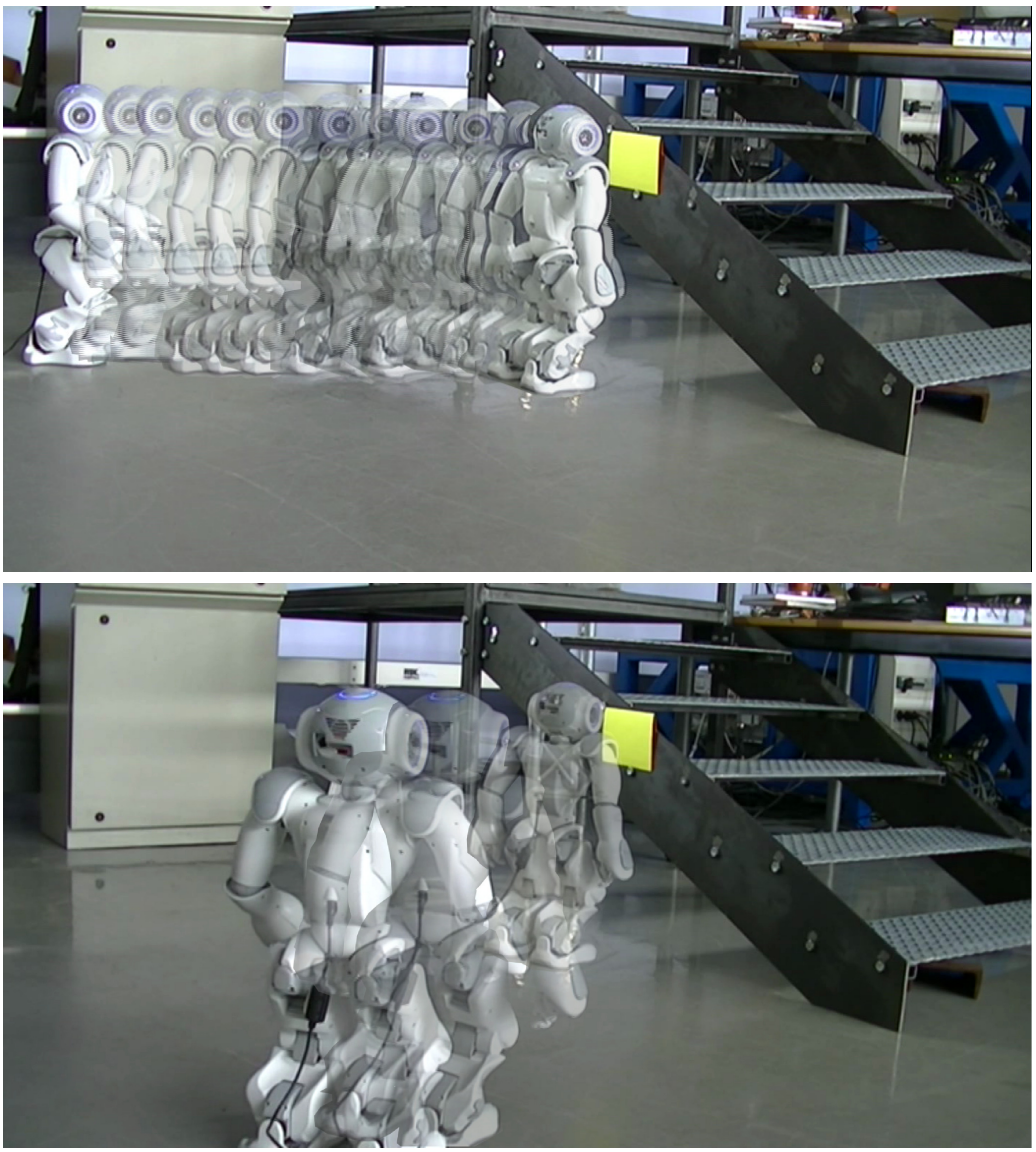


Figure 4.22 – Different trials of the yellow card approach experiment.

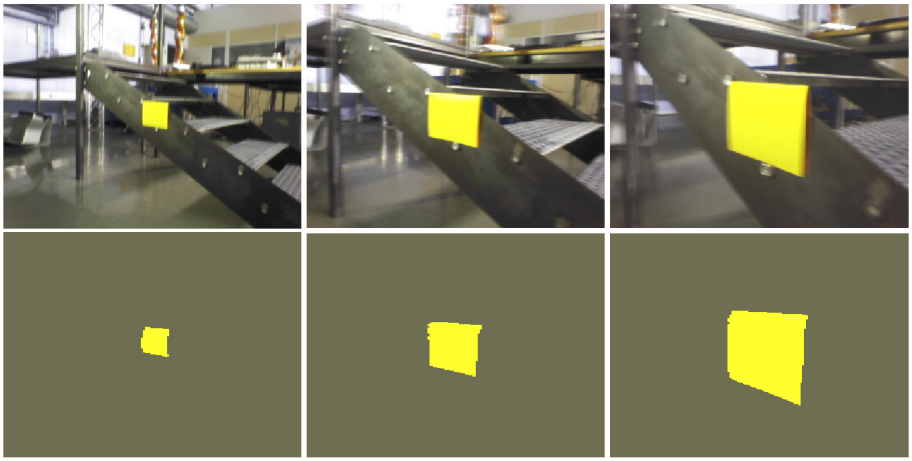


Figure 4.23 – On-board view and segmentations of the yellow card experiment.

complex object would invalidate the approach, since a more adequate filtering mechanism for selecting among the emerged blobs would be requited.

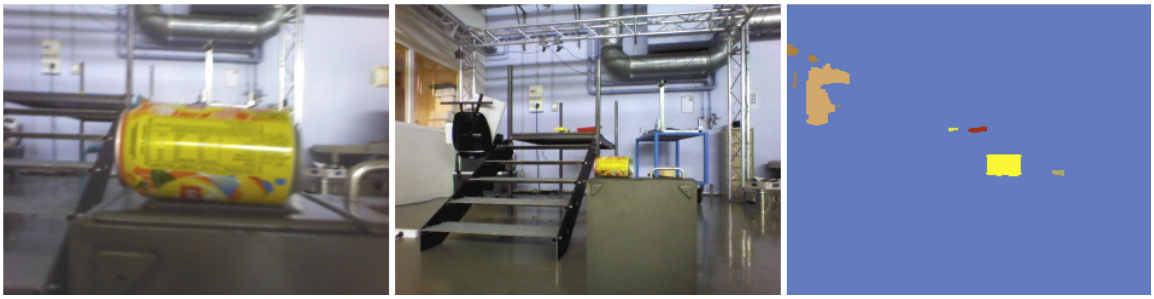


Figure 4.24 – Multi-color saliency. On the left a closeup of the object of interest: a colored can with different tonalities of yellow and orange. In the center the scene registered on-board. On the right different regions were salient due to the diversity of the color model.

Conclusions

This chapter has started by reviewing related studies on humanoid locomotion. Several approaches were found, thus the discussion focused on works that considered the on-board capture of visual data. A common aspect noticed is the use of feature tracking relying on the principle of Verification Vision, for processing localization. In this work a different approach was taken, by considering a lower acquisition rate. From the results reported in the study of Sec. 3.4.3, the region-based whole segmentation technique was employed for obtaining a top-down binarization of the scene. So the monocular vision mapping from the salient region on the image to the 3D space relied on a rough 3D model encompassing the object.

A behavior scheme based on visual processing was designed. For this, the modeling of the task followed the literature of visual servoing. The PBVS and IBVS control approaches were employed simultaneously, in order to maintain the object of interest in the field of view, and to steer the robot to the desired 2D pose in relation to the object. The solution considered the motion primitives of walking and directing the head. The results corroborated the plausibility of the model for the approaching task, though a fairly simple situation was considered where a single object was salient, and there were no obstacles between the robot and the object. Furthermore, the robot walked much of the trajectory in the frontal plane direction, which was not aesthetic and efficient. In Chapter 5 a control law that mimics human motion, and the problem of reliable approaching to objects under saliency ambiguity are investigated, so the agent has to do robust visual selection to discriminate and localize the object.

Other important topic treated was the study of some effects of embodiment over the perception of the stimuli. Given the lack of consensus in the literature about the placement for the ego-sensory structure, body- and eye-centered locations were investigated. The results suggested that, from the body posture that the agent adopts when walking on a plane, convergence was obtained for body-centered choices, even when the placement did not conform to the constraints imposed by the localization model. Thus, the fact that the agent walked in vertical posture restricted the mobility of the reference system. These results suggest that embodiment can be exploited to obtain an efficient solution in terms of resource consumption (e.g. the IMU was not required for G and N).

Eye-centered placements did not preserve the context of the body posture during the task, but in case heuristic knowledge is used, such that a body posture regulation task is enforced; a hybrid solution can be obtained where the correction in the position is deter-

mined eye-centered (i.e. the measurement and the representation frames are the same) but the regulation on the angular motion is body-centered. This combination produced interesting results in simulations and was less affected by noise on the proprioceptive acquisitions. It was also computationally more efficient. However, with the real robot, the placement G (which is the same reference for the walk primitive) produced more accurate motion, so the base frame was taken at the placement G. The hybrid location is considered in Chapter 6 where action-oriented representations of the stimulus are studied in simulations.

5

Embodied perception

Contents

5.1 Introduction 93
5.2 Grounding vision-based locomotion 94
5.3 EC-based task analysis 95
5.3.1 Mimicking human walking style 95
5.3.2 Resources available 97
5.3.3 Modeling a behavior scheme 98
5.4 Behavior autonomy 102
5.4.1 Embodied features 103
5.4.2 Bayesian network for information fusion 105
5.5 Case studies 107
5.5.1 Materials and resources 107
5.5.2 Behavior scheme implementation 108
5.5.3 Hybrid architecture implementation 109
5.5.4 CS-I: Model parameters estimation 110
5.5.5 CS-II: Simulation of object redundancy 113
5.5.6 CS-III: Approaching a real can 117
5.6 Designing reliable approach tasks in six-steps 121
5.7 Conclusions 123

Introduction

The automation of visually-guided walk has arguably adopted in its infancy the so called cognitivist approach to artificial intelligence (AI), which, under the Cartesian dualist influence, has tended to look at physical and mental processes as belonging to different realms. Significant progress has been obtained from this view, though the performance is still distant from the sophistication observed in natural behavior. Among the several

challenges reported in the literature, one is undoubtedly to achieve reliable perception from noisy data. Since the sensory input goes through a process of symbolization, and cognition would involve - under this view - computations over symbols, the physical context at which the latter emerged is no longer available to the cognitive process. In other words, in abstracting cognition from the context, information is inevitably lost.

To cope with the difficulties of perceiving the object while moving, several frameworks that flourished in the machine learning research have been employed (e.g. Markovian models, support vector machines, among others). These attempts have produced impressive results, although, by keeping intact the fundamental premise of decoupling between bodily and mental processes, they have relied on expensive resources in the form of disembodied explicit models, knowledge databases, and intensive computation. Thus, the processing bottleneck has impacted the autonomy and the reactivity of the agent. As it was discussed in the previous chapter, extraneous variables (i.e. un-modeled phenomena) have been controlled by adapting the scene to the task, which has compromised the generality of the solution.

This chapter focuses on the limitations pointed out by the studies developed in Sec. 4.6.4. That is, a) a fairly simple situation where a single object was salient, so no attention selection was required, and b) the fact that the robot walked in the frontal place direction much of the trajectory. For this, a more realistic solution to the approach task is proposed by redefining the behavior scheme according to the EC research perspective. Thus, from a first-person perspective analysis of the sources of information available, the agent is given a non-holonomic human-like walking style. In order to ensure robustness and reliability in the task, the behavior scheme is integrated to a hybrid architecture in charge of monitoring the execution. This functionality is obtained from the design of a Bayesian network in charge of information fusion. Moreover, the network grounds the attention selection mechanism developed for perceiving the object. These aspects are analyzed and integrated to a methodological proposal which synthesizes the development of robust humanoid approach tasks in six steps.

Grounding vision-based locomotion

The term *embodied cognition* reunites co-existing research interests with diverse subject matters. A thorough review on the conflicting views in EC is beyond the scope of this work (the reader is referred to the works of Shapiro [165] and Wilson [188] for a discussion on this topic). Thus, this study is in agreement with Anderson [6] when he identifies in the *physical grounding hypothesis* (Brooks [29]) the distinctive aspect of EC as opposed to a situated but cognitivist view of embodiment. Accordingly, behavior is studied as a complex system, where knowledge representation is thought to be grounded in the physical interaction.

By considering the emergent aspects of behavior, the research in EC has been experiencing a growing boom (Pfeifer & Pitti [143], Hoffmann & Pfeifer [84]). Initial works focused on the body morphology, particularly, in the aspect of energy consumption, robustness, and computation offloading. These studies showed how morphological computation and passive dynamics can significantly reduce the need for control and modeling, decentralize computations, and dissipate disturbances from the environment. Given the success of these experiments, the research has gradually evolved as to include the sensorymotor coupling, and higher cognitive processes such that perception and learning.

However, the study of enactment as reported by Vernon [182], poses significant challenges to robotists. From the phylogenetic point of view, the autonomous development of the cognitive system would also require the hosting platform to evolve. The acquisition of several human sensory-motor skills, for instance, is accomplished only once the body has either matured or adapted to new conditions (e.g. by increasing muscle strength). This capacity of self-modification is not easy to obtain in artificial bodies. From the ontogenetic point of view, the difficulty is to design structures for efficiently integrating information from different sensory modalities, that would enable learning diverse tasks and generating knowledge from previous experiences. Moreover, in the context of service robotics applications, perhaps the most restrictive aspect of enactment is the fact that knowledge acquisition is constrained by coupling, so it is a slow process analogous to natural learning.

Restrictions imposed by the study of enactment hinder at present an exclusive use of this methodology for service robotics. In the scenario envisaged, the agent may be required to deliberate a plan, or to extend learned skills to objects seen for the first time. In order to provide solutions to such requirements, this work adopts an intermediate perspective between the cognitivist and the EC methodology. Hence, a dynamic first-person description of the studied behavior is performed to rigorously restrict modeling. However, since flexible solutions are desired, the possibility of counting on a rough model of the object and the fact that the initial perception of the object is based on action-independent knowledge are tolerated. Thus, the aspect of *grounding* or instantiation of action-independent knowledge, and how contextual representations can contribute to perceive the object; are of central importance for this research. In this sense, this works is in agreement with Clark's assertion that the radical opposition of EC to cognitivism *invites competition where progress demands cooperation* (Clark [47]).

EC-based task analysis

According to Wilson & Golonka [187] the study of embodied cognition involves four essential steps: a) a dynamic analysis and description of the task, where b) a set of resources from the *body*, the *brain*, and the *environment* are identified, c) a research hypothesis on how these resources may contribute to the solution is formulated, and d) experimental evaluation is conducted in order to confirm that the agent is able to accomplish the task. This methodology is adopted in this work with the particularity that the interest is focused on behavior automation, and not on understanding the natural being. Next, the first three methodological steps are described, whereas the forth one is treated in the case studies of Secs. 5.5.4 and 5.5.5.

Mimicking human walking style

The dynamic description of behavior is a widely known methodology in the automation research, that is rooted in the study of complex systems in the fields of physics and mathematics. According to Thelen & Smith [175] human development can be viewed as a complex, far-from-equilibrium, open system. The degrees of freedom of such system are very large. Though, the interactions between the system components would produce patterns that emerge as behavior in the environment with a spatial and temporal order, which can be mathematically described. Thus, the state-space description of behavior

is an abstraction used for studying human development by reducing its dimensionality. The idea behind is that configurations emerging from collective interaction of individual elements increase until dominating the behavior of the system, and can be described by *order parameters* (i.e., the state variables). Regions in the state-space that function as *attractors* to the state variables characterize *behavioral modes*. An important goal of the dynamic analysis is to identify *control parameters*, which are endogenous or exogenous variables that assemble the state-space in a given attractor regime.

The dynamics of steering and obstacle avoidance has been investigated by Fajen & Warren [65]. Subjects were invited to participate in an experiment where they walked freely carrying a head-mounted display. They were instructed to walk toward the goal while dodging obstacles. The trajectories followed were recorded, and compared to a first- and second-order description of the behavior. In the experiment, the state-space included the heading direction and velocity of the subjects with respect to a fixed reference. The control parameters were perceptually available including the goal's and obstacles' bearing and distance.

Unfortunately, in the Fajen & Warren study bilateral symetric stimuli were used, so both the obstacles and the goal were conceptually represented as points on the plane. Normally, people tend to approach the front of a coffee machine or a drawer, so to facilitate the manipulability. For cases like those it is important to consider a particular perspective of the object. Furthermore, objects may be located at different heights to which the agent must direct the gaze. In this more complex scenario, as detailed in the previous chapter, the localization ${}^{B}\zeta$ of the object is represented according to Eq. (4.15) (see Fig. 4.2).

The egocentric first-order description of the approach *human-mimic walk* (HMW) is proposed in this work. HMW defines the motion of the agent from the observed localization error ${}^{B}\hat{\mathbf{e}}$ (see Eq. (4.16)). In order to take into account the aesthetics of motion, human walk is mimicked. That is, non-holonomic motion is used when human is far from the object, but holonomic motion is preferred when human is close enough to the goal. Let $\dot{\mathbf{h}}$ and $\dot{\mathbf{n}}$ denote respectivelly the holonomic and non-holonomic evolution of the walk, a sigmoid transition $0 \leq \lambda \leq 1$ between the motion policies can be established depending on the distance ${}^{B}\hat{\mathbf{e}}_{\rho}$, such that

$$\text{HMW} : \lambda\dot{\mathbf{h}} + (1-\lambda)\dot{\mathbf{n}}. \tag{5.1}$$

where $\lambda = 1/(1+\exp(s_1(\hat{\mathbf{e}}_\rho - s_2)))$. The notation for ${}^{B}\hat{\mathbf{e}}$ in simplified to $\hat{\mathbf{e}}$ to improve the readability. The parameter s_1 is a proportional gain, and s_2 is the sensitive distance for the transition.

The holonomic evolution $\dot{\mathbf{h}}$ of the walk is defined so the position components of $\hat{\mathbf{e}}$ are expressed in Cartesian coordinates for convenience (since the walk primitive of Nao uses this coordinate system). Thereby,

$$\dot{\mathbf{h}} = \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \omega \end{bmatrix} = \begin{bmatrix} k_1 \cos(\hat{\mathbf{e}}_\theta)\hat{\mathbf{e}}_\rho \\ k_2 \sin(\hat{\mathbf{e}}_\theta)\hat{\mathbf{e}}_\rho \\ k_3\hat{\mathbf{e}}_\phi \end{bmatrix}. \tag{5.2}$$

$\dot{X}$ and $\dot{Y}$ are the linear velocities, and ω is the angular velocity. Independent corrections along the 3 degrees of freedom are obtained from the proportional gains k_1, k_2, k_3

In non-holonomic motion the correction on the frontal plane and the orientation of the body are coupled (since motion on the y-axis direction is not allowed). As illustrated

in Fig. 5.1, the idea is to induce a rotational motion to reduce both the frontal distance d to the object and the orientation error $\hat{\mathbf{e}}_\phi$. This can be done by estimating d from on-board observations, such that $\hat{d} = \sin(\hat{\mathbf{e}}_\phi - \hat{\mathbf{e}}_\theta)\hat{\mathbf{e}}_\rho$. The desired angular correction ω is

$$\omega = -k_4\hat{d} + k_5\hat{\mathbf{e}}_\phi, \tag{5.3}$$

with k_4 and k_5 denoting proportional gains.

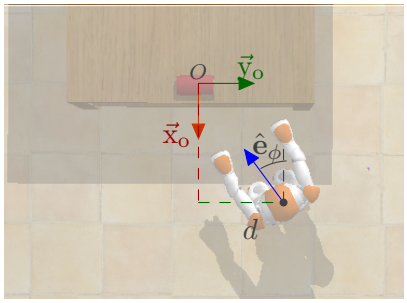


Figure 5.1 – HMW non-holonoic angular motion. The lateral distance with respect to the object is denoted by d. O is the object frame origin. The direction of the saggital plane of the robot is shown in blue.

By substituting $\hat{d}$ in Eq. (5.3) the expression becomes

$$\omega = k_4\sin(\hat{\mathbf{e}}_\theta - \hat{\mathbf{e}}_\phi)\hat{\mathbf{e}}_\rho + k_5\hat{\mathbf{e}}_\phi. \tag{5.4}$$

It is interesting to notice that the contribution of the term at the left of Eq. (5.4), which is related to the frontal correction, can be divided by $\hat{\mathbf{e}}_\rho$ so the correction is proportional to the error in the sagittal plane. Thereby, the non-holonomic evolution of the walk $\dot{\mathbf{n}}$ is define by

$$\dot{\mathbf{n}} = \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \omega \end{bmatrix} = \begin{bmatrix} k_6\hat{\mathbf{e}}_\rho \\ 0 \\ k_4\sin(\hat{\mathbf{e}}_\theta - \hat{\mathbf{e}}_\phi) + k_5\hat{\mathbf{e}}_\phi \end{bmatrix}. \tag{5.5}$$

Motion in the sagittal plane is regulated by the gain k_6.

By combining Eqs. (5.1), (5.2), and (5.5), HMW is defined such that

$$\text{HMW} : \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \omega \end{bmatrix} = \begin{bmatrix} \lambda(k_6\hat{\mathbf{e}}_\rho) + (1-\lambda)(k_1\cos(\hat{\mathbf{e}}_\theta)\hat{\mathbf{e}}_\rho) \\ (1-\lambda)(k_2\sin(\hat{\mathbf{e}}_\theta)\hat{\mathbf{e}}_\rho) \\ \lambda(k_4\sin(\hat{\mathbf{e}}_\theta - \hat{\mathbf{e}}_\phi) + k_5\hat{\mathbf{e}}_\phi) + (1-\lambda)k_3\hat{\mathbf{e}}_\phi \end{bmatrix}. \tag{5.6}$$

The comparison between HMW and the Fajen & Warren description is given in Fig. 5.2. As noticed, with HMW the agent would complete the approach with the body oriented according to the heading direction of the object.

Resources available

As listed in Tab. 5.1, the task solution requires a combination of resources from the *brain*, the *body* and the *environment* (Wilson & Golonka [187]). Short- and long-term

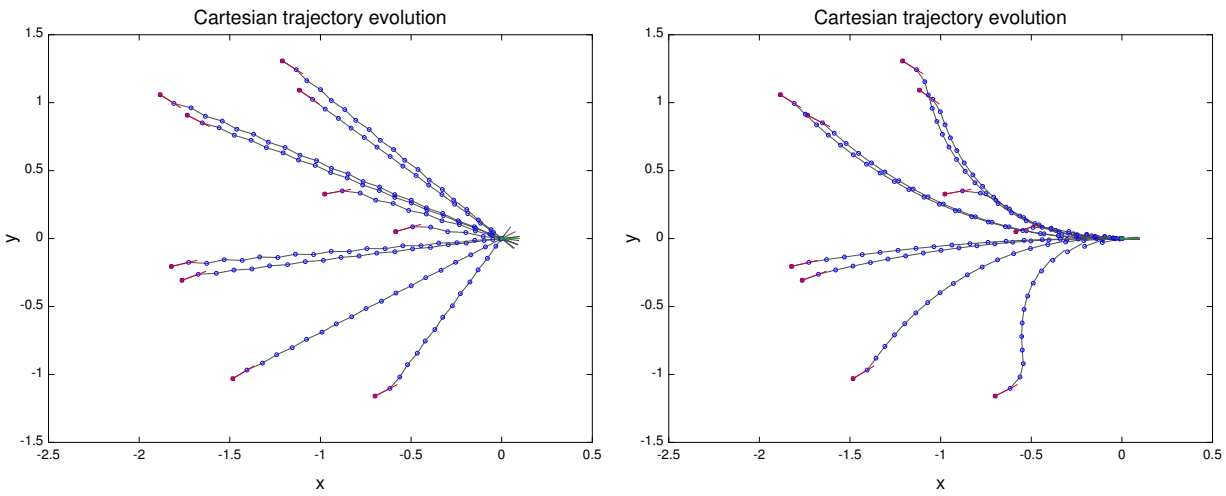


Figure 5.2 – Top view simulation of the Cartesian trajectory followed by the agent. The blue circumferences represent the position and the gray trajectories represent the heading direction. In red the initial configuration, in green the desired configuration. Distances are expressed in meters. On the left the Fajen & Warren description is presented, notice that the final orientation of the body is variable since only the object beating is observed. On the right the HMW proposal is depicted.

memory are required respectively to store information about the actual context and the desired state. The memory contents include endogenous (e.g., proprioceptive) and exogenous (e.g. visual) data, and more elaborated perceptions of spatial relations. The agent employs ego-centric localization relying on a top-down feature attention process. The perceptive system includes a sensory ego-structure for localizing stimuli. The actions in the task are ensured by the skills of walking and head direction. Finally, the environment is assumed to provide a plane surface for motion, where objects are considered to be convex and static.

Type	**Resource**	**Description**
Brain	Memory	Long- and short-term storage of bodily sensations and spatial relations.
	Feature attention	Top-down saliency.
	Localization	Egocentric, relying on a sensory ego-cylinder.
Body	Proprioception	Joint enconders.
	Vision	Color vision.
	Motion primitives	Walking and head direction.
Environment	Plane floor	The agent moves on a plane surface.
	Static scene	Stimuli are fixed, there are no obstacles along the way.
	Convex objects	The object morphology is expected to be convex, with the tracked face distinguishable from the other faces.

Table 5.1 – Resources available to solve the task.

Modeling a behavior scheme

The interest here is in exploring what Clark described as the action-oriented dimension of knowledge, as opposed to the action-independent dimension. Accordingly, the in-

dividual possesses action-independent knowledge, in the form of general properties about the object (e.g., shape, functionality, among others), and action-oriented representations, that include *idiosyncratic, locally effective features to guide behavior* (Ballard 1991, as cited by Clark [47]). Thus, general knowledge about the object category is accompanied by locally-driven bodily sensations that the agent experience in the presence of a particular instance of the object. As colloquially expressed, the task would be equivalent to ask the agent: *to approach the blue drawer at the left.* Notice that such description is valid in the context, and will not be useful to direct the agent toward other sort of drawers.

Behavior emergence

The behavior scheme proposed is inspired by the design of *sub-sumption architectures* (see Brooks [29]). In these architectures behavior is considered to emerge from the conjoint contribution of several independent and self-contained subsystems, where there is no central process in charge of goal coordination. As illustrated in Fig. 5.3, three different sort of tasks are defined: a) motor tasks control the actuation on the environment, b) sensory tasks handle feedback from the body and the environment, and c) localization tasks ensure the coupling between the motor and the sensory tasks. Next, these components are detailed.

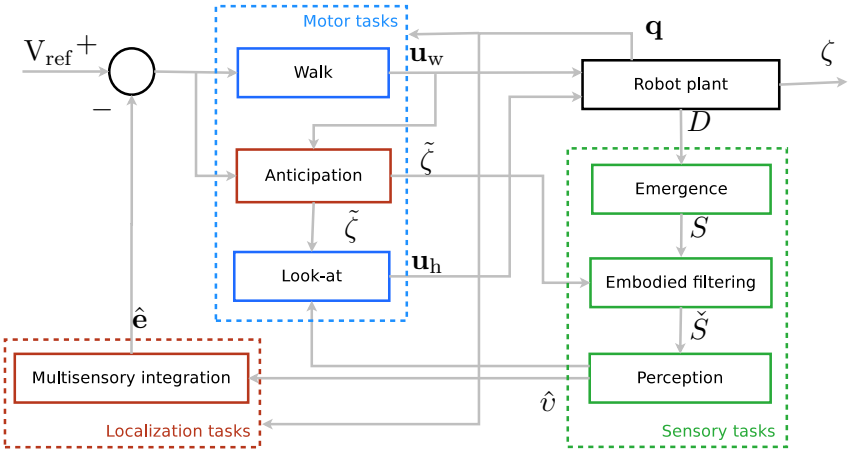


Figure 5.3 – Behavior block diagram view. The behavior corresponds to the regulation of the observed state $\hat{\mathbf{e}}$. The egocentric localization of the stimulus is represented by ζ, with prediction $\tilde{\zeta}$. The control signals $\mathbf{u}_\mathrm{w}$ and $\mathbf{u}_\mathrm{h}$ are sent respectively to the walk and the head-direction motion primitives of the robot. The information retrieved from the robot are the unprocessed image D and the current joint configuration $\mathbf{q}$. The embodied feature set S is ranked according to the anticipation in the feature set $\check{S}$. The observation of the object's pose in the vision system is denoted by $\hat{v}$.

Walk task

It is in charge of controlling the walk primitive to steer the agent toward the object. From the estimation of the localization error (see Eq. (4.16)), a motion command is sent to the robot in agreement to Eq.(5.6). The motion has to be expressed with respect to the reference system of the walk primitive. As illustrated in Fig. 4.13, in the case of the robot Nao the motion is expressed with respect to the frame G, that is placed at the intermediate point between the center of projection of both feet on the ground.

Look-at task

It is in charge of controlling the articulated neck of the robot to direct the view toward the object. This task is formally defined in Sec. 4.4.2. The idea is that two internal subtasks are executed in sequence. The first one corresponds to an open-loop predictive PBVS scheme that directs the gaze toward a predicted location on the scene. The second subtask corresponds to a regulatory close-loop IBVS scheme that receives by visual feedback the desired retinal distance to center the object in the field of view.

Emergence task

Visual exploration and scene understanding are efficiently accomplished by human beings. When approaching the object, exogenous (e.g. the visual stimulus) and endogenous (e.g. the body posture) information emerge. Moreover, models of human attention (see Sec. 3.2) have pointed out to a mechanism in charge of selecting the saliency of information originated from two independent processes, one occurring deliberatively (i.e. top-down), and the other involuntarily (i.e. bottom-up). The Emergence task is inspired by the Feature Integration Theory (FIT) of attention (see Sec. 3.2.3). In this sense, visual saliency is organized into separate features or layers, each of which registering the presence of different properties (e.g. color, edges, shapes, optic flow, etc). The endogenous saliency must also be represented. The source of information considered in the study is proprioceptive, that is, the instantaneous posture of the agent. The objective of the emergence task is thus to register the context of the behavior. The output of the task is the feature set S containing the saliency of information.

Anticipation task

A work by Lungarella & Sporns [114] has explored the relation between sensory-motor coordination, body morphology, and information processing. In the study, quantifiers for information content were defined, in order to estimate the temporal evolution of sensory and motor information under two experimental conditions: sensory-motor coordination, and uncoordinated motion. The results corroborated the research hypothesis, according to which, higher levels of information correlation occur when actions and perceptions are coordinated. Moreover, sensory-motor coordination reduced the dimensionality of the information content, given the perceptual regularities induced in the task (i.e. the entropy on sensory data was reduced).

Thus, the objective of the anticipation task is to conveniently exploit the two effects described for sensory-motor coordination (i.e., the induction of perceptive regularities and the information redundancy), in order to provide the discriminative process of the object with information on the coupling. Therefore, a prediction for the next observation of the state is produced from the last observation and the current action. Predictive models have been employed for diverse purposes in robotics and automation research. Just to mention a few applications, they have been used for calibration tasks (see Khalil & Dombre [96]), robot localization (see Thrun et al. [178]), and motor coordination (see Arbib et al. [7]).

Embodied filtering task

This task is inspired by the spotlight metaphor of attention (see Sec. 3.2.2) in the sense that knowledge about where in space a stimulus will occur can be used to improve the efficiency of detection. Therefore, the anticipation of perception can be conveniently related to the emergence of features. Here, the actual information saliency is compared to the predicted saliency. As shown in Fig. 5.4, an periphery-to-center flow projects information from salient regions (e.g, the blobs centroids) to the sensory ego-space. It is also possible to proceed in the opposite direction, a center-to-periphery flow can be employed to predict the evolution of visual features. Unlike the Verification Vision principle (in Bolles [21], see Sec. 3.3.4) the projections are expected to be coarse, since the model of the object is unavailable at this stage, and the frequency of acquisitions is expected to be low.

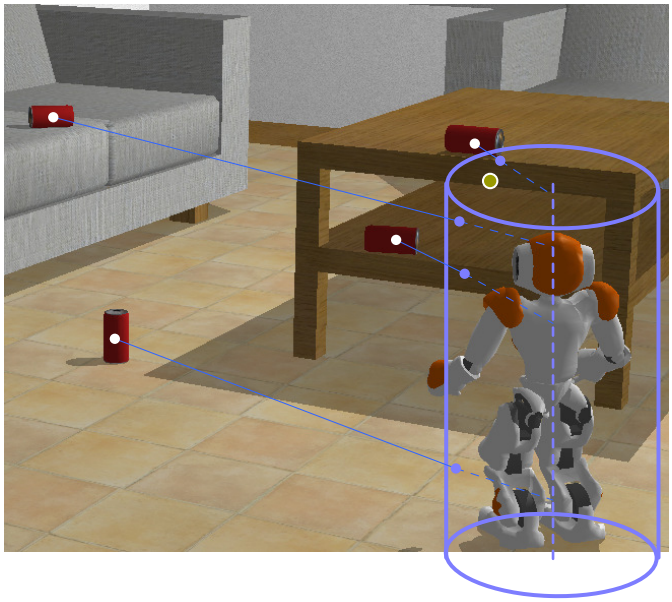


Figure 5.4 – Embodied filtering illustration. The agent is approaching the red can on the top of the table. The white dots correspond to the center of the salient objects. The estimate on the distance to the blob center is unavailable during the saliency analysis, thus, the last observation ${}^{B}\hat{\zeta}_{\rho}$ is heuristically used (see Eq. (4.15)). The projection of the blobs in the ego-cylinder is represented by the blue dots. The predicted localization is represented by the yellow dot.

Perception task

The goal of the task is to estimate the object's pose with respect to the visual system. From the information provided by the filtering process, where diverse features relate the emergence of information with the prediction, the agent solves the Perception task in two phases. Firstly, it selects the retinal region associated to the object with certain confidence. According to Brooks [29], one of the challenges encountered in the study of emergent behavior is to find efficient ways to fuse multiple sources of perceptual information when needed. For this, the discriminative process may rely on diverse techniques that are available in the literature of machine learning (e.g. artificial neural networks, support vector machines, among others). In this study a Bayesian network structure is proposed in Sec. 5.4. Secondly, in case the object is present in the scene view, its posture

with respect to the visual system is determined by fitting the salient region to the object 3D model (many geometries can be considered, the cases of a cylindrical container and a rectangular surface were detailed in Sec. 4.5.3).

Ego-localization task

The approach to the object is based on ego-centric localization. The choice for the ego-cylinder structure for multi-sensory integration was discussed in Sec. 4.5. Basically, the definition of the sensory ego-cylinder is obtained by heuristically setting the z-axis perpendicular to the ground, under the assumption of motion over a plane surface. The localization depends on the transformation between the reference frames G, T, and C, defined from the current joint configuration $\mathbf{q}$ of the robot. Thus, in this task the control parameters (see Fig. 4.2) and the localization error in Eq. (4.16) are used.

Behavior autonomy

According to the behavior scheme defined in Fig 5.3, the agent must accomplishing the task by relying on action-independent knowledge about the object (i.e. a rough 3D model), action-oriented representations obtained from the embodied perception of the object, and a supervised demonstration. Though, two important aspects remain to be discussed when developing robust applications in service robotics. The first one is how to integrate these elements in a organized, reusable, and efficient way in order to solve the task. The second one is how to know when things are not going as expected, so error recovery is possible.

Given the progresses in the fields of information technology and artificial intelligence, ubiquitous computing relying on the client-server computational paradigm has proved to be mature enough as to provide many solutions in the form of mobile applications. These success cases have been inspiring researchers in service robotics when facing the challenges of unstructured applications in heath-care, assistance, and other domains. Hence, the idea is that robot applications can share knowledge or be assisted by cloud-connected resources. There are currently initiatives that focus on the definition of robot architectures (e.g. Vasiliu et al. [181]) that integrate distributed resources to the task, so mitigating specific constraints of the robot platform. In parallel, several research communities have engaged in the definition of ontologies for knowledge sharing, distributed learning, and the collection and reuse of information for practical applications (e.g. Waibel et al. [184]).

Therefore, the idea is to provide the agent with a locally autonomous implementation of the behavior scheme, that can be continuously assessed. Figure 5.5 illustrates the hybrid architecture proposed. The Behavior node implements the local task (for this study it corresponds to scheme described in Sec. 5.3.3). Based on the analysis of local information obtained by the action-perception coupling (i.e. the embodied features described in Sec. 5.4.1), the Deliberation node estimates the confidence in the task, so remote resources are used in case this estimate is below a given tolerance. This is ensured by a Bayesian network implementation (detailed in Sec. 5.4.2) that observes the signals of embodied features to discriminate the object and to evaluate the task consistency. The Deliberative process must also preserve the safety on the local state (e.g. by stopping motion, or sending the robot to rest), since the remote access to resources is subject to disturbance (e.g. interruption in the network service). Therefore, the agent would ideally not depend

on remote resources. The philosophy behind the designed hybrid model is going to be better understood in Sec. 5.5.3, where the case study implementation is detailed.

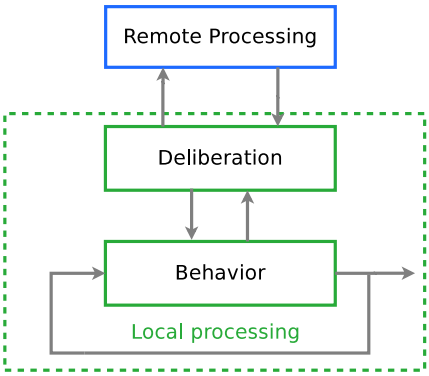


Figure 5.5 – Hybrid task architecture. The deliberative process evaluates the consistency of the autonomous local execution of the behavior. Once inconsistency is detected, remote processes are queried for support.

Embodied features

A set of embodied features is proposed in this work as action-oriented representations for the task. The features carry information on the spatial, morphologic, and topographic properties related to the visual stimulus, and the body posture during the task. Table 5.2 presents their definition whereas Fig. 5.6 illustrates the concept behind. The first two features are based on the blob moments (see Eq. (3.4)). The radio-aspect F_3 is defined from the width and height of the minimum bounding-box (MBB) enclosing the blob, where the angle between the MBB's principal axis and the image x-axis is $\gamma = 0.5(\text{atan}(2m_{11}/(m_{20} - m_{02})))$. The feature F_4 includes proprioceptive information from the instantaneous posture of the neck. The feature F_5 represents the topographic relation between the blobs, it is a descriptor of the presence of saliency at a four cardinal neighborhood.

Feature	**Description**
$F_1 = (\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}})$	Retinal blob centroid.
$F_2 = m_{00}$	Retinal blob area.
$F_3 = H_{\text{height}}/H_{\text{width}}$	Radio-aspect, where H denotes the oriented bounding box.
$F_4 = (\alpha, \beta)$	Posture, with α and β the pitch and yaw neck angles.
$F_5 = v(S, s)$	Topology, where v attributes a 4-bit vicinity code according to the saliency set S around the blob s.

Table 5.2 – Embodied features. The Features F_1 and F_4 are directly expressed in the sensory space. All but F_4 capture information about the stimulus (i.e the retinal location, area and morphology, the topological arrangement).

As it can be seen, the feature design has been inspired on theories of human attention. In this sense, based on the spotlight metaphor and FIT (see Sec. 3.2) some features represent information directly or relative to the sensory space (e.g. the centroid in the retinal, and joint positions). It is important to notice that despite some features are computationally derived (i.e. F_1, F_3, F_5), there is no obvious redundancy in the information they provide.

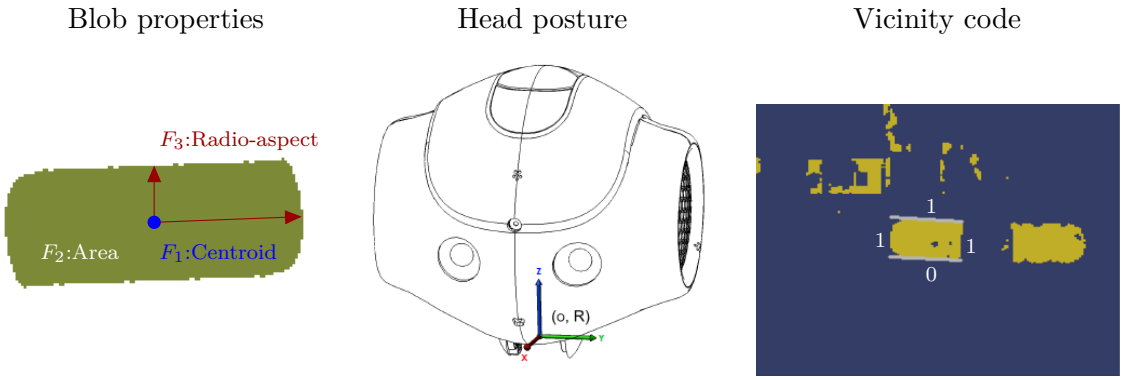


Figure 5.6 – Embody features illustration. On the left the first three features are shown corresponding to measurements related to the salient blob. At the center the feature representing the neck attitude is illustrated. On the right the topographic descriptor of the blob in relation to the retinal saliency is shown.

A second set of variables are defined by considering the anticipatory process. For this, a deterministic motion model is employed under the assumption of ideal noise-free robot, moving at constant velocity $\mathbf{v} = [\dot{X}\ \dot{Y}\ \omega]^{\mathrm{t}}$ along the time interval Δt. Thus, a prediction for the localization of the object ${}^{B}\tilde{\zeta}$ is obtained from the last observation available ${}^{B}\hat{\zeta}$, and the expected displacement $\mathbf{m} = \mathbf{v}\Delta t$. Table 5.3 presents the definition of the variables where the actual saliency is related to the anticipated information flow. The idea of the feature $\overline{F_1}$ has been illustrated in Fig. 5.4 where from a periphery-to-center flow, information from salient regions (e.g, the blobs centroids) are projected to the sensory ego-space. Notice that among the variables defined, some are not directly related to the prediction of the localization. Such is the case of $\overline{F_3}$ and $\overline{F_5}$, where the criteria used for anticipation are the statistical regularities induced by the sensory-motor coupling (e.g. the perspective from which the object is seen will change gradually between consecutive acquisitions, as the robot approaches it).

Expression	**Description**
$\overline{F_1} = \lvert\sigma(F_1' - {}^{B}\tilde{\zeta})\rvert$	F_1' denotes the projection of the blob centroid in the ego-space, ${}^{B}\tilde{\zeta}$ is the predicted localization of the object, and σ weights the contribution of each component.
$\overline{F_2} = 1 - \frac{F_2}{\left(\frac{\hat{\zeta}_\rho + m_\rho}{\hat{\zeta}_\rho}\right) F_{2(k-1)}}$	Relation between the actual blob's area F_2 and the simulated area from the expected motion m. Here $F_{2(k-1)}$ denotes the saliency during the last observation ${}^{B}\hat{\zeta}$.
$\overline{F_3} = \lvert F_{3(k)} - F_{3(k-1)}\rvert$	Absolute difference between the current radio-aspect and the last perceived denoted by $F_{3(k-1)}$.
$\overline{F_4} = \lvert\check{F}_4 - \tilde{F}_4\rvert$	Absolute difference between the simulated posture of the neck $\check{F}_4$, that would center the blob on the visual field, and the predicted attitude of the neck $\tilde{F}_4$.
$\overline{F_5} = \sum_{i \in N} \delta(F_{5(k-1)i}, F_{5i})$	Estimate of the topographic relation through the Kronecker delta function $\delta(a, b)$. The neighborhood set is defined by $N = \{left, right, up, down\}$.

Table 5.3 – Filtering features.

Bayesian network for information fusion

The information provided by the embodied features is diversified so various aspects of the local context are captured. Moreover, different frames of reference may be involved. A Bayesian network (BN) is defined to integrate the available information, so relevant saliency is discriminated for localizing the object. Thus, a BN is a directed acyclic graph that represents the conditional probabilities of interconnected random variables. BNs have been used for diverse automatic diagnosing and recognition tasks (see Ertel [63]). In a BN, a node is assumed to be conditionally independent from non-successors, given its parents. The joint probability $p(N_1, ..., N_n)$ of the nodes N_i is expressed by

$$p(N_1, \ \dots \ , N_n) = \prod_{j=1}^{n} p(N_j | parents(N_j)). \tag{5.7}$$

One important advantage of knowledge representation through BNs is that the information contained is directly understandable by humans, which facilitates doing future modifications (e.g. including new features, or more complex observations).

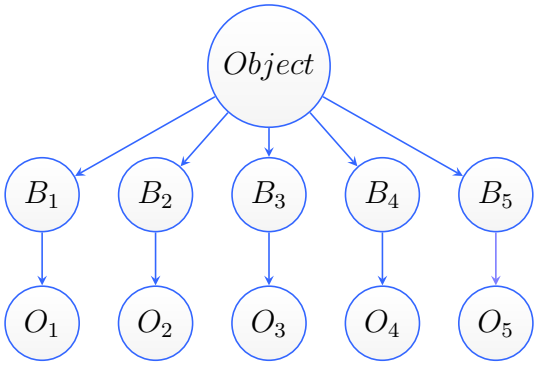


Figure 5.7 – Bayesian network for contextual information fusion.

As illustrated in Fig. 5.7, the structure proposed for the network corresponds to a tree of height 2. The root node is a binomial random variable, which represents the probability that the blob saliency observed is related to the object of interest. The intermediate nodes B_i are binomial random variables that represent the a posteriori probability of the features, given the observation of the object. This layer is included in order to simplify adjustments to the contribution of the features to the discriminative process. The leaves O_i are multinomial random variables that represent the a posteriori probability of observing a particular intensity of $\overline{F_i}$, given B_i. The tree can easily accommodate new features by horizontal expansion. Formally, the query of interest is defined by

$$p(Object|O_1, \ \dots \ O_5) = \frac{1}{Z} \prod_{i=1}^{5} p(Object) p(B_i|Object) p(O_i|B_i), \tag{5.8}$$

with Z a scaling factor depending on the observations O_i available. A tutorial on the calculation of $p(Object|O_i)$ is given in Appendix A.

Probabilistic independence between the branches of the network (the nodes B_i and O_i) is assumed for convenience, which is also known as a *naive Bayes classifier*. Research on physiology has shown that perception in cross-modal tasks can be described as a context-dependent Bayesian multi-sensory integration process (Denve & Pouget [54]).

It is believed that such knowledge is represented by a network of *basis functions* (Pouget & Sejnowski [148]). Thus, in physiology research, the hypothesis of probabilistic independence between multi-sensory cues would hardly be justified. However, as pointed out by Ertel [63], the naive classifier assumption has lead in practice to good results.

The most likely blob b among the saliency set S is obtained by maximizing the expression

$$s = \text{argmax}_{b \in S} p(Object|\ B_i, O_i). \tag{5.9}$$

Thus, the BN can be used to classify the saliency, while providing an estimate of the certainty in such classification. As it shall be discussed in the case study of Sec. 5.5.5, this information is of crucial importance to the Deliberative process of the hybrid architecture presented in Fig. 5.5, once it has to decide whether or not resorting to remote processing.

The process of object recognition consists in discerning between two hypotheses. The *null hypothesis* would state that a particular observation does not correspond to the expectation about the object, whereas the *alternative hypothesis* would consider it as adequate to the expectation. Therefore, it is possible to commit two types of errors. The error type T_1 consists in accepting an observation that does not belong to the object (i.e. a false positive), whereas with the error type T_2 an observation that is related to the object is rejected (i.e. a false negative). Let the events of interest be enumerated by $\mathrm{E}_{1i} = \{\mathrm{B}_i, \mathrm{Object}\}$, $\mathrm{E}_{2i} = \{\neg\mathrm{B}_i, \mathrm{Object}\}$, $\mathrm{E}_{3i} = \{\mathrm{B}_i, \neg\mathrm{Object}\}$ and $\mathrm{E}_{4i} = \{\neg\mathrm{B}_i, \neg\mathrm{Object}\}$. Ideally, the information provided by B_i to the discriminative process is maximal when $p(\mathrm{E}_{1i}) = p(\mathrm{E}_{4i}) = 1$, so $p(\mathrm{E}_{2i}) = p(\mathrm{E}_{3i}) = 0$. Contrarily, no information is provided at the uniform distribution, that is, when the probability of the aforementioned events is 0.5. As shown in Fig. 5.8, the probability distributions $p(B_i|\mathrm{Object})$ and $p(B_i|\neg\mathrm{Object})$ can be estimated at iteration t from the previous decisions taken by the BN, according to the expression

$$p(B_i|Object)_{(t)} = \frac{\sum_{d=1}^{k} \gamma^{d-1} p(B_i|Object)_{(t-d)}}{\left(\frac{1-\gamma^k}{1-\gamma}\right)}. \tag{5.10}$$

The parameter k corresponds to the size of the sliding window. The role of the constant $\gamma \in]0, 1]$ is analogous to the discounted reward factor employed in reinforcement learning, where the contribution of neighbor states is related to the proximity to the actual state. The denominator is a normalization term that corresponds to the solution of the geometrical series $\sum_{i=1}^{j} \gamma^i$. At each decision process the feature set $\check{S} = \{\check{S}_\mathrm{o}, \check{S}_{\overline{\mathrm{o}}}\}$ is partitioned into the set $\check{S}_\mathrm{o}$, that contains the observations related to the selected candidate, and the set $\check{S}_{\overline{\mathrm{o}}}$, that contains the observations related to the discarded candidates. Thereby, the previous distributions are obtained from

$$p(B_i|Object)_{(t-d)} = \frac{1}{n} \sum_{o_i \in \psi} p(E|o_i). \tag{5.11}$$

where $n = |\psi_i|$ and o_i is the observation of the intensity of feature $\overline{F}_i$. For the distribution $p(B_i|\mathrm{Object})$ the set considered is $\psi = \check{S}_\mathrm{o}$ and $E \in \{\mathrm{E}_{1i}, \mathrm{E}_{2i}\}$, whereas for the distribution $p(B_i|\neg\mathrm{Object})$ the set considered is $\psi = \check{S}_{\overline{\mathrm{o}}}$ and $E \in \{\mathrm{E}_{3i}, \mathrm{E}_{4i}\}$. In Sec. 5.5.5 different policies are studied for determining the probability distribution of features B_i.

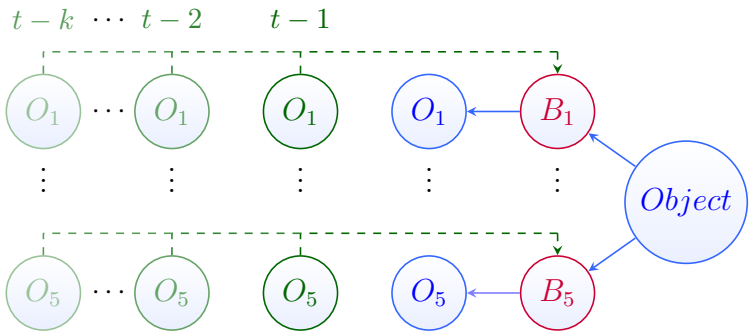


Figure 5.8 – Dynamic policies for the Bayesian network. In green the past observations $O_{i(t-1,\ \ldots\ ,k)}$. These nodes don't belong to the actual network, the gradual fading of the color illustrates a decaying contribution (taking $\gamma < 1$) according to recency (see Eq. (5.10)). In red the dynamically updated nodes B_i.

Case studies

The behavioral scheme proposed in Sec. 5.3.3 consists in a general description of the solution of the task. Thus, a particular implementation for the model components must be provided and evaluated. This is done through the definition of three case studies. Since embodied observations are used, the objective of the first study is to estimate the system parameters. For this, the spontaneous occurrence of the features is registered under distinct delay profiles. In the second study the task is evaluated in a simulated scene, where the robot has to approach a particular object among multiple instances. Different policies for determining the probability distribution of features B_i (see Fig. 5.7) are compared. In the third experiment, an implementation of the hybrid architecture of Fig. 5.5 is evaluated in a real task, where the robot has to approach multi-color tea cans. Therefore, the autonomous execution of the behavior is studied firstly to verify the extent to which the agent can perform the task without using remote resources. Then, the Deliberative module of the hybrid solution is activated so failures can be detected and the success rate can be improved.

Materials and resources

The platform is the humanoid robot Nao by Aldebaran Robotics. The control program is implemented in the C++ programming language. The images are captured at 320×240 pixels resolution. The vision processing is obtained with the support of the OpenCV 2.4.8 library. The Bayesian network implementation is provided by the dlib C++ Library version 18.13. For prototyping the network the SamIam tool version 3.0 is also used. The robot functionalities are accessed through the naoqi 1.14 library. The algorithms are developed in the Eclipse Juno IDE under Ubuntu 12.04.5 LTS (Precise Pangolin). The simulations are conducted in the Webots robot simulator 7.4.0 by Cyberbotics. The results are processed in Gnu Octave 3.2.4 and the KNIME data analytics, reporting and integration platform 2.10.4. The on-board calculations relied on an ATOM Z530 1.6GHz CPU, with 1 GB RAM, 2 GB flash memory, and 4 flash memory dedicated to user purposes. The study also included a DELL Vostro 1500 laptop (Intel Core 2 Duo 1.8GHz 800Mhz, 4.0GB DDR2 667MHz RAM, 256MB NVIDIA GeForce 8600M GT).

Behavior scheme implementation

Motor tasks

In the Walk task the agent has to move to the desired location, and to stop once all the components of the observed localization error ${}^{B}\hat{\mathbf{e}}_1$ (as defined in Eq. (4.16)) are smaller than a given threshold ϵ. The tolerance considered is the same as Sec. 4.6.2 which is a radial distance $\epsilon_\rho = 0.05$ meters (m), the azimuth $\epsilon_\theta = 0.04$ radians (rad), and $\epsilon_\phi = 0.1$ rad for the orientation component. The walk primitive is controlled in position since it provided more precise results. Motion commands are expressed in Cartesian coordinates conforming to Eq. (5.6), and sent to the walk primitive under the assumption of constant velocity motion. The robot may not faithfully execute the request, so the estimated motion ${}^{\mathrm{G}}\tilde{\mathbf{m}}$ obtained from the request ${}^{G}\mathbf{m} = [X\ Y\ \varsigma]^{\mathrm{t}}$ (with the linear components denoted by X and Y, and the angular component by ς), is defined by

$$ {}^{G}\tilde{\mathbf{m}} = \mathbf{H}\,{}^{G}\mathbf{m}, \tag{5.12} $$

where $\mathbf{H}$ is a 3×3 matrix that represents the estimated efficiency of motion, including the coupling between the motion components of the walk primitive ($\mathbf{H}$ is determined in the first study, and corresponds to the left side of Tab. 5.4). The mean walk velocity under the gait configuration recommended by the manufacturer is estimated to be around $\tilde{\mathbf{v}} = [0.022\ \text{m/s}\ 0.04\ \text{m/s}\ 0.106\ \text{rad/s}]^{\mathrm{t}}$. Continuous motion is achieved by sending commands at regular time intervals. In order to prevent that unforeseen delays affect the fluidity of the walk, the actual displacement sent considers a larger delay (e.g. 1.5 times the expected value). Thus, a new command would be ideally sent before the routine could finish the previous one. If this would not be the case (e.g. due to losing the object, a program crash, etc.), the robot would stop moving after a while. This strategy ensures a fluid walk while keeping the safety aspects. For speeding up convergence, given the observation noise and the fact that the walk primitive is less precise in continuous motion, once the robot is nearly at the desired location (at ${}^{B}\hat{\mathbf{e}}_{1\rho} < 0.1$ m), the Walk task switches to a step-by-step policy (i.e. a new correction is sent only after finishing the previous one).

The Look-at task is also controlled in position and the same implementation detailed in Sec. 4.6.2 is employed. Thus, a tolerance $\epsilon = 0.03$ rad is admitted for convergence of $\hat{\mathbf{e}}_2$ in the predictive motion sub-task, and a tolerance for 10 pixels is accepted for the object centering sub-task ${}^{\mathrm{T}}\mathbf{e}_3$ (see Eq. (4.21)). The head posture is regulated independently from the walk (i.e. the tasks run in parallel), which means that the motion induced by the Walk task can affect the convergence of the Look-at task, notably, at slow turning of the head. Thereby, a velocity profile of 4 rad/s is employed so convergence in both sub-task is easily obtained.

Localization tasks

The ego-localization implementation was detailed in Sec. 4.6.3. In this study B is selected as the same definition of frame G (see Fig. 4.11). The Anticipation task involves a deterministic prediction that assumes an ideal noise-free robot, moving at constant velocity (see Eq. (4.17)). A saturation is applied to the motion estimation ${}^{G}\tilde{\mathbf{m}}$ in Eq. (5.12), in order to take into account the maximum velocity ${}^{G}\mathring{\mathbf{v}}$ attainable under the actual gait settings. Thereby, the motion estimation considered is ${}^{G}\bar{\mathbf{m}} = \min({}^{G}\tilde{\mathbf{m}}, {}^{G}\mathring{\mathbf{v}}\Delta t)$, with $\mathring{\mathbf{v}}$ taken in this study as the mean velocity profile of the default gait configuration of Nao.

Sensory tasks

The Emergence task relies on the whole segmentation technique (see Sec. 3.4.3) for processing the top-down visual saliency (that is, only one saliency layer is used). From the supervised detection of the object, a color model is obtained by sampling the pixels that represent it on the image. The features detailed in Tab. 5.2 are calculated from the blob morphology and disposition on the image, and the posture of the neck is registered. The Embodied Filtering task requires no additional efforts that applying the definitions given in Tab. 5.3 in relation to the prediction of the localization ${}^{G}\tilde{\zeta}$ (see Eq. (4.17)).

The Embodied Filtering task contains the BN (see Fig. 5.7). The observation of the leaf nodes consider discrete events, so discretization through clustering is employed to rank the measured signals according to levels of intensity. Data partitions are defined from the statistical properties of the information flow recorded in the task (the numerical values and the way to proceed to calculate them is detailed in Sec. 5.5.4 and Tab. 5.6). Five partitions are established from the mean μ_i and the standard deviation σ_i of features $\overline{F_i}$ (see Tab. 5.3). Thus, the observations are grouped according to $f(\overline{F_i}, \mu_i, \sigma_i)$ such that

$$f(\overline{F_i}, \mu_i, \sigma_i) = \begin{cases} L_0 \leftarrow \overline{F_i} & \text{if} \quad \overline{F_i} \leq \mu_i - 4\sigma_i \\ L_1 \leftarrow \overline{F_i} & \text{if} \quad \mu_i - 4\sigma_i < \overline{F_i} \leq \mu_i - 2\sigma_i \\ L_2 \leftarrow \overline{F_i} & \text{if} \quad \mu_i - 2\sigma_i < \overline{F_i} < \mu_i + 2\sigma_i \\ L_3 \leftarrow \overline{F_i} & \text{if} \quad \mu_i + 2\sigma_i \leq \overline{F_i} < \mu_i + 4\sigma_i \\ L_4 \leftarrow \overline{F_i} & \text{if} \quad \overline{F_i} \geq \mu_i + 4\sigma_i \end{cases} \tag{5.13}$$

Given the symmetry around L_2, only three clusters are defined to represent the levels of intensity of features $\overline{F_i}$ in the leaf nodes O_i of the BN. Thereby, the *nearest neighboor method* (see Ertel [63]) is employed to classify the observations, so

$$O_i = \begin{cases} 0 & \text{if} \quad \overline{F_i} \in L_0 \cup L_4 \\ 1 & \text{if} \quad \overline{F_i} \in L_1 \cup L_3 \\ 2 & \text{if} \quad \overline{F_i} \in L_2 \end{cases} . \tag{5.14}$$

After the BN evaluation, the selected retinal region is passed to the localization routine in the Perceptive task which is in charge of determining the object pose in the camera frame. The principles behind the implementation of this routine were detailed in Sec. 4.5.3, where a cylindrical wrapper and a rectangular surface were modeled.

Hybrid architecture implementation

As illustrated in Fig. 5.5, three components are defined in the hybrid architecture: the Behavior, the Deliberation, and the Remote Processing nodes. The implementation for the Behavior node has been provided in Sec. 5.5.2. Thus, it includes the autonomous behavior scheme developed from the EC methodology. The implementation of the Deliberation node (which is in charge of detecting task inconsistence and requiring remote assistance), is based on two important estimates obtained from the activity of the BN: the degree of confidence ψ, and the discriminative power ϱ. Accordingly, ψ is defined by

$$\psi = p(Object|\ B_i, O_i), \tag{5.15}$$

where $p(Object|\ B_i, O_i)$ is the probability issued by the BN, given the certainty in B_i and the observed evidence O_i. The discriminative power is defined by

$$\varrho = \psi_1 - \psi_2, \tag{5.16}$$

so ψ_1 and ψ_2 is the degree of confidence on the two most likely candidates in the saliency set S, obtained by applying Eq. (5.9) twice: one to the full set S, and the other to the reduced set $S_\mathrm{r} = S - \{s_1\}$, excluding the first selection s_1.

The state automate is illustrated in Fig. 5.9. The transition events are determined by $t(.,.)$ which observes ψ and ϱ to decide whether remote help is requested, thus

$$t(\psi, \varrho) = \begin{cases} \text{True} & \text{if} \quad \psi > \epsilon_1 \wedge \varrho > \epsilon_2 \\ \text{False} & \text{otherwise} \end{cases} . \tag{5.17}$$

Therefore, the transition events are defined so $g_1 = t(\psi, \varrho)$, $g_2 = \neg g_1$, and g_3 is an interruption signal to stop the program. The thresholds for the transition are ϵ_1 and ϵ_2. In relation to the states, in Start the program is initialized. Remote Assistance includes a graphical user interface (GUI) of the application program where the user can access the on-board captures and perform actions, such that: a) providing the visual demonstration for the object recognition, b) clicking above a salient region to specify the desired object to be tracked, c) suggesting a search direction to relocate the object, or d) aborting the program. In the Deliberation state data received remotely is passed to the Behavior module. A start/stop signal is also sent to the robot in case the event g_2 or g_3 are produced (either from a remote user request of a local erroneous condition detected). In the Local Execution state robot motion is controlled by the Behavior node. In the End state the application program finishes the execution.

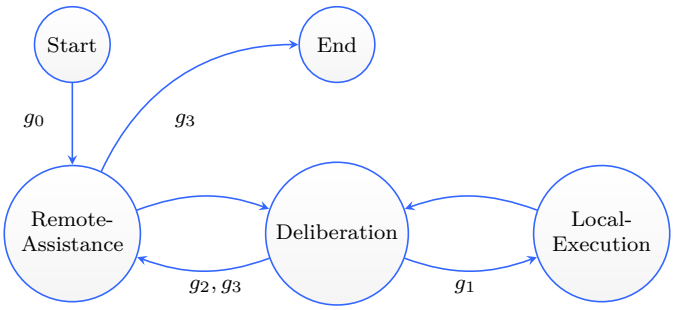


Figure 5.9 – Deliberation state automate. The transition events are denoted by g_i. Compulsory transition have no events associated.

CS-I: Model parameters estimation

The objective of this case study is to characterize the system and to estimate the model parameters, including the motion profile of the walk primitive and the evolution of the embodied features. In this sense, since the behavior model relies on motion primitives already acquired by the agent, it is necessary to provide the Anticipation task with knowledge concerning the characteristics of locomotion. Similarly, the Embodied Filtering task would rely on information on how the features $\overline{F_i}$ evolve under undisturbed conditions, so the object can be properly recognized. Moreover, in view of the multiple constraints affecting the generation of motion (e.g. keeping balance, energy safe, etc.), it is possible that the walk primitive does not execute accurately the motion requests, and such errors must be taken into account.

Experiments

In the first experiment the agent is required to walk in open-loop to distinct locations under the default gait profile as recommended by the manufacturer. The case conditions are variations in the step distance along the 3 DOF of motion. The ground truth is obtained from Webots. Variations of 0.01 m are applied in the positive and negative sense until a maximal distance of 0.1 m for the X and Y linear motion, and 0.04 rad until a maximal turn of 0.4 rad for the angular motion. Each experiment is repeated three times to avoid simulation bias, so the total number of cases are 3 (repetitions)$\times$ 3(DOF) $\times$ 20 (10 cases in the positive and negative senses) = 180. The second experiment is designed to study the anticipation process. For this, a controlled scene that contained a single salient soda can was designed. The robot's task is to approach the can from 3 different initial configurations. Twenty delay profiles for the visual feedback are simulated (from 100 ms until 2000 ms, varying in 100 ms), each case is repeated 2 times, so the total number of cases are 2 (repetitions) $\times$ 3 (initial conditions) $\times$ 20 (delay profiles) = 120.

Results

The results obtained for the first experiment are given in Tab. 5.4. On the left segment the mean coupling was calculated by relating the motion requested to the ground truth. As the figures on the diagonal suggest, the angular displacement ς is the more efficiently accomplished by the walk primitive, followed by the saggital displacement X. The less efficient motion corresponded to the lateral displacement Y, which, as shown on the right segment of the table, also presented greater variability.

μ	X	Y	ς	σ	X	Y	ς
X	0.613	0.351	0.199	X	0.057	0.054	0.122
Y	0.273	0.476	0.017	Y	0.102	0.174	0.024
ς	0.007	0.022	0.662	ς	0.008	0.018	0.055

Table 5.4 – Motion profile. The average relation $\mu_{ud} = \text{mean}(w_u/w_d^*)$ and the standard deviation σ_{ud} are calculated for $u, d \in \{X, Y, \varsigma\}$. The ground truth motion w_u was obtained from Webots, with ${w_d}^*$ denoting the motion request. The information of the table should be read as follows. When requested to move in the saggital plane direction w_{X}^* meters, the robot moved on average $w_{\text{X}} = 0.613 w_{\text{X}}^*$ meters in the saggital plane direction, $w_{\text{Y}} = |0.273 w_{\text{X}}^*|$ meters in the frontal plane direction, and rotated $w_\varsigma = |0.007 w_{\text{X}}^*|$ radians.

The results for the second experiment are provided in Tab. 5.5, where the difference between the observed localization ${}^G\hat{\zeta}$ and the predicted localization ${}^G\tilde{\zeta}$ under distinct delay profiles is given. A comparison between the mean localization discrepancy at distinct delay profiles is illustrated in Fig. 5.10. It is noted that the discrepancy in the bearing μ_θ appears to be more sensitive to increasing delay, whereas the height μ_ι would be the less sensitive. The heading direction discrepancy μ_ϕ appears to be always high. Table 5.6 presents some of the results obtained for the evaluation of the features $\overline{F_i}$ under distinct delay profiles. Figure 5.11 illustrates the comparison on the evolution of the feature means. The ascending tendency is more pronounced for the features $\overline{F}_2$, and $\overline{F}_4$.

r	μ_ρ	σ_ρ	μ_θ	σ_θ	μ_ι	σ_ι	μ_ϕ	σ_ϕ
100	0.026	0.055	0.013	0.009	0.004	0.008	0.292	0.466
300	0.020	0.024	0.920	0.062	0.021	0.045	0.040	0.030
500	0.045	0.063	0.046	0.025	0.010	0.032	0.538	0.564
1000	0.045	0.059	0.085	0.047	0.016	0.046	0.607	0.646
1700	0.034	0.040	0.110	0.073	0.024	0.066	0.479	0.483
2000	0.053	0.071	0.118	0.093	0.022	0.045	0.496	0.458
mean	0.042	0.061	0.080	0.054	0.015	0.041	0.461	0.475

Table 5.5 – Difference between the observed localization ${}^G\hat{\zeta}$ and the predicted localization ${}^G\tilde{\zeta}$. The delay profiles $r \in \{100, 200, \ldots, 2000\}$ are expressed in ms, distances in m, and angles in rad. The average $\mu_d = \text{mean}(|{}^G\hat{\zeta}_d - {}^G\tilde{\zeta}_d|)$, and the standard deviation σ_d for each localization component $d \in \{\rho, \theta, z, \phi\}$ (see Eq. (4.15)), are shown for some delay profiles r. The last row presents the global mean obtained by column.

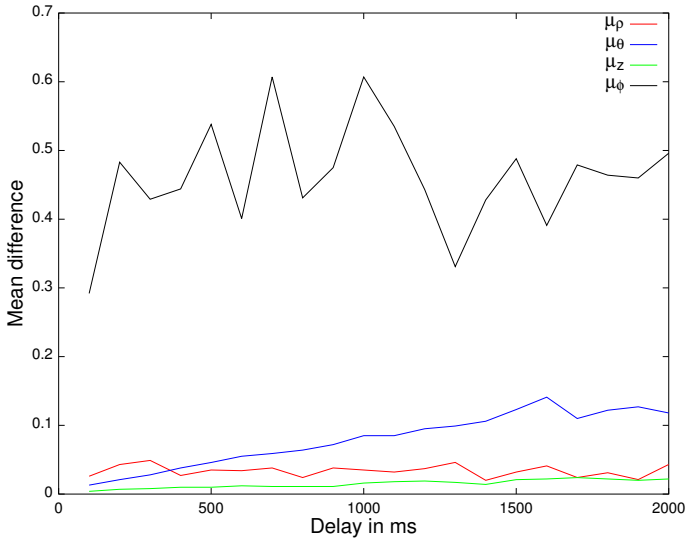


Figure 5.10 – Mean discrepancy between the localization and the prediction for each delay profile. Distances are expressed in m and angles in rad.

r	μ_1	σ_1	μ_2	σ_2	μ_3	σ_3	μ_4	σ_4
100	0.011	0.016	0.037	0.075	0.017	0.027	0.046	0.066
300	0.020	0.024	0.080	0.062	0.021	0.045	0.040	0.030
500	0.022	0.030	0.117	0.088	0.027	0.047	0.073	0.092
1000	0.027	0.022	0.119	0.080	0.031	0.037	0.079	0.062
1700	0.044	0.048	0.197	0.099	0.025	0.023	0.123	0.073
2000	0.042	0.042	0.190	0.116	0.032	0.030	0.126	0.094
mean	0.026	0.035	0.138	0.095	0.031	0.038	0.092	0.077

Table 5.6 – Embodied filtering observation. The delay profiles $r \in \{100, 200, \ldots, 2000\}$ are expressed in ms. The average $\mu_i = \text{mean}(\overline{F_i})$ and the standard deviation σ_i of some delay profiles r are given for features $\overline{F_i} = \{\overline{F_1}, \ldots, \overline{F_4}\}$ (see Tab. 5.3). Since there was only one salient object in the simulation, the effect of the delay over $\overline{F_5}$ could not be appreciated, thus, the values $\mu_5 = 0$ and $\sigma_5 = 0.5$ were fixed manually in the study. The last row presents the global mean obtained by column.

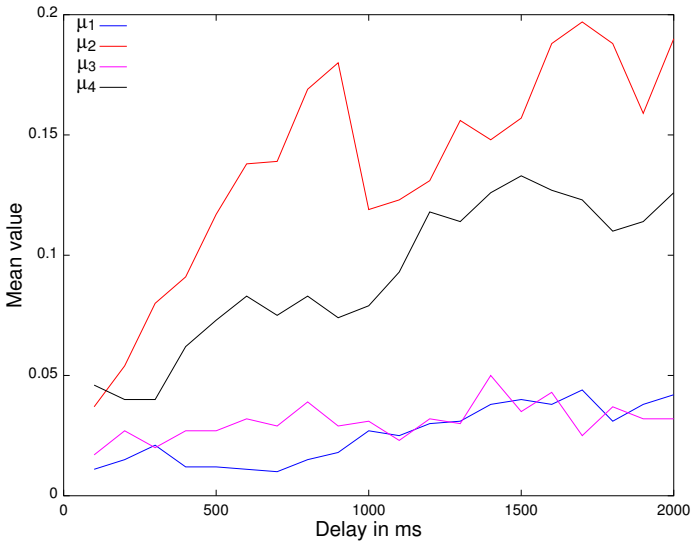


Figure 5.11 – Evolution of mean values of embodied features defined in Tab. 5.3. $\overline{F}_5$ in not considered since only one object was salient.

Discussion

The results obtained in the first experiment revealed that the walk primitive executes with distinct efficiency the motion requests along individual DOFs. Furthermore, a certain level on coupling is present, so the correction along one motion component may disturb convergence on others. In relation to the second experiment it was observed that increasing delay affects the quality of the anticipation of the localization. As shown in Fig. 5.10 the highest discrepancy is observed at the mean heading μ_ϕ, which has to do with the fact that the observation ${}^G\hat{\zeta}_\phi$ is also the noisiest (see Fig. 4.10). The quality of the features $\overline{F}_1$, $\overline{F}_2$, and $\overline{F}_4$ is also affected (see Fig. 5.11) since they are based on the prediction of the localization. Contrarily, $\overline{F}_3$ and $\overline{F}_5$ are related to visual characteristics of the scene (e.g. the morphology of the object, and the topological arrangements between salient stimuli).

CS-II: Simulation of object redundancy

The objective of this case study is to evaluate in simulations whether the robot is able to do visual selection to approach the object of interest, by discriminating consistent information in relation to the task context. For this, a scene is designed in Webots where many cans are disposed in the visual field of the robot, so it has to approach a particular one while ignoring the others. The desired configuration is specified by positioning the robot in front of the desired can.

Experiments

In the first experiment a static policy named $\mathrm{BN_f}$ is considered for the BN. Thus, the information provided by B_i to the discriminative process are heuristically fixed, so $p(\mathrm{B}_i|\mathrm{Object}) = p(\neg\mathrm{B}_i|\neg\mathrm{Object}) = 0.9$ (i.e., the probability that the feature B_i is true given the object is true, and the probability that the feature B_i is false given that the

object is false). The task is repeated at 4 initial locations under 26 delay profiles (from 100 ms until 2400 ms, increasing 100 ms; and at 3000 and 4000 ms), so a total of 104 trials are evaluated.

The second experiment is designed to verify whether the motion prediction is actually needed for the discriminative process. For exploring this issue, the features $\overline{F_1}$, $\overline{F_2}$, and $\overline{F_4}$ (see Tab. 5.3) are redefined in order to consider the last observation of the localization instead of the predicted localization. The $\mathrm{BN_f}$ policy is also employed. The task is repeated at 4 initial locations under 15 delay profiles (from 100 ms until 1500 ms, increasing 100 ms), so a total of 60 trials are evaluated.

In the third experiment dynamic policies are investigated for determining in runtime the contributions of the nodes B_i to the discriminative process. Thereby, a "pessimistic" policy $\mathrm{BN_{d1}}$ attempts to reduce the false positives (i.e. the error of type $\mathrm{T_1}$ of accepting an observation that does not belong to the object), by updating only the probability distribution $p(B_i|\neg\mathrm{Object})$ (see Eq. (5.11)). An "optimistic" policy $\mathrm{BN_{d2}}$ aims at reducing the false negatives (i.e. the error type $\mathrm{T_2}$ so an observation that is related to the object is rejected), by updating only the probability distribution $p(B_i|\mathrm{Object})$. A hybrid policy $\mathrm{BN_{d3}}$ attempts to reduce both types of errors. A time window of size $k = 3$ and a discount factor $\gamma = 0.7$ (see Eq. (5.10)) are set. In order to compare the policies, and given the fact that the observation of the features $\bar{F}_i$ is indeed a process of symbolization at three categories (or levels or intensity, see Eq. (5.14)); the observations O_i registered in the first experiment are evaluated with the dynamic policies.

Results

Figure 5.12 presents the trajectory followed from two trials of the first experiment. Despite many soda cans were placed on the scene, the agent was able to systematically approach the desired one while ignoring the others, until a delay profile of 2000 ms. Above this, it occasionally switched attention to the neighbor can.

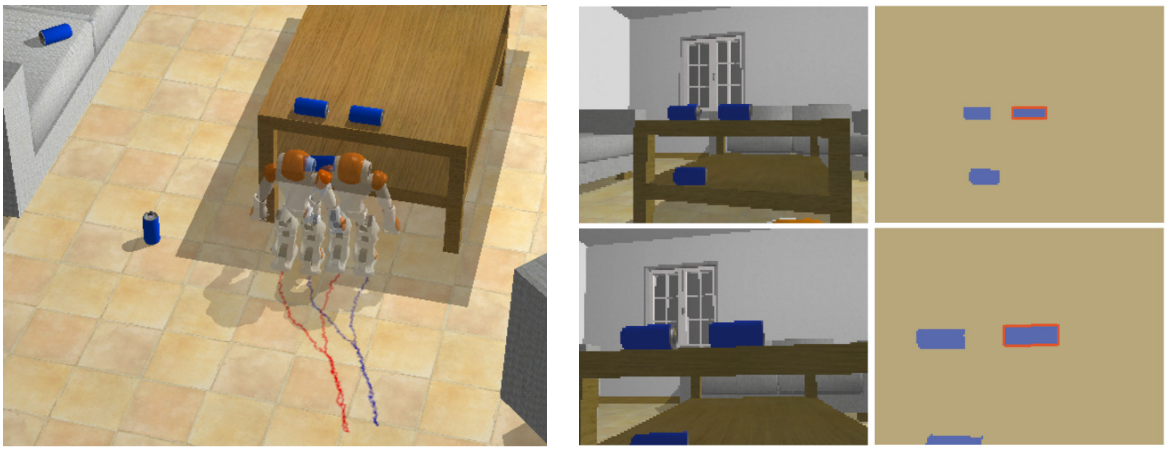


Figure 5.12 – Simulation of the task with object redundancy. On the left, starting from the same position the agent was required to approach a distinct can over the table (the resulting trajectories are superimposed). Some on-board views and the corresponding segmentations are shown on the right. Many blobs were detected, the one selected is highlighted in orange.

In relation to the second experiment (i.e. the last observation is used instead of the motion prediction to anticipate the next state) it was observed that the task could be

accomplished until a delay profile of 1100 ms. Notice that the mean walk velocity of the robot is approximately $\tilde{\mathbf{v}} = [0.022 \text{ m/s } 0.04 \text{ m/s } 0.106 \text{ rad/s}]^{\text{t}}$.

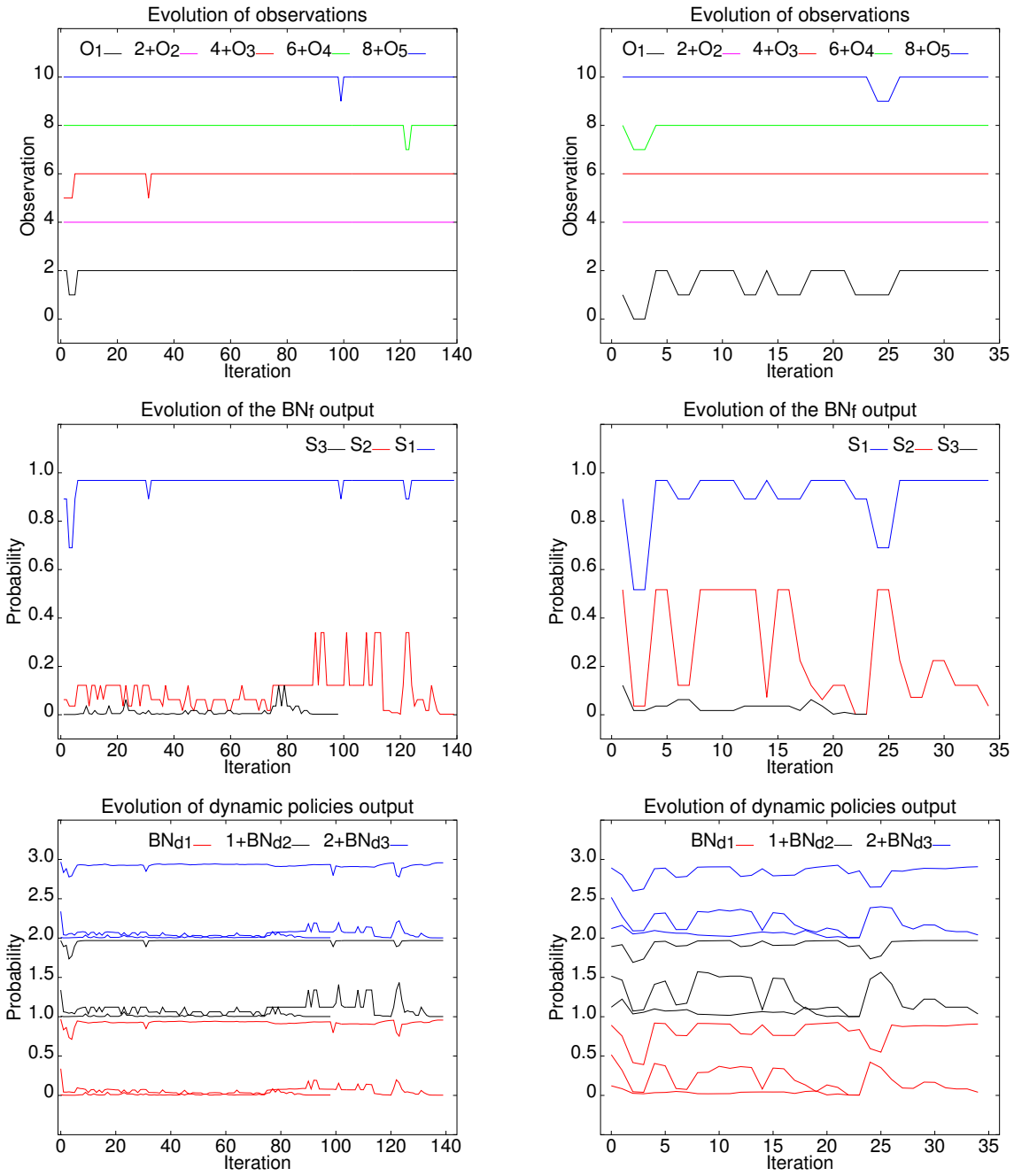


Figure 5.13 – Comparison between the network policies under two delay profiles. The left column presents the results for a 300 ms delay. The right column corresponds to a 1700 ms delay. The first row shows the evolution of the observations O_i as defined in Eq. (5.14), with 0 the lowest and 2 the highest intensity. The signal are shifted vertically for visualization. The second row shows the output of the network for the policy BN_f. The tracked can over the table is represented by s_1, and s_2 corresponds to the lateral neighbor (see Fig. 5.12). The third row presents the comparison between the outputs of the dynamic policies. The signals are also shifted vertically for visualization. The discriminative power on the left column is higher since distinct mean and standard deviation (see Tab. 5.6) are considered for defining the partition intervals in Eq. (5.13)

Figure 5.13 presents a comparison on the third experiment for two delay profiles. The

column on the left shows the results for 300 ms delay, whereas at the right the delay was set to 1700 ms. The plots on each column are related to the case of Fig. 5.12 where the robot was requested to approach the rightmost can on the top of the table. The registered feature intensities are shown in the first row (the desired value is the maximal intensity of 2). A comparison on the classification obtained by the static and dynamic versions of the BN is shown in the second and third rows. The discriminative power of the BNs at the right column was reduced at about the 3^{th} iteration, probably because of the minimal intensity detected in O_1 (that is related to the localization prediction). Also, around the iteration 25 the disappearance of the third can on the field of vision disturbed O_4 (that is related to the topology of the saliency). Subtitle differences were observed among the BN policies, so equivalent selections were made. However, as seen in Tab. 5.7, where some iterations from another trial are compared at 2200 ms delay, under this condition the policies differed, so only $\mathrm{BN_{d2}}$ and $\mathrm{BN_{d3}}$ were able to select the correct blob. Extra simulations where performed using these policies in the same scenario. The task could be successfully accomplished by $\mathrm{BN_{d2}}$ until a delay profile of 4000 ms.

Iteration	Blob	O_1	O_2	O_3	O_4	O_5	$\mathrm{BN_f}$	$\mathrm{BN_{d1}}$	$\mathrm{BN_{d2}}$	$\mathrm{BN_{d3}}$
3	s_0	0	2	2	0	2	0.339	0.269	0.534	0.451
	s_1	0	2	2	0	0	0.062	0.045	0.129	0.096
4	s_0	0	2	2	0	2	0.339	0.225	0.666	0.529
	s_1	0	2	2	0	0	0.062	0.036	0.205	0.127
5	s_0	2	2	2	0	2	0.799	0.616	0.883	0.754
	s_1	2	2	2	0	0	0.339	0.172	0.495	0.285
6	s_0	2	1	2	0	2	0.517	0.433	0.694	0.618
	s_1	2	1	2	0	0	0.122	0.090	0.227	0.173
7	s_0	0	2	2	0	2	0.339	0.252	0.604	0.500
	s_1	1	2	2	2	0	0.517	0.379	0.447	0.315

Table 5.7 – Analysis of 5 iterations of an experiment where two blobs were salient. The delay profile corresponded to 2200 ms. No threshold was fixed as a minimum requirement for acceptance, so that the most likely blob was selected. In the experiment the agent originally applied the fixed policy $\mathrm{BN_f}$, and failed the approach by switching attention at the seventh iteration. The decisions based on observations O_i are simulated for the other policies. The dynamic versions considered a sliding window size $k = 3$, and a discount factor $\gamma = 0.7$ (see Eq. (5.10)). The cells in blue correspond to the correct selection at each iteration, whereas the cells in red correspond to the wrong selection. The policy $\mathrm{BN_f}$ selected s_1 in iteration 7 simply because it has an additional observation of level 1 for O_1. The policy $\mathrm{BN_{d1}}$ has mostly valued the fact that the last observations O_4 and O_5 were of minimal intensity for the rejected candidates, thus, the features were considered as good discriminants, and s_1 was assigned the highest probability. As noted, only the dynamic policies that attempted to reduce the error type $\mathrm{T_2}$ were able to chose the correct candidate at the last iteration. These policies have valued the contributions of O_3 and O_5, that were of maximal intensity for the winner blob at all the iterations.

Discussion

The results obtained in the first experiment suggested that the behavior scheme implemented is able to produce the desired comportment at very high delay profiles. In this sense, the information redundancy in the sensory-motor coordination provided the agent with the means to anticipate the evolution of the features, so discriminating the

object in the scene without relying on ubiquitous knowledge about the environment (e.g. a localization map). Furthermore, the dynamic analysis of the agent's locomotion resulted in aesthetic trajectories that mimicked the human walk style. Despite this aspect is a non-functional requirement, it is of crucial importance for the acceptance of the solution in a human-machine interaction context. The trajectories obtained were also more efficient that the ones in the study of Sec. 4.6, since the robot walked more in the sagittal plane direction and had less difficulty to converge to the desired location once close to the object; which is a clear improvement.

In the first experiment it was also observed that the discriminative power of the network is conditioned to the delay profile employed (see Fig. 5.13). This is reasonable since when using predictive models, the more delay involved, the more uncertainty. In addition, the predictive model considered was fairly simple (i.e. a deterministic continuous motion assumption).

The second experiment revealed that the statistical regularities induced by the coupling can be conveniently exploited to assist the discrimination of the object, so the criteria for anticipation used was simply the last observed context. However, a model-less version of the task would not ensure the conversion in the proximity of the object (since at short distances the blob's size would increase considerably, so it easily leaves the field of vision). As detailed in Sec. 4.4.2, the Look-at task performs a predictive gaze that is useful to relocate the object.

The third experiment showed that the differences between the BN policies are subtle until a delay profile 2000 ms. However, as the comparison presented in Tab. 5.7 revealed, the dynamic policies $\mathrm{BN_{d2}}$ and $\mathrm{BN_{d3}}$ that aimed at reducing the error type $\mathrm{T_2}$ were more robust under high delays. These policies were sensitive to the fact that pure visual information was consistent despite the high delays, so more weight was assigned to the corresponding features. In fact, in several additional trials it was verified that the best results are obtained from the "optimistic" policy $\mathrm{BN_{d2}}$.

CS-III: Approaching a real can

The objective of this case study is to evaluate the full implementation of the hybrid architecture (see Sec. 5.5.3) in a real task. The experiments are conducted with Nao, in an unstructured environment under natural and artificial illumination. The desired pose in relation to a multicolor tea can is shown to the robot by pressing the head tactile sensor. The robot is then moved away from the desired configuration, so it has to return as close as possible to the demonstrated position.

The parameters of the system were adjusted for execution in the real platform. The estimation of the delay profile relied on the measurement of the time required for the simplest case of the perceptive loop, that is, when the object is already centered on the field of vision, so a single iteration of the Emergence task is needed. The programs were firstly compiled to run natively on the robot, though the average delay obtained was too high (with mean $\mu = 2736.5$ ms, and standard deviation $\sigma = 573.3$ ms). Therefore, the control programs are executed remotely, by retrieving the visual and the proprioceptive data from the robot though a wireless connection. Under this condition, the mean delay obtained was $\mu = 811.1$ ms with standard deviation $\sigma = 373.6$ ms. Since the Look-at task operates in closed-loop, the number of iterations required to center the blob is limited to a maximum of two, such that the expected delay is 1700 ms.

Experiments

Two experimental scenarios are considered. In the first one, a single tea can is placed in the scene so the robot has to approach it while ignoring other saliency emerging in the task. In a more challenging scenario, two cans were placed one beside the other at a distance around 4 cm, and the agent is requested to approach one of them. Two motion modalities are also considered. In the step-by-step motion policy (as in Sec. 4.6.4), the Walk task waits until the robot stops before sending additional commands to the motion primitive. In the continuous motion modality the robot walks fluently towards the object (as explained in Sec. 5.5.2). Finally, two control modalities for the runtime execution are also evaluated. In the off-line modality the task is unsupervised, that is, the thresholds ϵ_1 and ϵ_2 (see (5.17)) for the Deliberative process are set to zero (which is equivalent to let the Behavior module to run uninterruptedly until either convergence is obtained or the object is lost). In the on-line modality the thresholds are activated ($\epsilon_1 = 0.6$ and $\epsilon_2 = 0.2$) so the Deliberative module can resort to Remote Processing when the confidence in the task progress is low. Thereby, a total of 2 (scenarios) $\times$ 2 (motion styles) $\times$ 2 (control modalities) = 8 experimental cases are designed. Each case is evaluated at 10 distinct initial locations.

Results

The results for the cases designed are presented in Tab. 5.8. The apparent precision of the system (no ground truth was measured) is illustrated in Fig. 5.14. As the images suggest the robot was able to converge to a very similar location. Some of the trajectory followed are given in Fig. 5.16. Some on-board views are shown In Fig. 5.17.

Id	Case	Successes/trials	Supervision
1	Off-line/One can/Step-by-step	9/10	NA
2	Off-line/One can/Continuous	7/10	NA
3	Off-line/Two cans/Step-by-step	7/10	NA
4	Off-line/Two cans/Continuous	5/10	NA
5	On-line/One can/Step-by-step	10/10	1
6	On-line/One can/Continuous	10/10	3
7	On-line/Two cans/Step-by-step	10/10	4
8	On-line/Two cans/Continuous	10/10	5

Table 5.8 – CS-III experimental cases results. The column Supervision indicates the number of trials where the Remote Processing was activated. The other column headers are self explanatory. NA denotes non-available data.

Discussion

In the results reported in Tab. 5.8 a first aspect to be noticed is the fact that the off-line execution of the task produced lower success rates. Several reasons can explain the failures obtained. When step-by-step motion was employed, though more success trials were registered, momentary degradation of the saliency detection made the robot to rotate to the wrong direction, so it lost the view of the can. Depending on the view perspective and the head's motion, the cans were eventually merged in the saliency detection, this is illustrated in Fig. 5.15. A workaround would be to pre-evaluate the images as to ignore

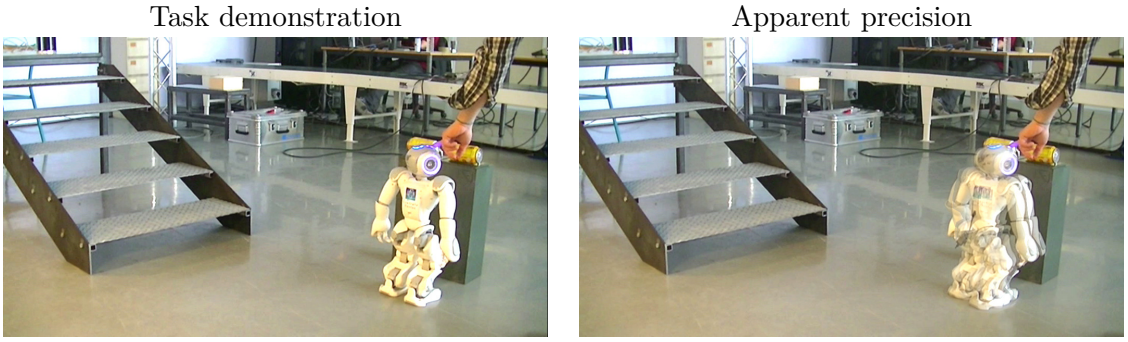


Figure 5.14 – On the left the demonstration of the desired pose in relation to the can. On the right the final position of some trials are superimposed.

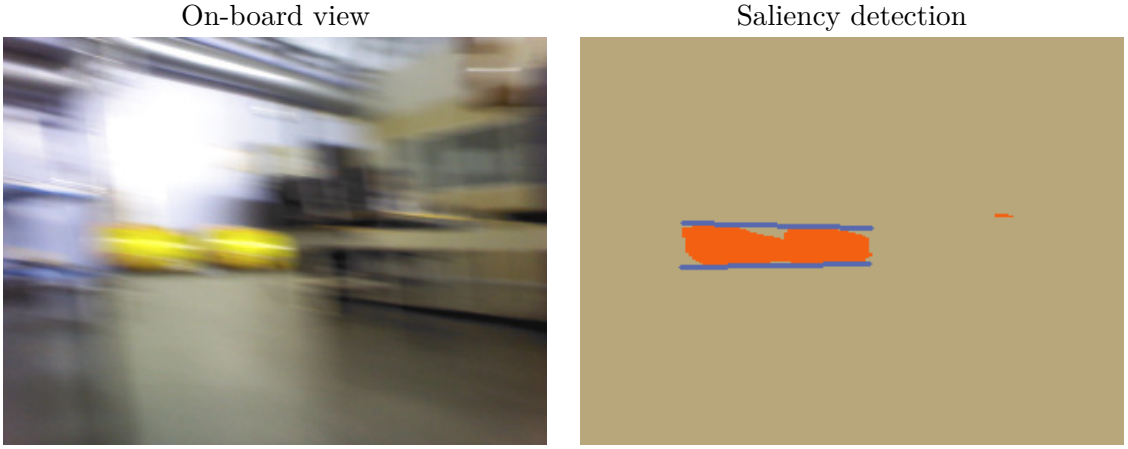


Figure 5.15 – On-board view of the experimental modality of continuous motion. Saliency degradation due to the view angle to the cans and the motion blur produced.

noisy captures. Though, in the experimental platform a new acquisition would have an extra time cost of 800 ms, which prevented the adoption of this solution.

The lower success rates in the continuous motion trials can be explained by unexpected peaks in the feedback delay (e.g. the robot exceeded the delay profile in more than 2000 ms), which affected the precision of the prediction. Also, a less accurate performance of the walk primitive was observed under continuous motion. Other source of perturbations were unexpected variations in the scene topology due to motion at the background (e.g. the orange robot arm in front moved during some trials, and people walked in front of the robot, see Fig. 5.17).

In the on-line trials the performance of the network was very consistent. The threshold established for the estimate of the degree of confidence and the discriminative power, were sufficient to produce the transition to the Remote Assistance state in the Deliberation process (see Fig. 4.7). Thus, the user either suggested a search direction to re-locate the object when it was lost, or clicked above the correct blob in the image when the certainty was low. This is reflected in the column Supervision of Tab. 5.8, so the maximal success rate is obtained through the remote assistance in the task solution.

Finally, the same approach used in Sec. 5.5.6 was adopted in order to compare the distinct BN policies. That is, the observations registered under the BN_f policy were simulated with the dynamic policies. Therefore, the interest was to verify whether the dynamic versions would have chosen the correct can once the attention shifts were produced in the

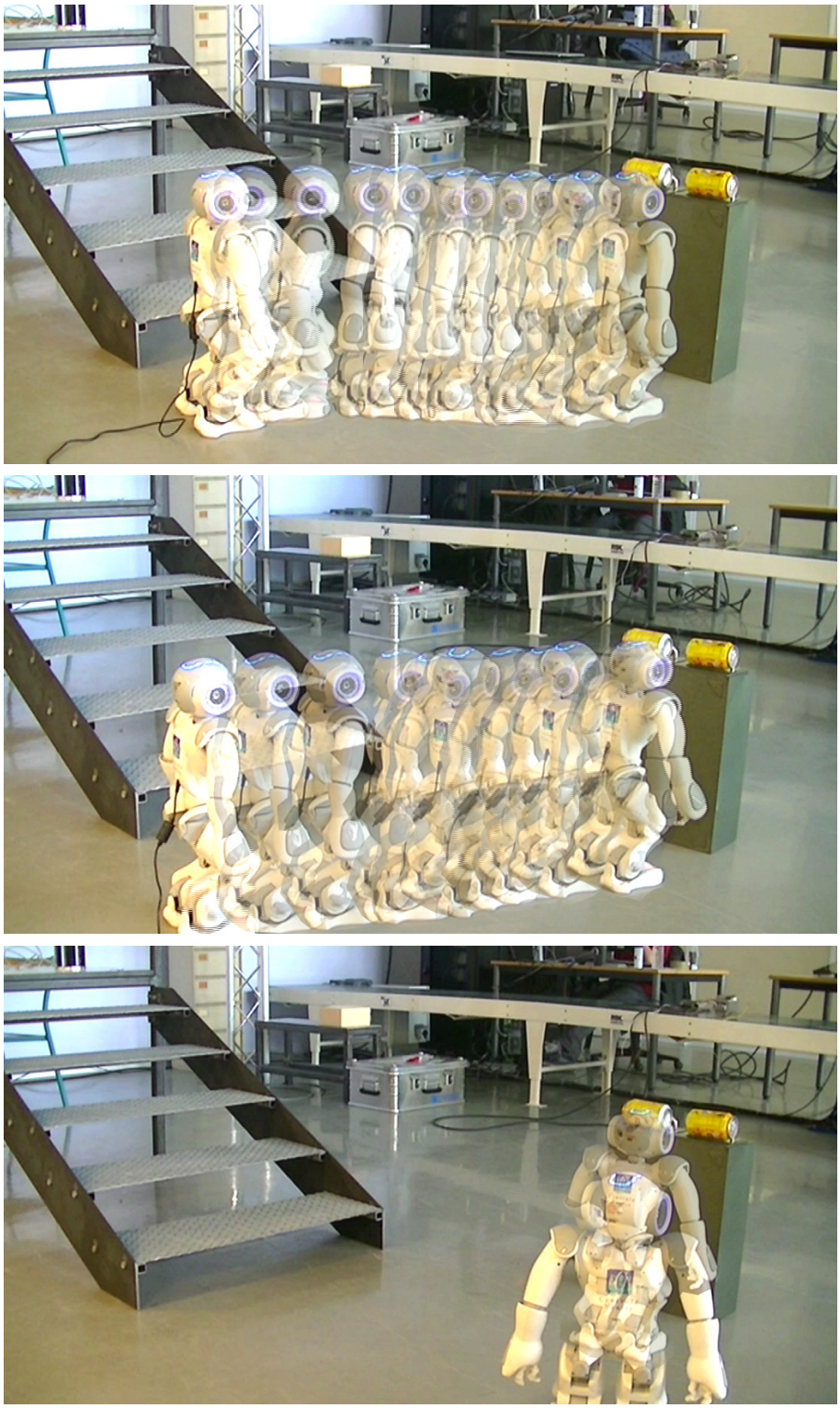


Figure 5.16 – Experimental evaluation with object redundancy. The sequences for three trials are shown. The robot was required to approach the can on the right.

off-line trials. Unfortunately, no significant advantages could be found in the dynamic versions for the real experiments. As discussed previously, in view of degradations in the visual saliency and unexpected variations in the scene topology, the consistency of pure visual information could not be ensured.

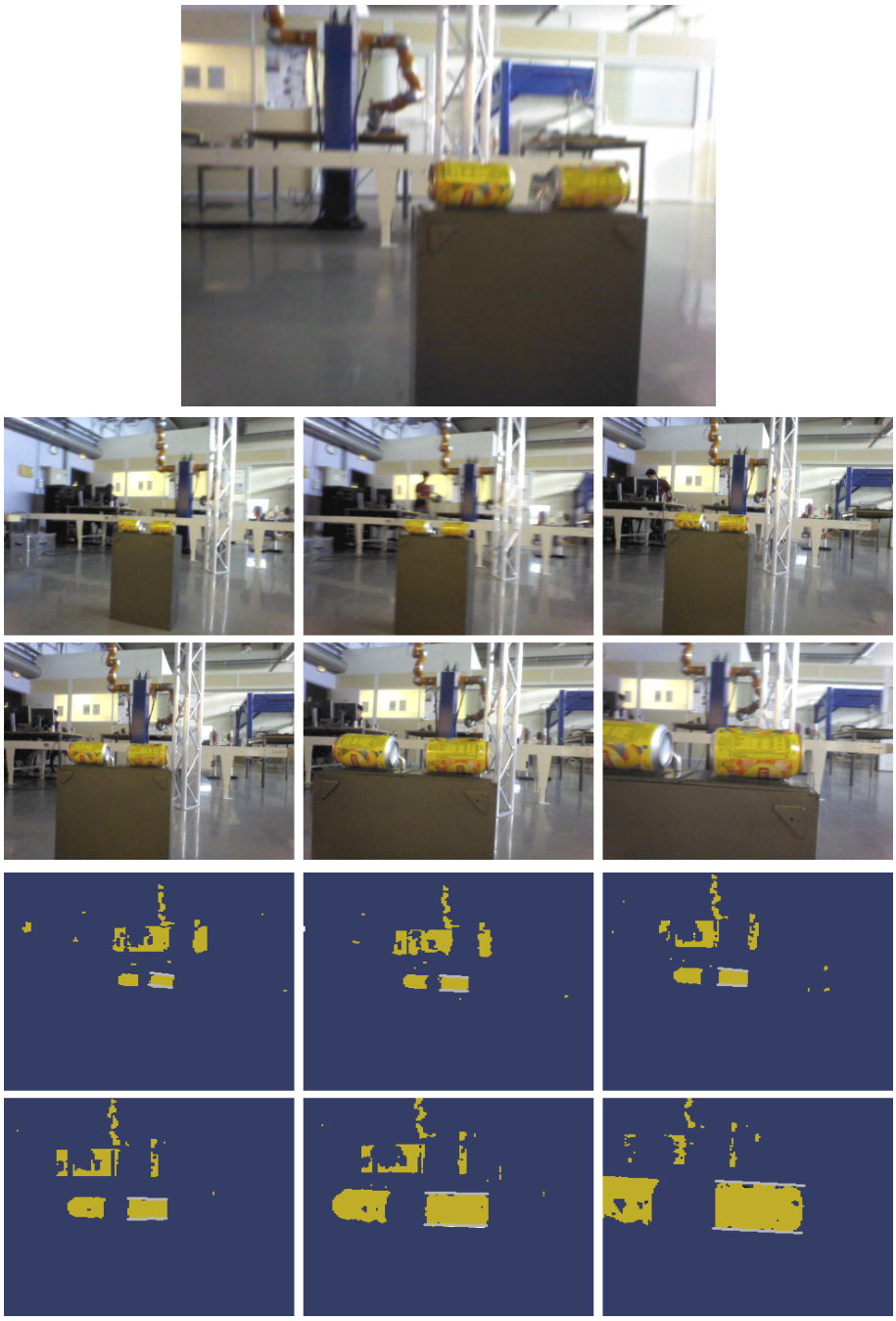


Figure 5.17 – On-board views of the experimental task. In the first row the view of a frame is enlarged. As it can be seen, the scene is illuminated irregularly given the presence of big windows in one of the walls of the lab. In the intermediate row an approach sequence is shown. On the bottom the corresponding saliency is presented.

Designing reliable approach tasks in six-steps

Throughout this chapter two main topics related to the locomotion guided by vision were investigated. One is the behavioral dimension of the task. Here the efficient use of information emerging in the coupling was studied, in order to design the behavior scheme for controlling the robot so mimicking the human motion style. The other topic is situated at a meta-behavioral level of concern, where the robot is viewed as a limited resource system. Thus, the safety and the robustness of the solution were analyzed in the

hybrid architecture design, that included the possibility of integrating remote processing to the solution. These developments can be organized in a methodological proposal that orderly exposes the steps to be followed, either to replicate the reported results, or to design solutions to other sensory-motor tasks.

Thereby, reliable humanoid object approaching can be obtained by applying the following six steps:

1. *Studying the behavior as emergent.*
2. *Defining embodied features related to the task.*
3. *Anticipating the context from a predictive model.*
4. *Relating actual measurements with the anticipation.*
5. *Doing attention selection.*
6. *Evaluating the task consistency.*

The **first** step involves the use of the EC research methodology (see Sec. 5.3) to analyze the task at hand as a dynamic system, such that identifying efficient control parameters. Thus, it is crucial to consider the task from a first-person perspective and rigorously restrict modeling.

In the **second** step, starting from the sensory resources available and the characteristics of the task, a set of features providing information about the context of the task are designed. In this study the visual and proprioceptive sensory modalities where considered (see Tab. 5.2), though other sensory modalities (e.g. acoustic, inertial, etc.) could be included.

In the **third** step a predictive model is employed to anticipate the measurements for the next execution cycle. A deterministic model was considered in this study, but other approaches are available (e.g. probabilistic models in Thrun [178]). In case the acquisition and control rate are high enough, by exploiting the statistical regularities induced in the sensory-motor coupling, the next state can be anticipated from the current state (i.e. no motion prediction would be required).

In the **fourth** step a set of variables is defined to conveniently relate the embodied features measurement with the anticipation (see Tab. 5.3). In this study discretization through clustering was applied to the measurements, so they are classified into levels of intensity. This is due to the fact that the algorithm used for attention selection (the BN) is discrete.

In the **fifth** step the attention selection model is employed. In this study the spatial congruence is considered (see the spotlight metaphor in Sec. 3.2.2), but other endogenous criteria can also be used (e.g. the opportunity for actions or the *affordance* of stimuli, see Horton et al. [87]). In addition, as mentioned before, the selection occurs in a Bayesian network, although other frameworks are available in the machine learning literature (e.g. neural networks, support vector machines, among others).

Finally, in the **sixth** step the consistency of the task is evaluated through a probabilistic criteria. In this study the discriminative power of the attention selection mechanism and the anticipation congruence were proposed, but other criteria could be developed. This step also involves the design of a deliberative transition model (e.g., see Fig. 5.9).

Conclusions

This chapter has focused on the development of more realistic solutions to the problem of approaching and positioning in relation to objects on the environment based on vision. From the analysis of the dynamic aspects of human locomotion guided by vision, the first-order description proposed allowed the agent to mimic the human walking style. This is of crucial importance since the resultant behavior is more efficient and aesthetic, which are valued aspects for the acceptance of the solution in the context of human-machine interaction and service robotics applications.

The methodology proposed to design reliable solutions illustrated an interesting combination between the cognitivist and the EC research. In this sense, the attention selection mechanism was inspired by the spotlight metaphor, which is an information processing model of attention. Moreover, Bayesian networks are usually employed for information fusion and knowledge representation in applications related to the cognitivist AI research (e.g. probabilistic diagnosing). However, in the network structure designed, multi-sensory information is fused from features that exploited embodiment, so they were carefully defined from the EC perspective. The anticipative aspect of the behavior scheme was also an interesting opportunity to study the effect of the statistical regularities induced by the coupling, and the information redundancy in the sensory-motor coordination.

The results obtained in the case studies suggested that the BN is a convenient and easy to use technique, which produced reliable information about the degree of confidence and the discriminative power of the attention selection mechanism. This consisted in a significant contribution to the autonomy of the agent through the efficient use of available resources, where the solution was operational at high delay profiles with a low-cost robot. Furthermore, the designed hybrid architecture ensured that remote resources could be used in a safe way, and opened the possibility for enriching the local behavior repertory, so constituting a distributed and extensible solution for the task.

In general, the studies conducted in this chapter illustrated a potential and feasible strategy that can be adopted for prototyping and exploring more complex sensory-motor coordinations. The fact of counting on modular motion primitives that are already available to the agent, handles much of the security aspects of the task, as for example, maintaining the body balance. Therefore, the possibilities of exploring other behaviors schemes seem to be vast, and perhaps require of less modeling efforts than the pure cognitivist definition of the task. Thus, more skills can be organized into behavior architectures. This aspect is investigated in the next chapter through the action selection problem, where reactive motion and learning is considered for approaching the object and avoiding obstacles.

6

Reactive walking

Contents

6.1 Introduction 125
6.2 Related work 126
6.3 The framework iB2C 128
6.3.1 Model components 128
6.3.2 Behavior coordination 130
6.3.3 The sequence node extension 131
6.4 Reinforcement Learning 132
6.5 Case studies 134
6.5.1 Materials and resources 135
6.5.2 Behavior-based models implementation 135
6.5.3 CS-I: Action-oriented approach 136
6.5.4 CS-II: Object approach and obstacle avoidance 144
6.5.5 CS-III: Learning-based approach 152
6.6 Conclusions 161

Introduction

The two previous chapters have dealt with different aspects of the problem of visually guided walking under relatively favorable circumstances. This was the case since the access to the object of interest was free from obstacles. In service robotics more difficult scenarios may be encountered, so obstacles must be contoured before reaching the object. For this, different objectives may be achieved (e.g. steering to the object, and avoiding inconvenient locations). Although the behavior scheme presented in Sec. 5.3.3 included the parallel execution of the Walk and the Look-at tasks, it did not evoke the problem of concurrent access of available resources. As discussed in Sec. 2.5, this is known by the

Action Selection Problem (ASP), which is a difficult problem from the control and the AI point of view, given the uncertainties on the task.

In this chapter the central topics of concern are the study of embodiment, knowledge representation, and learning; under different ASP scenarios. By keeping the first-person perspective adopted throughout the work, and the proposal of distributed representations of the task; a behavior-based framework is selected to model concurrency, so emergent behavior is produced. Three case studies are developed. In the first one, from the concurrent access to the walk primitive of the robot, different walk modes are proposed so the approaching to the object is reactive and based on action-oriented representations of the object (differently from the previous chapters where it was based on a 3D model of the object). In the second study, the behaviors of object approaching and obstacle avoidance are studied in the model, so the agent has to bypass obstacles to reach the object, without using a global representation of the scene. In the third study, visual encoding is proposed as an embodied description of the task, so more efficient solutions can be learned.

Related work

Behavior-based architectures are conceptualized from Reactive models (Mataric [116]). Though behaviors are given a larger connotation than merely reflexive actions, so they may also refer to learned skills including a state representation (see Sec. 2.5.4). Thus, the scope of behavior-based architectures covers the reactive model and hybrid models of the types *managerial* and *state-hierarchy* (Murphy [131]). There are numerous architectures reported in the literature that would fall into this range. A detailed exploration of available models is out of the scope of this work. Next, some contributions are briefly discussed by focusing on the aspects of the model structure and the strategy for behavior selection.

A hierarchical organization of behavior has been proposed by Brooks [27] as a bottom-up methodological design principle for studying the task at hand. This has influenced many architecture designs. Thus, in Burghart et al. [31] a three-layer hierarchy is proposed, with the low-level containing fast interpretation methods of sensor data, the middle-level layer containing various recognition components of the system having access to persistence, and the highest layer providing multimodal fusion and situation recognition. In Lenser et al. [107] sensor, motor, and control hierarchies are distinguished. The sensor hierarchy represents the knowledge that the robot has about the world. Sensors are classified as real (i.e. provided by hardware) or virtual (i.e. processed information from real sensors). The behaviors are organized according to the complexity, so the control hierarchy buffers the communication between different behavior levels. An important aspect to be noticed is that the frequency of sensors, behaviors, and control processes decrease when moving up in the hierarchy.

Different approaches have been followed to do behavior selection. In the work by Conde et al. [50] fuzzy logic is employed for establishing a correspondence between detected events and the weighted contribution of behaviors. Fujita et al. [71] have proposed the evaluation of external and internal drives to determine the behavioral mode of the agent. Thus, the *homeostasis regulation rule* for action selection is employed (see Arkin et al. [9]). For this, the control system has to evaluate the potential activation of the behavior in relation to the current situation. This is also close to Minsky's [123] ideas where meta-knowledge about a process (e.g. preconditions, effects on the system, and

postconditions after successful execution) is considered to reduce the difference between the system's current state and a goal state.

Case-based reasoning (CBR) has also been used for behavior selection. According to Kolodner [100], in CBR old experiences are used to understand and solve new problems. Under this approach a work by El-Bagoury et al. [61] has proposed a hierarchical case-based controller for the robot Nao, for the distinct situations (or roles) taking place in the Robocup soccer league. Liu & Hitoshi [110] have resorted to *genetic programming* in a simplified simulation of the task to identify high-level decisions, and then to CBR as an on-line adaptation means for obtaining low-level decisions in real world environments. In these studies the knowledge required to do action selection is explicitly provided to the model. CBR would alleviate these efforts by defining a core case set that can be generalized to the other situations encountered. It is also possible to obtain this knowledge automatically. This aspect is treated in Sec. 6.4 where reinforcement learning is discussed.

Given its wide use, and the fact that Nao is a standard platform for the RoboCup SPL competition, several architectures have been proposed to control this robot. In this sense, in Ferland [69] the *hybrid behavior-based architecture* (HBBA) was implemented to provide learning and sharing past experiences related to episodic memory. In Testart et al. [174] the functionalities are organized in four parallel modules (i.e. perception, actuation, world-modeling, and hybrid control) for applications in the soccer competition. In Niemüller et al. [134] a behavior engine was developed to provide the functionalities of the skill level (in a three-layer hierarchy), relying on a *hybrid state machines* implementation. In Agüero et al. [2] tree-graph representations is proposed to hierarchically organize the activation of behavior in runtime. There are certainly many other works that could be mentioned, so an important question to be asked is: which option is the more adequate for the current study? To answer this question some important criteria are considered, such that: the relevance to the research context, the availability of the software, and the usability (e.g. the learning curve).

Hawes and Wyatt [81] have proposed a useful classification for robot architectures that can base the discussion on their relevance to the current study. Three level of abstraction are identified. The more general level corresponds to a *computational architecture* (CA), where a structure to process information is described without a specific problem in mind. At a less general level, *instantiated information-processing architectures* (IIpA) are proposed in a specific problem domain (most of the previously discussed works would fall into this category). In the lowest level of abstraction are *software architecture* (SA), that consist in a concrete implementation under a hardware and software platform. By definition, behavior-based architectures are embodied so they are tied to the particular domain of the task. Thus, the decision to reuse an architecture to similar tasks has to be judiciously taken. Even at the more favorable scenario (i.e. to change only the host robotic platform by keeping the same virtualization hierarchy, hoping that the robot bodies are similar enough) adaptations would be required to adjust the bottom layer to the available sensory and motor equipment. In case the task would also change, more adjustments would be required. For instance, a RoboCup model would consider events in the context of a football match (e.g. a ball pass, an opponent attack, etc.) so it can hardly be used for an application in robot navigation. Similarly, architectures for wheeled outdoor navigation would include a series of functionalities (e.g. cartography, communication, etc.) relying on super-human sensory, which may be irrelevant to a biped locomotion case study. The learning curve of these models is generally slow, so considerable efforts may be invested to master the conceptualization of the model, and to proceed later with major

customizations. Indeed, perhaps this would explain the diversity of models reported in the literature. Since the *integrated behavior-based control* (iB2C) framework is a CA, it is selected to develop the case studies. The main characteristics of iB2C are discussed next.

The framework iB2C

The iB2C by Proetzsch et al. [152] defines a set of architectural design principles that provide support for several behavior-based mechanisms, such that, coordination, behavior interaction, and hierarchical abstraction. Many practical applications have been developed within the framework. Such is the case of wheeled navigation and exploration on rough off-road terrain (e.g. in Proetzsch et al. [151], and Armbrust et al. [11]), the control of a humanoid head for interaction using emotional states (Berns & Hirth [18]), and indoor service task in home and office environments (Schmidt et al. [162]); among others. In the following, the mathematical formalism of a iB2C model is detailed.

Model components

The fundamental unit of the framework is the behavior module (see Fig. 6.1), which is an atomic wrapper around a specific task. Thus, a behavior B is a container providing a uniform interface to diverse functionalities of the system. It is defined as a three-tuple, such that

$$B = (F, f_a, f_r), \tag{6.1}$$

where f_a is the activity function, f_r is the target rating function, and F is the transfer function. Table 6.1 describes the inputs and outputs of a behavior.

The activity ϑ (or effective relevance) of B in the network at a given moment depends on the stimulation $\mathbf{s}$ and the inhibition $\mathbf{i}$, received from other nodes. It is defined by

$$\vartheta = \mathbf{s} \cdot (1 - \mathbf{i}). \tag{6.2}$$

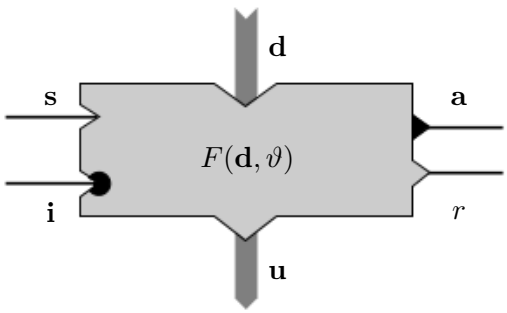


Figure 6.1 – Basic iB2C behavior module (Proetzsch et al. [152]).

Var	I/O	Definition
d	input	The input data $\mathbf{d} \in \Re^m$ can contain sensory data (e.g. joint positions, image measurements) or information from other behaviors (e.g. their target rating)
s	input	A behavior can be stimulated by k others such that $\mathbf{s} \in [0,1]^k$ is the intended relevance of B_k. In case $\mathbf{s}_k = 0$ indicates no stimulation and $\mathbf{s}_k = 1$ a fully stimulation from behavior k. Values between 0 and 1 refer to a partial stimulation.
i	input	Inhibition has the inverse effect of stimulation. Each behavior can be inhibited by k other via $\mathbf{i} \in [0,1]^k$. Thus, $\mathbf{i}_k = 1$ refers to full inhibition and $\mathbf{i}_k = 0$ to no inhibition from behavior B_k.
a	output	The activity signal $\mathbf{a} \in [0,1]^k$ of a behavior B represents the amount of influence of B in the current state of the system. With $\mathbf{a}_k = 1$ all output values are intended to have highest impact, whereas $\mathbf{a}_k = 0$ indicates inactivity.
r	output	The behavior signal target rating $r \in [0,1]$ is an indicator for the contentment of a behavior. A value of $r = 0$ indicates that the behavior is satisfied with the actual state, while $r = 1$ shows maximal dissatisfaction.
u	output	Output data $\mathbf{u} \in \Re^n$ is generated by the behavior which can be used for actuator control or as input for other behaviors.

Table 6.1 – Definition of the input and output variables of a behavior.

The transfer function F provides the intelligence of the behavior. The output produced depends on the inputs received and the internal representation. This can be a reflexive response to an input, a more complex operation in the form of a state machine, or a sophisticated algorithm. The transfer function is defined by

$$F : \Re^m \times [0,1] \rightarrow \Re^n, \mathbf{u} = F(\mathbf{d}, \vartheta). \tag{6.3}$$

The activity function f_a of B is defined by

$$f_a : \Re^m \times [0,1] \rightarrow [0,1] \times [0,1]^k, \mathbf{a} = f_a(\mathbf{d}, \vartheta) = [a \ \underline{\mathbf{a}}]^{\mathrm{t}} \tag{6.4}$$

where $\underline{\mathbf{a}} = [\underline{a_1} \ \underline{a_2} \ ... \ \underline{a_k}]^{\mathrm{t}}$ are the derived activities, so the behavior can transfer part of its activity to other behaviors. Thus, a is the activation of B, and $\underline{a_i} \leq a \ \forall_i \in \{1, \ ... \ , k\}$ are the derived activities sent to the connected nodes.

The target rating function f_r depends on the characteristics of the task executed by B. In case of continuous state space representations, the normalized distance to the desired state is usually employed. It is important to point out that the fact of reaching the goal state does not necessarily mean that B is inactive (e.g. in on-line applications of motion imitation the behavior must never become inactive). Therefore, there is no direct influence on the activation of B and its target rate r.

As shown in Fig. 6.2, hierarchical abstraction can be defined in iB2C through a behavioral group, which embeds a collection of modules with a new interface, so externally it is viewed as a single behavior unit. Groups possess the same standardized interface illustrated in Fig. 6.1 and described in Tab. 6.1.

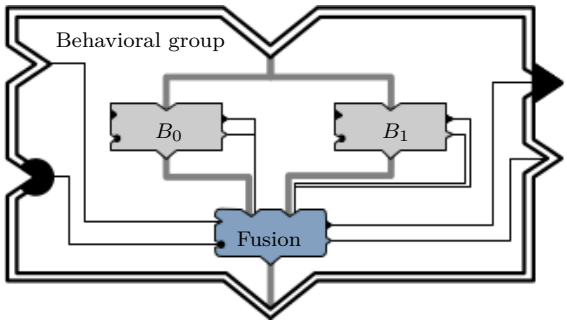


Figure 6.2 – Illustration of a group behavior interface (Proetzsch et al. [152]). Internally the outputs of two behaviors are combined in the fusion behavior represented in blue.

Behavior coordination

In the study of behavior coordination a distinction has been established (Pirjanian [146] and Hoffmann [83]) between arbitration and command fusion. In the former, one or various behaviors have control over the system resources for a period of time, that is, the actions of the selected behaviors are transferred without modifications. Contrarily, in command fusion the output is obtained by a combination of individual contributions. The framework defines a distinct type of node (represented in blue in Fig. 6.2) for command fusion. Though it shares the same interface of basic nodes. From the control inputs $\mathbf{a}$ and $\mathbf{r}$ received, the activity signal a and the rate signal r of the node must comply to the following conditions

$$\min_j(\mathbf{a}_j)\vartheta \leq a \leq \min\left(1, \sum_{j=1}^{k} \mathbf{a}_k\right)\vartheta, \tag{6.5}$$

$$\min_j(\mathbf{r}_j) \leq r \leq \max_j(\mathbf{r}_j). \tag{6.6}$$

Several fusion strategies are reported in the literature (e.g. *voting* in Rosenblatt [155], *fuzzy logic* in Saffiotti et al. [160], among others). Three simple criteria that can be employed are *maximum*, *weighted*, and *weighted sum* fusion. Let $\mathbf{u}_i$, a_i, r_i denote respectively the output $\mathbf{u}$, the activation a, and the target rating r produced by the i^{th} behavior connected to the fusion node. For maximum fusion the model components are defined such that

$$\mathbf{u} = \mathbf{u}_s, \;\; a = \mathbf{a}_s, \;\; r = r_s \tag{6.7}$$

where $s = \text{argmax}_c(a_c)$. For weighted fusion the model components are

$$\mathbf{u} = \left(\frac{\sum_{j=0}^{k} a_j \mathbf{u}_j}{\sum_{l=0}^{k} a_l}\right), \;\; a = \left(\frac{\sum_{j=0}^{k} a_j^2}{\sum_{l=0}^{k} a_l}\right)\vartheta, \;\; r = \left(\frac{\sum_{j=0}^{k} a_j r_j}{\sum_{l=0}^{k} a_l}\right). \tag{6.8}$$

For weighted sum fusion the model components are

$$\mathbf{u} = \left(\frac{\sum_{j=0}^{k} a_j \mathbf{u}_j}{\mathbf{a}_s} \right), \; a = \min \left(1, \sum_{j=0}^{k} \frac{a_j^2}{\mathbf{a}_s} \right) \vartheta, \; r = \left(\frac{\sum_{j=0}^{k} a_j r_j}{\sum_{l=0}^{k} a_l} \right) \tag{6.9}$$

In relation to behavior arbitration, perhaps the more commonly used criteria are priority- and state-based. Priority-based arbitration is originated from the research on subsumption architecture (Brooks [27]). Accordingly, the models consist of a set of behaviors forming a network of hardwired finite state machines. Action selection occurs when a higher-level competency (i.e. a more specific desired class of behaviors) overrides the output of lower-level ones. In state-based arbitration behaviors are selected based on their relevance to the current situation. There are many approaches available. In discrete event systems (e.g. Kosecka & Bajcsy [103]) the arbitration is based on the detection of events under a *finite state automata* model. The states correspond to the execution of actions/behaviors, and the events are observations and actions that cause transitions between the states. In *bayesian decision analysis* (Kristensen [104]) sensory operations are evaluated according to the cost/benefit of the information they provide. Other approach available is Reinforcement Learning, which is going to be discussed in Sec. 6.4.

The sequence node extension

Armbrust et al. [10] have proposed an extension to the architecture. As shown in Fig. 6.3, it consists in the inclusion of a new type of node for representing sequences, which is called *conditional behavior stimulator* (CBS). This node becomes active if certain conditions related to the activity or target rating inputs are met. Consequently, the connected nodes to the output ports can be also stimulated. Once active, a CBS monitors the values of a second set of its input ports. If the conditions concerning these values are fulfilled, the nodes activity goes down to zero again. Thus, arbitrarily complex behavior sequences can be created.

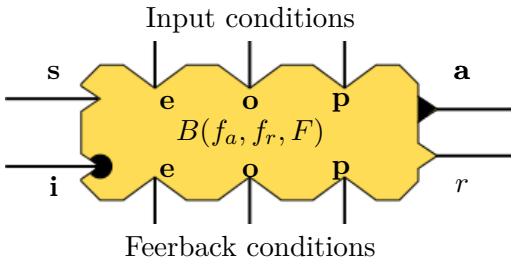


Figure 6.3 – Structure of the CBS module (Armbrust et al. [10]). Three different types of ports (Enabling, Ordering, and Permanent) for input conditions (top) and feedback conditions (below). As a CBS is a behaviour, it also features the standard behaviour ports.

A relation $\mathrm{ir}_j(t)$ occurring on time t, where the input value v_j is compared to a threshold ϵ_j, is denoted by

$$\mathrm{ir}_j(t) = \begin{cases} 1 & \mathrm{v}_j \odot_j \epsilon_j \\ 0 & \text{otherwise} \end{cases}, \tag{6.10}$$

so $j = \{1, \; \ldots \, , m\}$ and $\odot_j \in \{<, \leq, =, \geq, >\}$.

As shown in Fig. 6.3 three different types of conditions $\mathrm{ic}_j(t)$ are distinguished for the activation of the behavior. In *permanent activation* the corresponding relation from the input $\mathbf{p}$ has to be fulfilled during the whole time when the behavior shall be active (see Eq. (6.11)). In *ordering activation* the corresponding relation from the input $\mathbf{o}$ has to be fulfilled at some point in time before the behavior shall get active (see Eq. (6.12)). In *enabling activation* the corresponding relation from the input $\mathbf{e}$ has to be fulfilled at the exact point in time when the behavior shall get active (see Eq. (6.13)). Formally, the expression for these conditions are

$$\mathrm{ic}_j(t) = \begin{cases} 1 & \text{if} \quad \mathrm{ir}_j(t) = 1 \\ 0 & \text{otherwise} \end{cases}, \tag{6.11}$$

$$\mathrm{ic}_j(t) = \begin{cases} 1 & \text{if} \quad \exists t_0 \leq t : \mathrm{ir}_j(t_0) = 1 \\ 0 & \text{otherwise} \end{cases}. \tag{6.12}$$

Enabling conditions are the most complex ones. In order to determine whether an enabling condition is fulfilled at time t, it has to be checked whether there is a point in time $t_0 \leq t$ at which all other conditions were fulfilled. Furthermore, all conditions have to be fulfilled from t_0 on, thus

$$\mathrm{ic}_j(t) = \begin{cases} 1 & \text{if} \quad \exists t_0 \leq t : \left(\bigwedge_{k=1}^{m} \underset{enabling}{\mathrm{ir}_k(t_0)} = 1 \right) \\ & \qquad \wedge \left(\bigwedge_{k=1}^{m} \underset{ordering}{\mathrm{ic}_k(t_0)} = 1 \right) \\ & \qquad \wedge \left(\bigwedge_{k=1}^{m} \underset{permanent}{\mathrm{ic}_k(t_1)} = 1 \; \forall t_1 : t_0 \leq t_1 \leq t \right) \\ 0 & \text{otherwise} \end{cases}, \tag{6.13}$$

where $\bigwedge$ is the logical AND operator. The behavior signals of a CBS are calculated as follows

$$\mathbf{a}(t) = \mathbf{s}(t)(1 - \mathbf{i}(t)) \prod_{j=1}^{m} \mathrm{ic}_j(t) = \vartheta \prod_{j=1}^{m} \mathrm{ic}_j(t), \tag{6.14}$$

$$r(t) = \prod_{j=1}^{m} \mathrm{ic}_j(t). \tag{6.15}$$

The terms $\mathbf{a}$, $\mathbf{s}$, $\mathbf{i}$ were defined in Tab. 6.1, and ϑ is the activation of the behavior conforming to Eq. (6.2). In the case study of Sec. 6.5.3, the CBS node is used for synchronizing a top-down and bottom-up visual saliency tasks under enabling conditions.

Reinforcement Learning

According to Kaelbling et al. [92], *reinforcement learning* (RL) can be viewed as the mapping from situations to actions so a reward signal is maximized. The learner is not told which actions to take, instead, it must discover those that yield the most reward by trying them. In the most interesting and challenging cases, actions may affect not only the immediate reward, but also the next situation, and through that, all subsequent

rewards. These characteristics (trial-and-error search and delayed reward) are the two most distinguishing features of RL.

RL-based policies have been studied in the context of service and industrial automation. In a previous work (see Chame & Martinet [40]) a cognitive model considering RL was proposed to automate a pick-and-place task. Some other applications include grasping (e.g. in Baier-Lowenstein & Jianwei [14], Moussa & Kamel [129]), and navigation (e.g. Zhu & Levinson [192]). In Peters et al. [141] the scalability of RL to higher dimensional spaces for applications with humanoid robots is discussed, under the natural policy gradient representation.

One of the challenges in RL is the trade-off between exploration and exploitation (Russell & Norvig [159]). To obtain maximal reward, an agent must prefer actions that it has tried in the past and found to be effective in producing reward. But to discover such actions, it has to try actions that it has not selected before. The dilemma is that neither exploration nor exploitation can be pursued exclusively without failing at the task. The agent must try a variety of actions and progressively favor those that appear to be best. On a stochastic task, each action must be tried many times to gain a reliable estimate of its expected reward.

Beyond the agent and the environment, four main sub-elements of a RL system can be identified: a *policy*, a *reward function*, a *value function*, and, optionally, a *model* of the environment. This elements are described in Tab. 6.2.

Element	**Description**
Policy	A policy $\pi : S \rightarrow A$ defines the agent's behavior at a given time. It maps from perceived states of the environment to actions to be taken when in those states.
Reward	It is a function that relates each perceived state (or state-action pair) of the environment to a single number indicating the intrinsic desirability of that state.
Value function	It specifies what is good in the long run. Roughly speaking, the value of a state is the total amount of reward an agent can expect to accumulate over the future, starting from that state.
Model of the environment	It mimics the behavior of the environment. For example, given a state and action, the model might predict the next state and reward. Models are used for planning, by considering future situations before they are actually experienced.

Table 6.2 – Components of a RL problem.

In the action selection problem Thrun et al. [178] have distinguished between uncertainty in the action effects and uncertainty in perception. Two important tools for designing RL-based tasks are *Markov decision process* (MDP) and *partially observable Markov decision process* (POMDP). The MDP framework considers the state as fully observable under stochastic effects of actions, whereas in POMDP the agent actively gathers information about the task, due to the lack of observability of the state. However, according to Barto & Mahadevan [15], despite considerable research is based on these formalisms, RL is not restricted to discrete state and action representations. Thus, continuous representations have been derived from statistical estimation theory, so the policies are parametrized (e.g. the PI^2 algorithm by Theodorou et al. [176], the black-box optimization approach PI^{BB} by Stulp & Sigaud [173], among others).

In this work the discrete MDP framework is going to be discussed. Formally, an

MDP is composed by a 5-tuple $(S, A, \tau(.,.,.), \psi(.,.,.), \gamma)$, where S is the state space, A is the action space, and $\gamma \in [0, 1]$ is the discount factor. The transition probability function $\tau(a, s, s')$ gives the probability that taking the action a in the state s at a given time-step t, would lead to the state s' at time-step $t + 1$. It is defined by

$$\tau(a, s, s') = p(s_{(t+1)} = s' | s_{(t)} = s, a_{(t)} = a). \tag{6.16}$$

The reward function $\psi(s, a, s')$ gives the expected value of the next reward $r_{(t+1)}$, when being in the state s, tacking the action a, and getting to the state s'. It is defined by

$$\psi(a, s, s') = E(r_{(t+1)} | s_{(t)} = s, a_{(t)} = a, s_{(t+1)} = s'). \tag{6.17}$$

Figure 6.4 illustrates the graphical representation of a MDP.

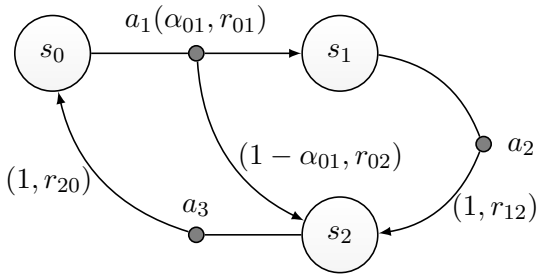


Figure 6.4 – Graphical representation of an hypothetical 3-state MDP. State nodes are represented by light circles. Action nodes are represented by small dark circles. The transition probabilities α and reward r are also shown.

The *Q-learning* algorithm was firstly introduced by Watkins [186]. It is suited to the case when the agent does not possess a model of the world (differently from the *value iteration* algorithm, see Thrun et al. [178]). Let the function $Q : S \times A \to \mathbb{R}$ provide the value of a state-action combination. The expected discounted reinforcement $Q^*(s, a)$ of taking action a in state s, then continuing to select actions optimally, can be defined recursively so

$$Q^*(s, a) = Q^*(s, a) + \gamma \sum_{s' \in S} \tau(a, s, s') \max_{a'}(Q^*(s', a')). \tag{6.18}$$

The learned action value function $\hat{Q}(s, a)$ directly approximates $Q^*(s, a)$, the optimal action value function. Thus, the Q-learning rule is such that

$$\hat{Q}(s, a) = \hat{Q}(s, a) + \nu(\psi(a, s, s') + \gamma \max_a(\hat{Q}(s', a')) - \hat{Q}(s, a)) \tag{6.19}$$

where $\nu \in [0, 1]$ is the learning rate. That is, the extend to which newly acquired information will override the old information. An episode ends when state $s_{(t+1)}$ is a final state (also called *absorbing state*). The algorithm is illustrated in Fig. 3 (Ertel [63]).

Case studies

The studies conducted is this section explore different aspects in the context of the ASP, such that, embodiment, knowledge representation, and learning. In the models pro-

Algorithm 3 Q-learning

```
1: procedure LEARN
2:     Q̂(s, a) ← initialize()                          ▷ Arbitrary
3:     repeat                                          ▷ for each episode e
4:         s ← initialize(initialstate)
5:         repeat                                      ▷ for each step of the episode
6:             Choose a from s using the policy derived from Q̂(s, a)   ▷ e.g., ϵ-greedy
7:             Take action a
8:             Observe r, s′
9:             Q̂(s, a) ← Q̂(s, a) + ν[r + γ max_a′ Q̂(s′, a′) − Q̂(s, a)]
10:            s ← s′
11:        until s is terminal
12:    until e is the last episode
```

posed, local representations obtained from proprioceptive data and visual processing are considered (i.e. the computation of color-based and dense- optic-flow-based segmentation, see Secs. 3.4.3 and 3.4.4). In the first case study, concurrent walking modes are defined so a reactive implementation of the approach task is proposed, by relying on distributed action-oriented representations of the object (differently from the last two chapters, where a rough 3D model of the object was used, and a unique task controlled the walk primitive of the robot). In the second case study a multi-objective navigation task is designed where the agent has to avoid obstacles as it approaches the object of interest. In the third case study learning is considered to obtain more efficient solutions to the task. The models proposed require of an acquisition rate (around 30 Hz) that is not available in the platform Nao, thus, the evaluations are conducted in simulation under Webots.

Materials and resources

The algorithms were implemented in the C++ programming language. Scheduling was obtained with the boost library 1.54.0. The vision processing was obtained with the support of the OpenCV 2.4.8 library. The robot functionalities were accessed through the naoqi 1.14 library. The programs were developed in the Eclipse Juno IDE under Ubuntu 12.04.5 LTS (Precise Pangolin). The simulations were carried out under the Webots robot simulator 7.4.0 by Cyberbotics. The host plattform was a HP Compaq Elite 8300 Convertible Microtower (8x Intel(R) Core(TM) i7-3770 CPU @ 3.40GHz, 8GB DDR3 RAM, Intel HD Graphics, HD ATA WDC WD5000AAKX-6).

Behavior-based models implementation

The models are implemented from scratch. In the hierarchical architectures proposed behaviors execute asynchronously. Moreover, different behaviors may concurrently attempt to access a sensory or motor resource of the robot. This is a more complex scenario that in the studies of Chapters 4 and 5. Hence, the runtime organization of behaviors has to be discussed in more details. The scheduling algorithm used is first in first out (FIFO), also known as first come first served. Processes are added to the ready queue in the order of arrival. Context switches only occur upon process termination, so no reorganization of the queue is required, and the scheduling overhead is minimal. Since the host platform is multi-core, concurrency is ensured. The risk that a process would hold others is practically negligible, so deadlines are easily met (this was evaluated in

simulations, delays of 1 ms were rarely observed).

CS-I: Action-oriented approach

The study developed in Sec. 5.5.4 showed that, by observing the egocentric localization of the object, from the knowledge of a rough 3D model, and the color of the object; the robot could perform the approaching task while mimicking human walk style. That is, the knowledge required to represent the object in the sensory ego-space was not obtained from the context of the task (i.e. a 3D model is an action-independent representation, see Sec. 5.3.3). Thus, an interesting question to be answered is whether similar results can be achieved by fully relying on action-oriented representations (i.e. locally effective features to guide behavior).

Thereby, in this case study the task is defined according to the model presented in Fig. 6.5. Inspired by the FIT and GS theories of attention (see Sec. 3.2), image features obtained from color and optic flow saliency are combined to perceive the object of interest. A two-layer hierarchy is proposed. The data and motion buffers are related to the host platform. The top layer contains nodes that operate as virtual sensors and nodes in charge of controlling the motion primitives of the robot. Regular nodes are represented in gray, the fusion behavior in blue, and the conditional node in yellow. The data signal is represented by the thick gray arrow, whereas meta-data (i.e. control data) is represented by the thin black line. For comparison, in Fig. 6.6 the analogous version of the behavior scheme of Fig. 5.3 is given in the iB2C framework. Sensory and localization tasks have been grouped in the same behavior for simplicity. As noticed, single behavior modes control the walk and the head direction primitives.

Behavior definitions

As shown in Fig. 6.5, a total of ten behaviors were defined for handling specific aspects of the task. Since the walk primitive of the robot considers motion expressed in Cartesian coordinates, the output of the Saggital, Frontal, and Angular Motion behaviors, are related respectively to the regulation of the components of the 2D pose $m = [X\ Y\ \phi]^{\mathrm{t}}$. Next, the implementations of the behaviors are detailed.

Data Acquisition

It is in charge of querying the robot for the proprioceptive and visual data. The objective of this behavior is to guarantee a centralized and more efficient access to the resources, thus avoiding overheads in the network protocols. The internal state consists in a buffer that stores consecutive acquisitions. The output $\mathbf{u}$ of the behavior contains a list of sequence of raw images captured on-board and joint measurements. The activity signal is set to $a = 1$ if no runtime exceptions occur, and $a = 0$ otherwise. The target rating signal is $r = a - 1$.

Dense Optic Flow

It is in charge of computing the dense optic flow induced by the robot motion (the scene is assumed to be static). The technique is detailed in Sec. 3.4.4. Thus, the output $\mathbf{u}$

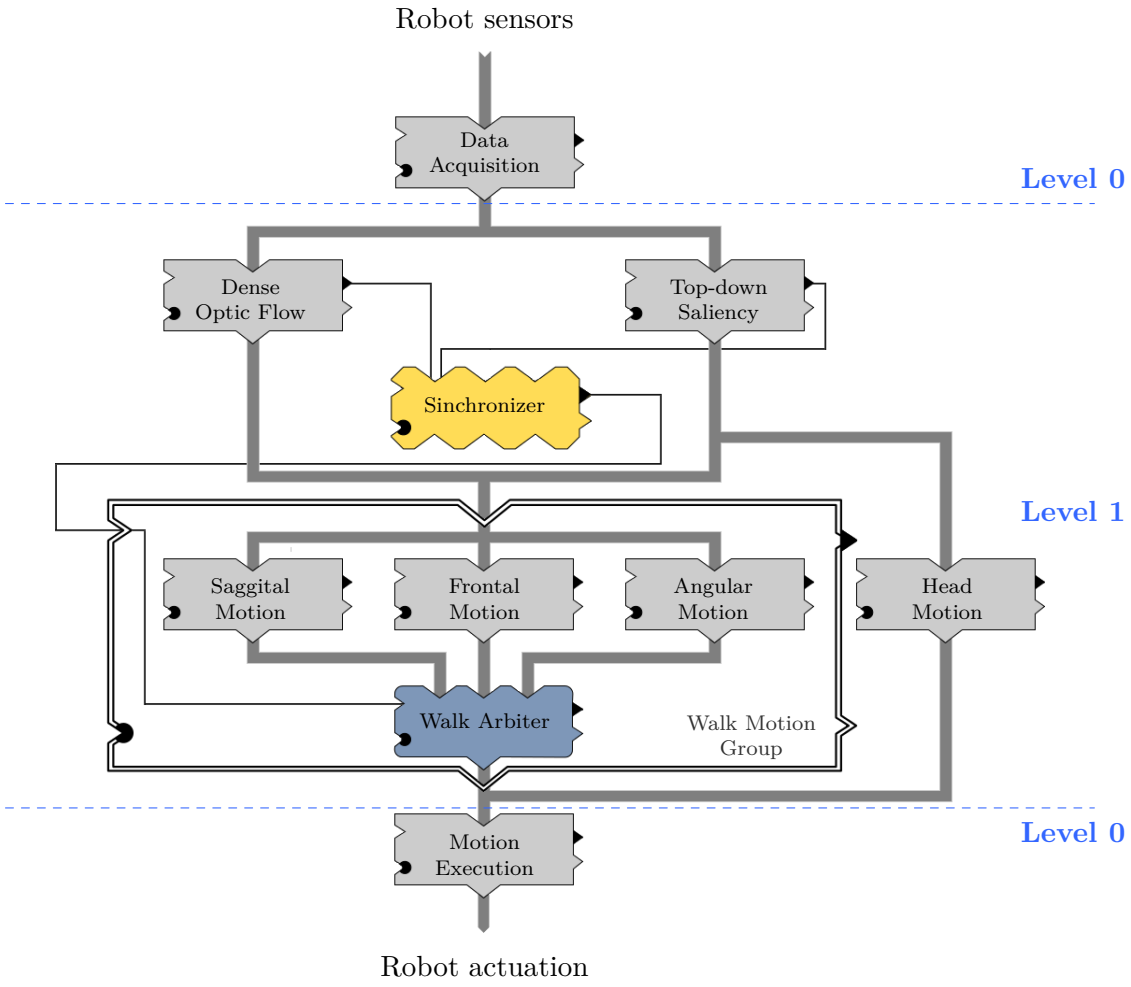


Figure 6.5 – Two-level hierarchy model. The level 0 corresponds to the real sensory and motor data. Virtual sensors and behavior primitives are defined in the level 1.

of the behavior is the image $f(x, y, t)$ representing the optic flow in the x and y components during the time interval t, as defined in Eq. (3.10). The activity signal is set to $a = 1$ if no runtime exceptions occur, and $a = 0$ otherwise. The target rating signal is set to $r = a - 1$.

Top-down Saliency

It is in charge of processing the supervised detection of the object of interest. For this, the segmentation algorithm detailed in Sec. 3.4.3 is employed. The binary image is obtained from a MRF modeling framework that considers the color model of the object under Gaussian noise. The output $\mathbf{u}$ of the behavior is a binary image O indicating the regions related to the object of interest and the centroid of the salient blob area (see Eq. (3.5)). In the task the problem of visual attention is not of concern. A single object of interest is set on the scene, so in case many blobs are salient, the biggest one is heuristically selected. The activity signal is set to $a = 1$ if there is a salient object and $a = 0$ otherwise. The target rating signal is set to $r = a - 1$.

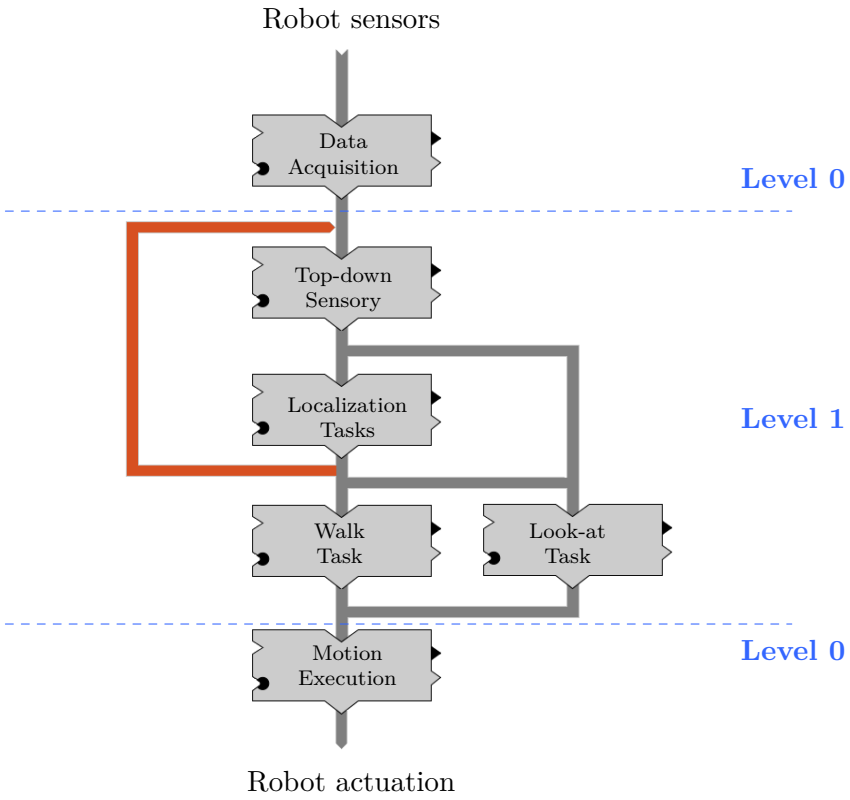


Figure 6.6 – Equivalent iB2C model for the behavior scheme describes in Sec. 5.3. The Sensory and Localization tasks were grouped in single nodes for simplicity. The orange arrow corresponds to the feedback sent for predicting the evolution of the embodied features detailed in Sec. 5.4.1

Synchronizer

As its name suggests, this node is in charge of synchronizing the output of the Dense Optic Flow and the Top-down Saliency nodes, since the information provided by them are related when controlling the walk (i.e. the region in the image where the object is, and the measured optic flow at such region). The condition to be fulfilled is of the type Enabling (see Eq. (6.13)). The activation signals of the input node set J are evaluated in Eq. (6.10), such that the condition is $a_j(t) = 1 \ \forall j \in J$. In case a synchronization is detected the Walk Motion Group is activated.

Saggital Motion

In this node the magnitude of the flow is taken as informative on the scene depth. Thus, from the optic flow vector $[\delta x \ \delta y]^{\mathrm{t}}$ (see Eq. (3.12)) associated to each pixel (x, y) of the image, and the binary image O obtained by the color-based segmentation of the object; the average magnitude $\check{\zeta}$ of the optic flow related to the object is defined by

$$\check{\zeta} = \frac{1}{n} \sum_x \sum_y O(x, y) \| [\delta x \ \delta y]^{\mathrm{t}} \|, \tag{6.20}$$

where $n = m_{00}$ is the zero moments of O, as defined in Eq. (3.4). The output $\mathbf{u}$ of the behavior includes $\check{\zeta}$ and the associated correction in the saggital motion plane, so

$$\mathbf{u} = \begin{cases} \bar{X} & \text{if} \quad \check{\zeta} - \check{\zeta}^* < -\epsilon_1 \\ -\bar{X} & \text{if} \quad \check{\zeta} - \check{\zeta}^* > \epsilon_1 \\ 0 & \text{otherwhise} \end{cases} . \tag{6.21}$$

Here $\check{\zeta}^*$ is the learned by kinesthetic demonstration. Thus, the robot is put to a static march so it moves to generate flow without changing the position. The parameter ϵ_1 is a threshold tolerance. Thereby, the behavior assumes that the mean flow magnitude produced by the object approaching in the saggital plane direction, is similar to the one registered in static march (this is reasonable for walking, but it may not hold in case of running). As seen, the correction is a step signal of magnitude $\bar{X}$. The activity signal is set to $a = 1$ if no runtime exceptions occur, and $a = 0$ otherwise. The target rating signal is set to $r = 0$ if $\mathbf{u} = 0$, and $r = 1$ otherwise.

Frontal Motion

This behavior relies on the analysis of the bilateral symmetry of the object (other features are reported in Hauagge [80]). Let the salient blob b be split into a left and a right half by the vertical axis of symmetry (i.e. a parallel to the image y-axis passing through the center c of the bounding box enclosing the blob), so $b = b_{\mathrm{L}} \bigcup b_{\mathrm{R}}$. The proportion k is defined such that

$$k = \frac{m_{\mathrm{L00}}}{m_{\mathrm{R00}}}, \tag{6.22}$$

where m_{L00} and m_{R00} are the zero moments associated to b_{L} and b_{R} respectively. The output $\mathbf{u}$ of the behavior is the correction in the frontal plane, defined by

$$\mathbf{u} = \begin{cases} \bar{Y} & \text{if} \quad k - k^* > \epsilon_2 \\ -\bar{Y} & \text{if} \quad k - k^* < -\epsilon_2 \\ 0 & \text{otherwhise} \end{cases} . \tag{6.23}$$

Here k^* is the demonstrated value. It is registered by placing the robot in the desired configuration with respect to the object. The parameter ϵ_2 is a threshold tolerance. Similarly to the previous case, motion in the frontal plane is defined by a step signal of magnitude $\bar{Y}$. The activity signal is set to $a = 1$ if no runtime exceptions occur, and $a = 0$ otherwise. The target rating signal is set to $r = 0$ if $\mathbf{u} = 0$, and $r = 1$ otherwise.

Angular Motion

This behavior is in charge of regulating the angular motion of the robot. The desired correction ϕ is obtained from the heuristics knowledge that the z-axis of the neck yaw is approximately aligned to the motion plane normal direction (see Sec. 4.6.3), so, it is informative on the heading direction of the object. Thus, it is defined by

$$\phi = \max(\min((\alpha - \alpha^*), \bar{\phi}), -\bar{\phi}), \tag{6.24}$$

where α is the yaw posture of the neck, α^* is the desired state learned by demonstration, and $\bar{\phi}$ is a saturation to the angular motion. The output $\mathbf{u}$ is defined by considering a threshold tolerance ϵ_3, such that

$$\mathbf{u} = \begin{cases} \phi & \text{if} \quad |\phi| > \epsilon_3 \\ 0 & \text{otherwhise} \end{cases} . \tag{6.25}$$

The activity signal is set to $a = 1$ if no runtime exceptions occur, and $a = 0$ otherwise. The target rating signal is set to $r = 0$ if $\mathbf{u} = 0$, and $r = 1$ otherwise.

Head Motion

Similarly to the Look-at task defined in Sec. 4.4.2, this node is in charge of directing the head towards the object of interest. However, only the close-loop regulation is considered (i.e. there is no open-loop anticipation of the object's pose, since the localization in the ego-cylinder is not observed).

Let $\mathbf{q}_\text{h} = [\alpha \ \beta]^\text{t}$ be the correction of the joint neck positions to center the object in the field of view (see Eq. (4.21)). The output $\mathbf{u}$ of the behavior is defined from the threshold ϵ_4, such that

$$\mathbf{u} = \begin{cases} \mathbf{q}_\text{h} & \text{if} \quad d > \epsilon_4 \\ 0 & \text{otherwhise} \end{cases} . \tag{6.26}$$

Here $d = \|(c_\text{x}, c_\text{y}) - (\text{i}_\text{x}, \text{i}_\text{y})\|$ is the euclidean distance between the coordinate of the blob centroid (c_x, c_y), and the coordinate of the center of the image $(\text{i}_\text{x}, \text{i}_\text{y})$. The activity signal is set to $a = 1$ in case the object is detected, and $a = 0$ otherwise. The target rating signal is set to

$$r = \frac{1}{2}\left(\frac{|c_\text{x} - \text{i}_\text{x}|}{\text{i}_\text{x}} + \frac{|c_\text{y} - \text{i}_\text{y}|}{\text{i}_\text{y}}\right), \tag{6.27}$$

Walk Arbiter

This behavior is in charge of determining the walk correction to steer the robot towards the objects. As discussed in Sec. 5.3.1, human locomotion is mostly non-holonomic when approaching the object, though holonomic corrections may be applied in the proximity of the object. A state-based arbitration scheme is considered to select the motion style. As illustrated in Fig. 6.7, the motion policy transits between the Non-holonomic and the Holonomic states.

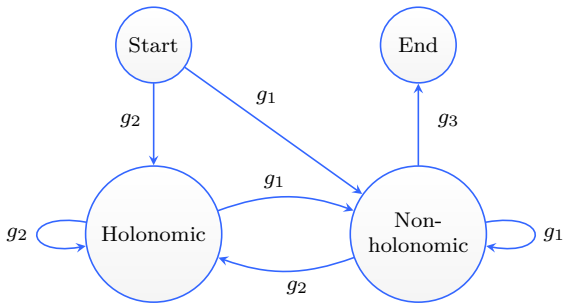


Figure 6.7 – State automate for discrete events arbitration between holonomic and non-holonomic walk style. The transition events are denoted by g_i

The module is initialized in the Start state. The Non-holonomic state combines the information from the output of Saggital and Angular Motion, such that the motor command issued by the state is $\mathbf{u}_1 = [X\ 0\ \phi]^{\mathrm{t}}$. The Holonomic state combines the information from the three motion behaviors, so the motor command issued by the state is $\mathbf{u}_2 = [X\ Y\ \phi]^{\mathrm{t}}$. The state End is reached once an interruption signal is produced, in this case the motor command $\mathbf{u}_3 = 0$ is issued to stop the robot. Thereby, the arbitrated output $\mathbf{u}$ of the behavior is defined by

$$\mathbf{u} = (1 - g_3)(g_1\mathbf{u}_1 + g_2\mathbf{u}_2) + g_3(\mathbf{u}_3), \tag{6.28}$$

where the event g_1 is the activation of the Non-holonomic state, such that $g_1 = 1$ if $\check{\zeta} < \epsilon_5$, and $g_1 = 0$ otherwise. Here $\check{\zeta}$ is the average flow related to the object of interest (see Eq. (6.20)), and ϵ_5 is a parameter representing the transition threshold. The event $g_2 = 1 - g_1$, and the event g_3 is a user interruption or a runtime exception. The activation signal is $a = 1 - g_3$. The target rating signal is set to $r = 0$ if $|\mathbf{u}| = 0$, and $r = 1$ otherwise.

Motion Execution

This behavior is in charge of sending the most recent commands to the motion primitives of the robot. So it centralizes the access to the robot actuation in order to avoid concurrency issues. The input data is $\mathbf{d} = [\mathbf{d}_{\mathrm{w}}\ \mathbf{d}_{\mathrm{h}}]^{\mathrm{t}}$, where $\mathbf{d}_{\mathrm{w}}$ is a relative 2D pose to be execute by the walk primitive, and $\mathbf{d}_{\mathrm{h}}$ is the desired motion of the robot neck. The activity signal is set to $a = 1$ if no runtime exceptions occur, and $a = 0$ otherwise. The target rating signal is set to $r = a - 1$.

Experiments

A scene was simulated in Webots where the agent has to approach a painting on the wall. The frequency for the Data Acquisition and Motor Execution nodes is set to 30 Hz. The rest of the nodes run at a frequency of 20 Hz. The model parameters are detailed in Tab. 6.3. Similarly to the study cases of Chapters 4 and 5, the color model of the object is provided by first-person demonstration. Likewise, the desired states of the Saggital, Frontal, and Angular Motion nodes are also provided by kinesthetic demonstration. Three experiments are designed. In the first experiment the model presented in Fig. 6.5 is evaluated. In the second experiment the texture of the object is changed such that the bilateral symmetry is affected. The idea is to evaluate whether the criteria employed to observe the lateral displacement is effective with less symmetrical objects. In the third experiment a simpler and computationally less expensive version of the model is studied. As shown in Fig. 6.8, the calculation of dense optic flow is approximated by the estimation of the sparse flow generated by the centroid of the emergent blob, which operates as a virtual sensor. Thereby, the state space of the Top-down Saliency node is extended to include the image position of centroid in the last two consecutive acquisitions, which is denoted respectively by $\mathbf{c}_{(t)}$ and $\mathbf{c}_{(t-1)}$. So the average flow considered in Saggital Motion (see Eq. (6.20)) is approximated by the sparse flow

$$\tilde{f} = \|\mathbf{c}_{(t)} - \mathbf{c}_{(t-1)}\|, \tag{6.29}$$

Id	Description	Value
ϵ_1	Tolerance for Saggital Motion convergence.	0.5
ϵ_2	Tolerance for Frontal Motion convergence.	0.02
ϵ_3	Tolerance for Angular Motion convergence.	0.06 rad
ϵ_4	Tolerance for Head Motion convergence.	5
ϵ_5	Mean flow threshold to switch between the holonomic and the non-holonomic motion styles.	20

Table 6.3 – CS-I task parameters.

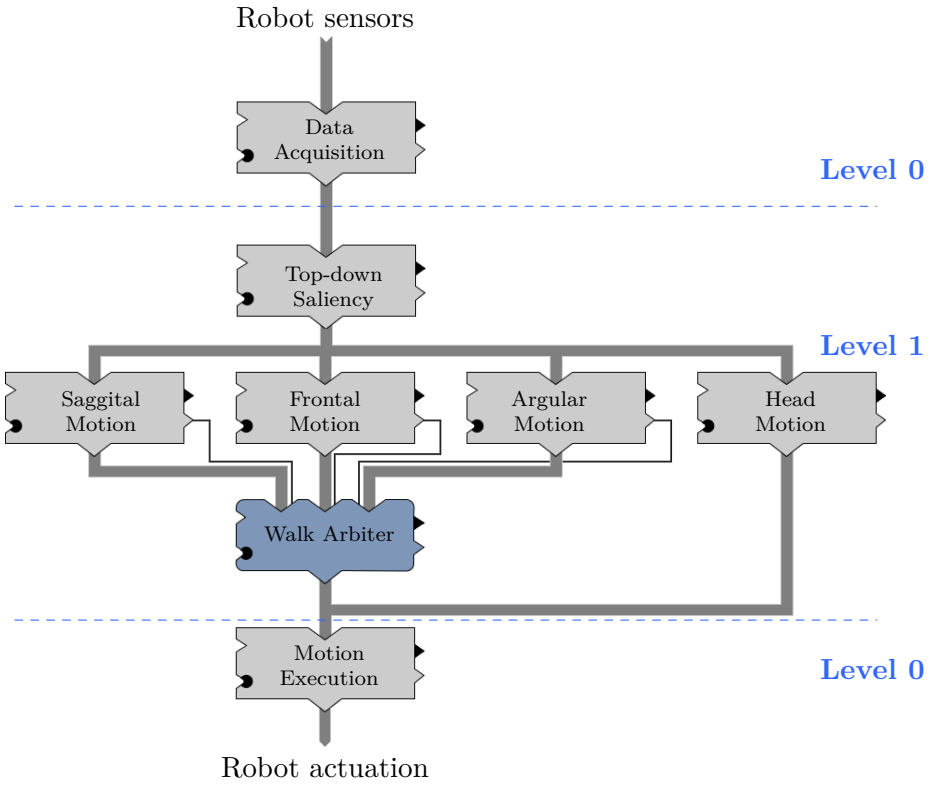


Figure 6.8 – CS-I sparse flow task model. The optic flow of the object is approximated by the virtual flow of the centroid. The two-level hierarchy is maintained.

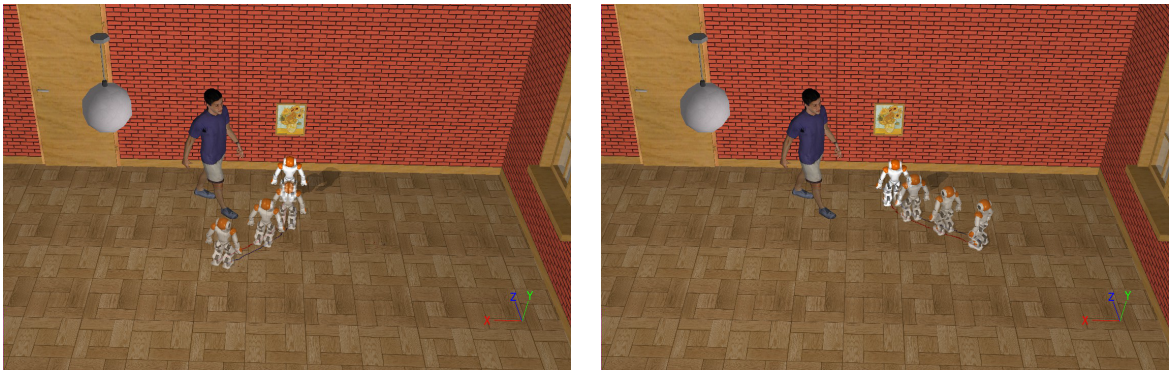


Figure 6.9 – Reactive object approach. Some frames have been superimposed to illustrate the trajectory followed by the agent under two distinct initial conditions. The scene is static (the human model was not moving).

Results

In the first experiment the agent was able to approach the painting. Figure 6.9 shows the results for two different initial conditions. As it can be noticed, the robot performed

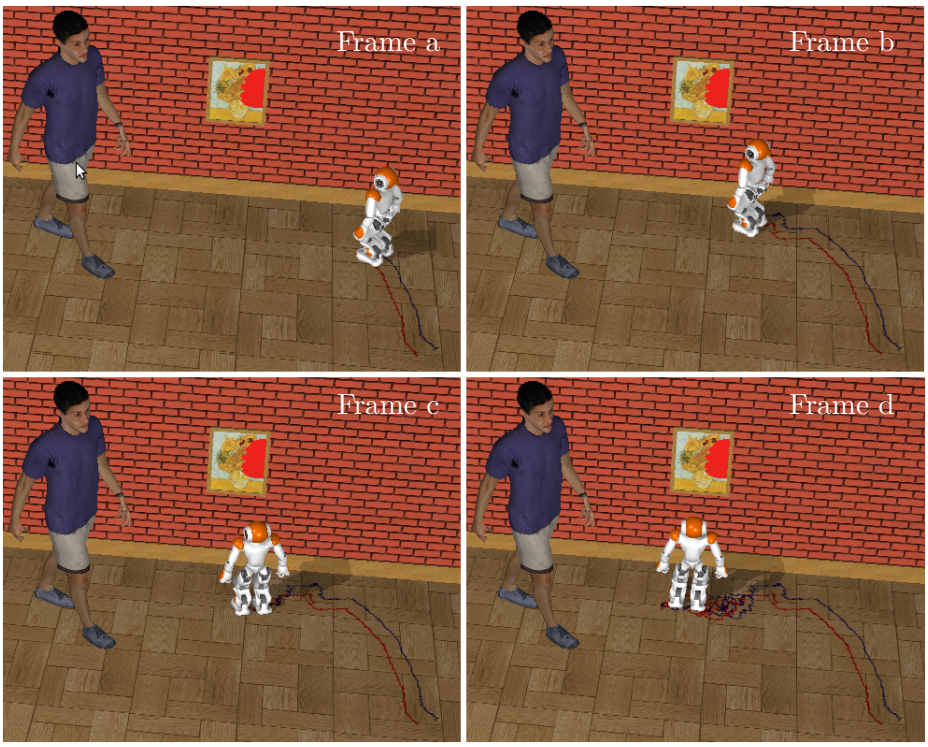


Figure 6.10 – Snapshots of the evaluation of the bilateral symmetry condition.

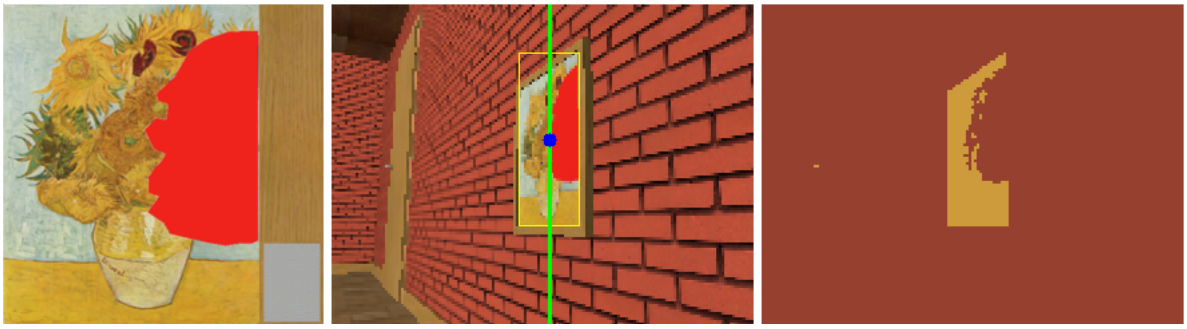


Figure 6.11 – On-board view of the bilateral symmetry condition. On the left the painting's texture is changed to generate a perceived asymmetric blob. In the center the features are marked. The centroid of the bounding box is plotted in blue and the vertical axis is green. On the right the salient blob. The on-board view corresponds approximately to the situation of "Frame a" in Fig. 6.10.

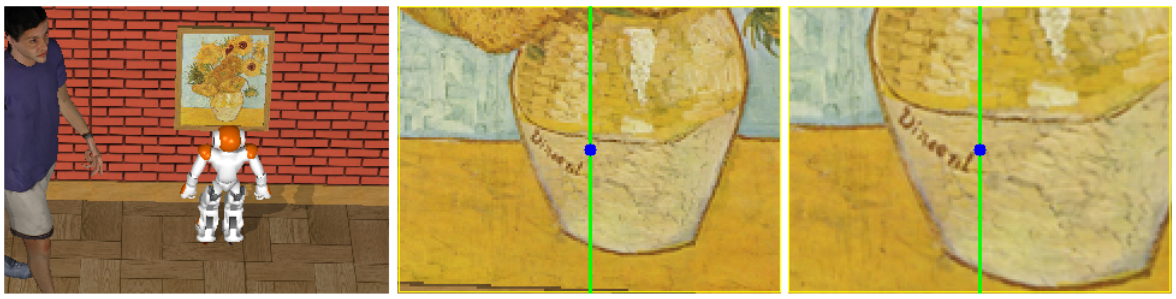


Figure 6.12 – On-board view of the sparse flow evaluation. In case the robot is critically proximal to the object, the centroid flow is no longer informative for the Saggital Motion behavior.

most of the time non-holonomic motions, so the trajectories obtained were efficient. The results for the second experiment where less satisfactory. As illustrated in Fig. 6.10, the robot required more time to converge to the desired location since it took a less optimal path. This is due to the fact that the heuristics employed in Frontal Motion is satisfied from more locations, since the object is less bilateral symmetric (see Fig. 6.11, the agent is attracted to a diagonal location that corresponds approximately to "Frame A" in Fig. 6.10). In the last experiment the robot was able to efficiently approach the object when it was completely visible. Though, when it was cropped in the field of view the virtual flow detected was noisy, which induced errors in Saggital Motion. As shown in Fig. 6.12, the worse case occurs when the robot is critically proximal to the object, so the blob spans over the whole image and the sparse flow is null.

Discussion

From the results obtained an interesting aspect to be noticed is that, by exploiting embodiment and local heuristics, and in case a sufficiently high acquisition rate is available in the platform, the model proposed can provide similar trajectories to those obtained in Sec. 5.5.5 (that were based on the observation of the ego-cylindrical localization, from a disembodied representation of the object). This is consistent with the *physically grounding hypothesis* (Brooks [29]), so action-oriented representations would ground the solution of the task. That is, the representations would be obtained from the embodied experience (e.g. the color perceived from the object, the optic flow, the neck posture, and the bilateral proportion of the object). However, as seen in the second experiment, the heuristics used by Frontal Motion did not produce the same results for a less symmetric stimulus. Thus, alternative representations should be investigated to determine the corrections on the frontal plane. Generic frameworks are available in the literature of machine learning (e.g. Self Organization Maps, Artificial Neural Networks, among others), though training is normally required to produce reliable observations. This is in fact a distinctive aspect of pure EC models, where the generalization of the solution is not ensured, even to small changes in the task specifications (which can be disadvantageous for service robotics applications). In practice different sort of representations may be required. Finally, other interesting aspect observed was the fact that, in case a color model of the object is available, and the object is fully visible, virtual features (e.g. the blob centroid) can be used to obtain a computationally less expensive solution, based on the sparse flow estimation of depth.

CS-II: Object approach and obstacle avoidance

The previous case study showed that optic flow can provide an estimate on the depth of a known object with respect to the camera sensor, so a question to be answered is: can the information obtained from optic flow be used to avoid unknown obstacles in the scene? In the context of mobile robot control some studies have explored the use of optic flow processing. A work by Yoo et al. [190] investigated the control of an unmanned aerial vehicle by heuristically balancing between the right and the left optical flow vectors. The same strategy was employed by Souhila & Karim [169] for controlling a wheeled robot. A work by Low & Wyeth [111] has reported that consistent information about obstacles can be obtained from sparse optical flow in a wheeled robot Pioneer. However, when considering the aspect of embodiment, the mobility of the visual system of these robots may be different than the one induced by walking robots, so it is not clear whether reliable

information about obstacles can be obtained in humanoids. In the RoboCup competition sensorimotor mapping and activity mining, relying on optical flow patterns, has been used to learn different situations of the game (e.g. to select most relevant skill, such that kicking, approaching, or catching the ball; see Ogino et al. [136]), though the visual task was strongly bound to the structure of the environment (e.g. the assumption that a ball rolls on the floor).

In this study a multi-objective navigation task is designed in which the agent has to avoid obstacles as it approaches the object of interest. Figure 6.13 presents the model proposed. Similarly to the precedent case, regular nodes are represented in gray and the fusion behavior in blue. CBS nodes are not used. The two-layer hierarchy of the model is maintained. The data signal is represented by the thick gray arrow. It is interesting to notice that feedback represented in orange, from the last motion command sent to the robot, is now available to all the nodes. The control signals are represented by the thin black lines. The model comprises a total of ten concurrent behaviors, which are generally more complex than in the previous case. Next, these behaviors are detailed.

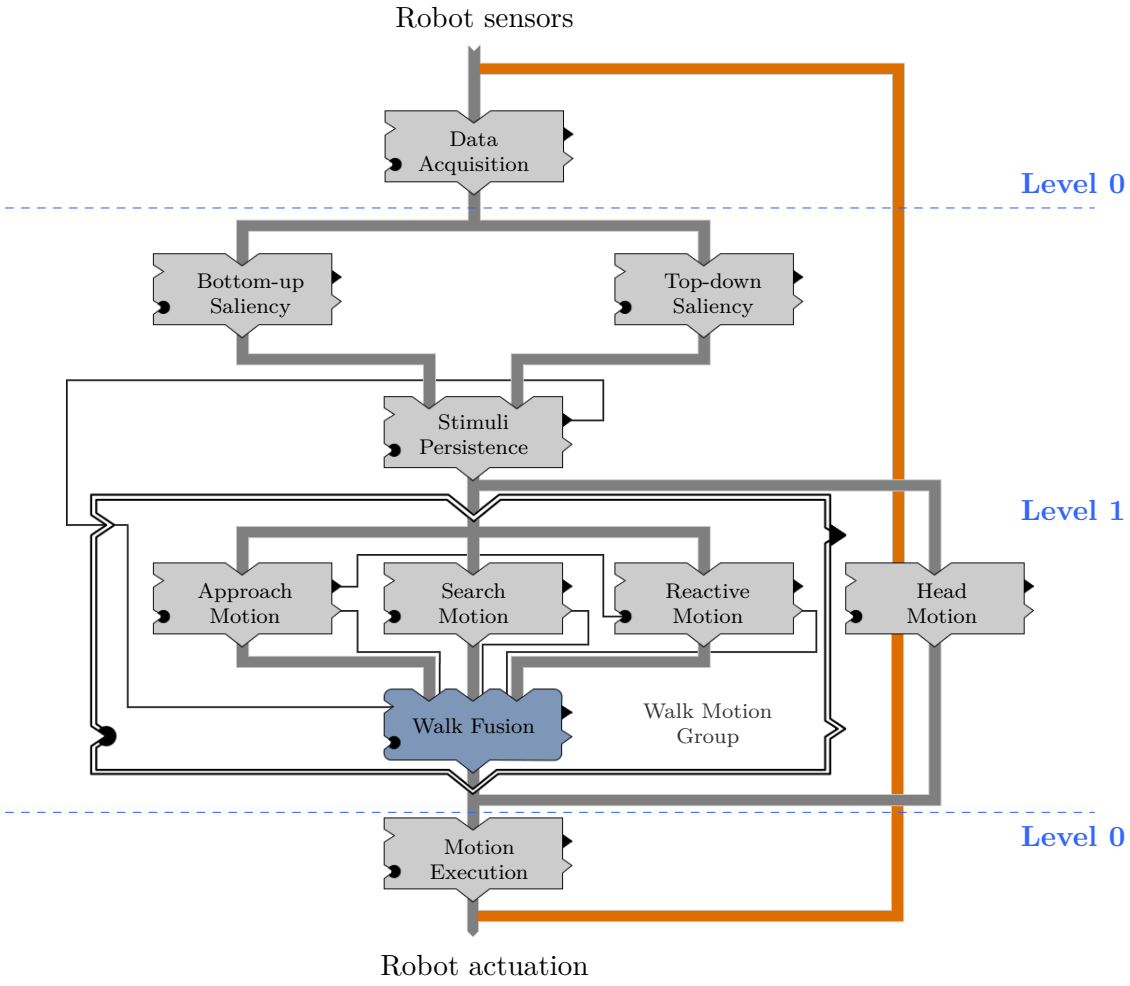


Figure 6.13 – CS-II task model.

Behaviors

Some behaviors, such that Motion Execution, have been fully defined in the preceding study. Some others behaviors have been extended in the current architecture. This is the

case of Data Acquisition, that receives and propagates the last motion command sent to the walk primitive as feedback. Top-down Saliency is also extended to implement the full version of the Look-at task as detailed in Sec. 4.4.2 (i.e. by including both the predictive and the regulation control of the neck). The remaining behaviors are quite different, so are discussed in more detail.

Bottom-up Saliency

This node is in charge of unsupervised saliency detection based on the computation of dense optic flow. For this, the segmentation algorithm detailed in Sec. 3.4.4 is employed. The binary image is obtained by applying an heuristic threshold test ϵ_1 to the magnitude of the measured flow $\bar{f}$ (see Eq. (3.40)). Thereby, the output $\mathbf{u}$ of the behavior includes the measured flow, the binarization obtained from Eq. (3.9), and the centroid of the salient areas (see Eq. (3.5)). The centroid is used by the Stimuli Persistence node to represent locations related to the obstacle, so the Reactive Motion behavior can produce the control signal to avoid such locations. The activity signal is set to $a = 1$ if an obstacle is detected (that is, by applying a threshold test ϵ_2 to filter out noisy detection, so $m_{00} > \epsilon_2$), and $a = 0$ otherwise. The target rating signal is set to $r = 0$ if no runtime exceptions occur, and to $r = 1$ otherwise.

Stimuli Persistence

This behavior implements the sensory ego-cylinder, as detailed in Sec. 4.5. Thus, it works like a sensory buffer that ensures persistence of recent locations related to the object of interest and the obstacles. That is, given the camera motions and the fact that the vision sensor has a limited view angle, it is important to retain recent locations to keep motion consistency; an aspect also noticed by Fujita [71].

The input $\mathbf{d}$ of the behavior includes the information provided by the saliency detection (both bottom-up and top-down). In realistic scenarios, many regions may be identified in the retinal space by Top-down Saliency, so the information related to the object of interest has to be discriminated. As explained in Sec. 5.4, the selection relies on embodied filtering obtained through the Bayesian network. Thereby, the localization of the object of interest in the ego-cylinder is estimated from the retinal saliency and the 3D model of the object (see Sec. 4.5.3).

No model is available for observing an obstacle location ${}^{B}o$, so only the position component can be estimated. The bearing and the height of the blob centroid issued by the Bottom-up Saliency node are directly observable. Knowledge acquired from known stimuli can be exploited to estimate the distance to the obstacles. By kinesthetic demonstration, the robot is put to walk toward an object in the saggital plane direction, so the mean optic flow $\check{\zeta}$ associated to the object and its localization ${}^{\mathrm{C}}\zeta$ are registered. Thus, a rough estimation of the obstacle's depth ${}^{\mathrm{C}}\tilde{o}_{\rho}$ with respect to the camera frame C can be obtained, by establishing a linear correspondence between the mean flow magnitude $\check{o}$ of the bottom-up salient blobs, the mean flow magnitude $\check{\zeta}$ detected for the known object (i.e. it is calculated over the mask resulting from applying the logical AND operator to the top-down and the bottom-up saliency segmentations), and the observed depth ${}^{\mathrm{C}}\zeta_{\rho}$. Thereby, the obstacle's depth is estimated such that

$$^{\mathrm{C}}\tilde{o}_{\rho} = \frac{\breve{o}}{\breve{\zeta}}{}^{\mathrm{C}}\zeta_{\rho}, \tag{6.30}$$

where the position of the obstacle $^{B}\tilde{o}$ with respect to the base frame B is obtained from $^{B}\tilde{o} = {}^{B}\mathbf{T}_{\mathrm{C}}(\mathbf{q})^{\mathrm{C}}\tilde{o}$, so it depends on the current joint configuration $\mathbf{q}$.

Locations of stimuli are updated in the ego-cylinder by considering a motion model of the walk (which in this study is deterministic). Thereby, the prediction for the evolution of stimuli are determined according to Eq. (4.17). The heading direction for the obstacles are set to zero since they are not observed. The representations expire according to the forgetting factors $\gamma_{\mathrm{o}(i)}$ and γ_{ζ}, associated respectively to the obstacle o_i and the object of interest ζ. They are defined by

$$\gamma_{\mathrm{o}(i)} = 1 - \min\left(\frac{t_{\mathrm{o}(i)}}{\epsilon_3}, 1\right) \tag{6.31}$$

and

$$\gamma_{\zeta} = 1 - \min\left(\frac{t_{\zeta}}{\epsilon_4}, 1\right). \tag{6.32}$$

Here the parameters ϵ_3 and ϵ_4 represent respectively the expiration time for locations related to obstacles and the object on interest. The timers $t_{\mathrm{o}(i)}$ and t_{ζ} are independent (i.e. the information related to the obstacle and the object of interest may arrive at different instants). Once a location is stored its timer is initialized to zero. Since only one object of interest is tracked, the observation of the localization of the object overrides the previous information. In case the object leaves the field of view, the forgetting factor γ_{ζ} would provide a valuable information, so the search motion can be activated in the Walk Fusion node. Contrarily, the detection of an obstacle location does not override the previous ones (that continue to exist until the expiration time ϵ_3 is reached). However, a unique fused location $^{B}\tilde{o}$ is issued by the behavior (as if only one object would be detected), which is defined by

$$^{B}\tilde{o} = \frac{1}{L}\sum_{i=1}^{n} \exp(1 - \gamma_{\mathrm{o(i)}})^{B}o_{(i)}, \tag{6.33}$$

where L is a normalization term, and the issued forgetting factor is

$$\gamma_{\tilde{\mathrm{o}}} = \max(\gamma_{\mathrm{o}(i)}). \tag{6.34}$$

That is, the one associated to the most recent obstacle location.

To summarize, the output of the behavior is the vector $\mathbf{u}$ containing the active locations $^{B}\zeta$ and $^{B}\tilde{o}$, and the associated forgetting factors γ_{ζ} and $\gamma_{\tilde{\mathrm{o}}}$. The activity signal is $a = \max(\gamma_{\zeta}, \gamma_{\tilde{\mathrm{o}}})$. In case $\gamma_{\zeta} = 0$ the Search Motion policy is used in Walk Fusion to find the object. The target rating signal is set to $r = 0$ if no runtime exceptions occur, and $r = 1$ otherwise.

Approach Motion

This node is in charge of steering the robot toward the object of interest. It handles the control of locomotion as detailed in Sec. 5.3.3. Thus, the node implements the first-

order description of motion given in Eq. (5.6), which mimics the human walk style. The regulation of the view direction is delegated to Head Motion in order to preserve the modular philosophy, so the control of the head and the walk is decoupled. Thereby, the input $\mathbf{d}$ includes the current localization of the object in the ego-cylinder. The output $\mathbf{u} = [X\ Y\ \phi]^{\mathrm{t}}$ is the desired regulation of the walk. The activity signal is set to

$$a = \begin{cases} 1 & \text{if } \exists\ i \in \{1,2,3\} \mid (|\mathbf{u}_i| - \epsilon_{5i}) > 0 \\ 0 & \text{otherwise} \end{cases} . \tag{6.35}$$

where ϵ_5 is a 2D pose tolerance. An inhibitory signal is heuristically sent to Reactive Motion when $\hat{\mathbf{e}}_\rho < \epsilon_6$, in order to ensure the convergence of the task. That is, once closed enough to the object of interest the reactive motion is no longer relevant, since the agent would react to the presence of the object of interest itself, instead to an obstacle. The target rating signal is defined by

$$r = \min\left(\frac{1}{3}\sum_{i=1}^{3}\left(\frac{|\mathbf{u}_i|}{\epsilon_{5i}}\right), 1\right). \tag{6.36}$$

Reactive Motion

This node is in charge of directing the robot away from the obstacles. The input $\mathbf{d}$ contains the estimated position of the obstacle ${}^{B}\tilde{o}$ in the ego-cylinder (see Eq. (6.33)), and the associated forgetting factor $\gamma_{\tilde{o}}$ (see Eq. (6.34)). The principle is illustrated in Fig. 6.14. The robot heuristically moves in the opposite direction of the obstacle's bearing to leave the security region. The angular correction ϕ is obtained by

$$\phi = -\text{sign}(\theta)\max(\epsilon_7 - |\theta|, 0), \tag{6.37}$$

where θ is the bearing of the obstacle. The function sign(.) returns the signed unity, and ϵ_7 defines the bounds of the security region.

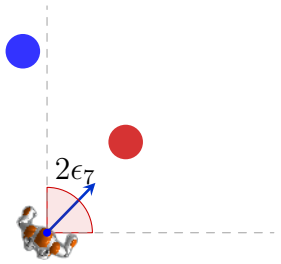


Figure 6.14 – The obstacle in red induces clockwise reactive motion, since it falls within the security region delimited by the dashed lines. The blue obstacle would not produce reactive motion in the depicted situation.

Motion in the saggital plane (see Fig. 2.12)) is also induced to bypass the obstacle. The idea is to advance while turning in a non-holonomic fashion. The closer the obstacle is, the less the robot should move in the sagittal plane to avoid the collision. Thereby, the desired motion is defined by

$$X = (1 - |\phi|/\epsilon_7)\epsilon_8 \tag{6.38}$$

where ϵ_8 is a parameter describing the saturation for the saggital plane displacement. The output of the behavior is $\mathbf{u} = [X\ 0\ \phi]^{\mathrm{t}}$. The activation signal is set to $a = 0$ if an inhibition signal is received from Approach Motion, and $a = \tilde{\gamma}_{\mathrm{o}}$ otherwise. The target rating is defined by

$$r = \frac{|\phi|}{\epsilon_7}. \tag{6.39}$$

Search Motion

The objective of this behavior is to search for the object once it has left the field of view. The input to the node is the prediction of the object's bearing $\tilde{\theta}$, conforming to Eq. (4.17). The output is the non-holonomic search motion $\mathbf{u} = [X\ 0\ \phi]^{\mathrm{t}}$, where the angular component is obtained by

$$\phi = \mathrm{sign}(\tilde{\theta})\mathrm{min}(|\tilde{\theta}|, \epsilon_9). \tag{6.40}$$

The parameter ϵ_9 is a saturation on the angular motion. Motion in the saggital plane is heuristically induced once the robot has turned to the expected location of the object with a tolerance ϵ_{10}, so the robot can wander until eventually re-locating the object. Thus,

$$X = \begin{cases} \bar{X} & \text{if} \quad |\tilde{\theta}| < \epsilon_{10} \\ 0 & \text{otherwise} \end{cases}. \tag{6.41}$$

The activation signal is set to $a = 1$ if $\gamma_\zeta = 0$, and $a = 0$ otherwise. The target rating is defined by

$$r = \min\left(\frac{|\tilde{\theta}|}{\epsilon_9}, 1\right). \tag{6.42}$$

Walk Fusion

This node is in charge of combining the three walk policies available. For this, two behavioral modes are defined. In the *approaching mode*, the object of interest is considered to be available if the forgetting factor $\gamma_\zeta > 0$ (see Eq. (6.32)). That is, even though the object can be eventually occluded, the localization is persisted in the sensory ego-cylinder for a while. A fusion between Approach and Reactive Motion is produced, according to the scheme described in Eq. (6.8). The other scenario corresponds to the *searching mode*, when $\gamma_\zeta = 0$ (i.e. the object is considered to be lost). In this case, the fusion is produced between Search and Reactive Motion.

Experiments

A scene was simulated in Webots where the agent has to approach to the blue can over the sofa, while avoiding the static columns. The frequency for the Data Acquisition and Motor Execution nodes was set to 30 Hz, the rest of the nodes run at a frequency of 20 Hz. The model parameters are detailed in Tab. 6.4. A first experiment was designed in order to evaluate whether the agent is able to accomplish the task, by relying only on

Approach Motion. This is to verify whether the task is challenging enough. The second experiment includes the approach to the object from ten distinct initial locations with the model fully operational. In the third experiment more columns are added to increase the difficulty on the task.

Id	Description	Value
ϵ_1	Bottom-up threshold test for segmentation.	12.0
ϵ_2	Bottom-up segmentation noise tolerance.	20
ϵ_3	Expiration time for obstacles in the ego-cylinder.	15
ϵ_4	Expiration time for the object of interest in the ego-cylinder.	20
ϵ_5	Approach Motion convergence tolerance. Distance in meters and angles in radians.	$[\rho\ \theta\ \phi]^t = [0.05\ 0.04\ 0.1]^t$
ϵ_6	Object distance test for Reactive Motion inhibition.	0.4 m
ϵ_7	Bounds of the security region for obstacle avoidance.	0.79 rad
ϵ_8	Saturation for saggital plane displacement in Reactive Motion.	0.1 m
ϵ_9	Saturation for angular motion search.	0.26 rad
ϵ_{10}	Tolerance for orientation correction in Search Motion.	0.03 rad

Table 6.4 – CS-II task parameters ϵ_i.

Results

As shown on the left side of Fig. 6.15, in the first experiment the agent lost the object of interest and did not finish the task when only the Approach Motion walk was active. In the second experiment the robot could approach the object while avoiding the obstacles. Figure 6.16 illustrates the reactive motion produced based on the segmentation of the optic flow. As seen in Fig. 6.17, the robot was able to avoid the locations of the obstacles.

Figure 6.15 – Non-reactive vs. reactive approach. On the left Reactive Motion is deactivated. Some frames capturing the evolution of the robot are superimposed. The robot ended up blocked by the obstacle. On the right, with the activation of the Reactive Motion the agent is able to accomplish the task.

Discussion

The experiments showed that the task could be accomplished, so the robot approached the object of interest and reacted to obstacle locations, from color and dense

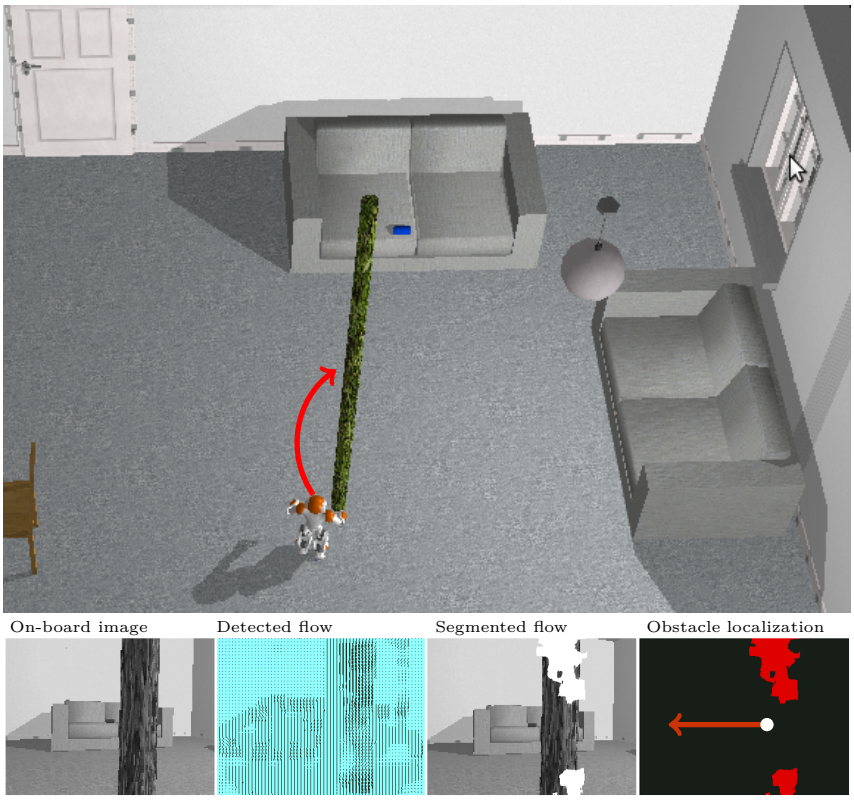


Figure 6.16 – Model performance. The robot avoids the column in the saggital plane direction by processing the optic flow.

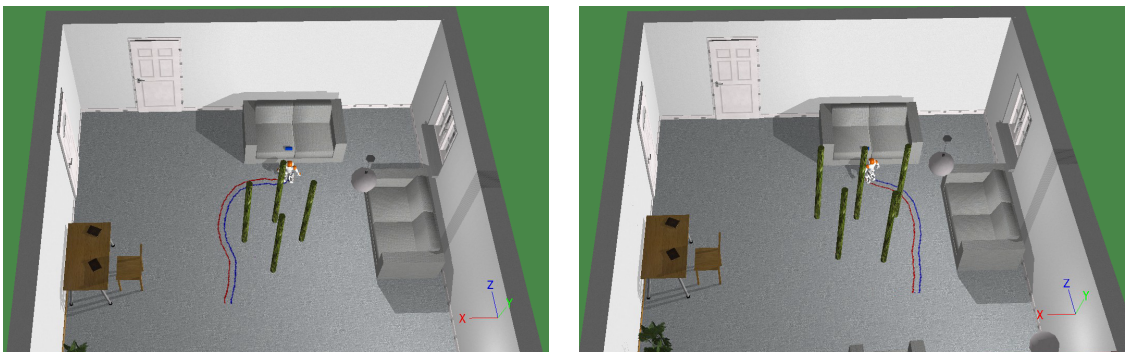


Figure 6.17 – Evaluation of motion consistency. During the approach the agent losses the view contact with the object of interest, so it has to rely on the temporarily persisted locations.

optic flow saliency processing. The robot was able to do the task despite the fact of counting on a reduced field-of-view sensory system (i.e. monocular vision), and not planning the motion in a cartographic representation of the scene. Through the persistence mechanism designed in the ego-cylinder, the desired behavior could be obtained even when the object temporarily left the field of view. Thus, motion coherence is observed through short-term persistence, a distinguishable aspect of behavior-based models.

As noticed on the right side of Fig. 6.15, though the inhibition of Reactive Motion, based on the heuristics consideration of the distance to the object, allowed the agent to do the task; the resulting trajectories after bypassing the obstacles were not necessarily the most efficient. This happened since the agent reacted to noisy obstacle detections, and also by the fact that Reactive Motion directed the robot away from the sofa, where the object of interest was. Furthermore, as shown in Fig. 6.17, the path taken may also

not have been the most efficient from the beginning of the task, so the robot could have gone through the columns instead of around them. This is because in the model the trajectory is largely dependent on the initial pose of the robot in relation to the obstacles, since motion is not planned but emerges on-line. In the next case study the possibility of improving these results is explored by defining alternative behavioral mode profiles in the Walk Fusion node, and by considering motion primitives to start the approach from different directions.

Finally, since objects were represented by particles in space, occasionally the robot's hand touched the columns when bypassing them. This can be controlled by assigning more weight to reaction in relation to the approach component of the walk, in order to increase the distance to obstacles. Thus, a compromise between producing safer but longer trajectories must be found. This seems to constitute a limitation of the representation chosen. Alternatively, the exploration of volumetric representations of the obstacle in the ego-space could motivate the study of more efficient trajectories. This aspect remains for future research.

CS-III: Learning-based approach

In order to improve the results obtained in the precedent case study, a more flexible scheme in Walk Fusion is studied. The idea is to arbitrate between different walk profiles, depending on the evaluation of the current state of the task. Hence, by exploiting the embodied aspect of behavior, the agent learns visual descriptors of the scene from kinesthetic demonstrations, that helps it to distinguish between the situations of free and blocked access to the object of interest. Accordingly, two walk profiles are defined by assigning different weights to Reactive and Approach Motion. In addition, by representing the task as a Markov Decision Process (MDP), more flexibility is obtained through the definition of motion primitives, so the robot can start the approach from different directions and learn the actions that produce the best performance. Learned policies are extended to new cases by case-based reasoning.

Visual encoding

A visual description of the scene is proposed as a means to select between distinct walk profiles. For this, the nodes Bottom-up Saliency, Stimuli Persistence, and Walk Fusion, of the model shown in Fig. 6 are modified, as detailed next.

Bottom-up Saliency

The output of this node is slightly changed, so instead of producing a binary segmentation, a ternary image M is obtained by applying two global threshold tests ϵ_1 and ϵ_2 to the magnitude of the measured flow $\bar{f}$ (see Eq. (3.40)). These thresholds correspond respectively to a high and a moderate flow condition. Thus,

$$M_i = \begin{cases} 2 \text{ if } \bar{f}_i > \epsilon_1 \\ 1 \text{ else if } \bar{f}_i > \epsilon_2 \\ 0 \text{ otherwise} \end{cases} . \tag{6.43}$$

The thresholds are defined so $\epsilon_1 > \epsilon_2$. They are set from kinesthetic demonstration. That

is, the robot is put to walk toward an obstacle placed in the saggital plane direction, and the mean flow magnitude is registered at two given proximities in relation to the object. Figure 6.18 illustrates the obtained segmentation by following this approach. Thereby, the output $\mathbf{u}$ of the behavior is changed so it includes the ternary image M obtained from Eq. (6.43), and the centroid of $M = 2$ (high flow saliency condition) conforming to Eq. (3.5). Similarly to the previous study case, this centroid is used by the Stimuli Persistence node to represent locations related to the obstacle, so the Reactive Motion behavior can produce a walk motion signal to avoid such locations.

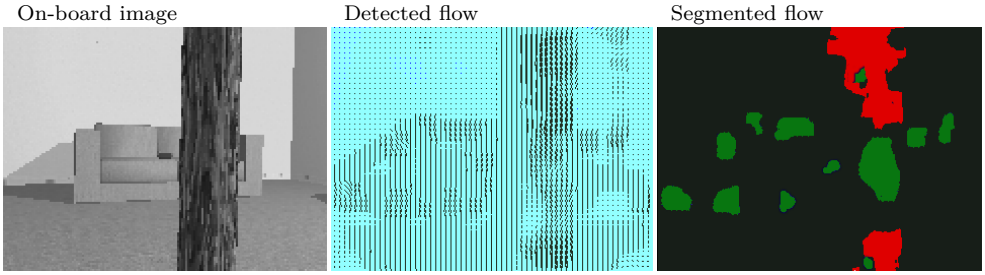


Figure 6.18 – Bottom-up segmentation. The red regions correspond to high flow, whereas the green regions correspond to moderate flow.

Stimuli Persistence

The purpose of this node is now twofold. It is in charge of persisting the ego-localization of stimuli (as defined previously), but also of fusing the information from Bottom-up and Top-down Saliency, in order to produce a visual encoding of the scene, as a means to assist the behavior arbitration conducted in the Walk Fusion node.

From the bottom-up segmentation of the optic flow into moderate and high intensity regions, and the binary mask obtained by top-down processing so the region that belongs to the object is identified; a scene encoding is proposed as an egocentric visual description of the task. As illustrated in Fig. 6.19, the current view is partitioned into three sectors $H = \{\mathrm{I}, \mathrm{II}, \mathrm{III}\}$. Three flow modalities associated to obstacles are: low 'o', moderate 'o̊', and high 'O'. Similarly, the flow associated to the object of interest is described by: low 'σ', moderate '$\mathring{\sigma}$', and high 'ς'. Since the tracked object is considered to be unique in the scene (due to the embodied filtering selection described in Sec. 5.4), in case it would span over more than one sector, it is heuristically assigned to the one containing the biggest proportion of the blob. Therefore, the scene is encoded by words of length 6 (i.e. 3 binomials), according to the regular expression (RegExp): $([\mathrm{o\mathring{o}O}]\sigma)^*(([\mathrm{o\mathring{o}O}][\sigma\mathring{\sigma}\varsigma])(?=.\sigma))([\mathrm{o\mathring{o}O}]\sigma)^*$. A total of 189 unique encodings can be obtained. Formally, the encoding for the sectors $h \in H$ is computed from the binary mask φ associated to the object of interest, and the ternary mask M (see Eq. (6.43)) associated to obstacles, according to the functions $g_{\mathrm{o}}(.)$ and $g_{\varsigma}(.)$, defined such that

$$g_{\mathrm{o}}(M,\varphi) = \begin{cases} \mathrm{O} & \text{if} \quad ((M=2)\wedge\neg\varphi) \\ \mathring{\mathrm{o}} & \text{else if} \quad ((M=1)\wedge\neg\varphi) \\ \mathrm{o} & \text{otherwise} \end{cases}, \tag{6.44}$$

$$g_{\varsigma}(M,\varphi) = \begin{cases} \varsigma & \text{if} \quad ((M=2)\wedge\varphi) \\ \mathring{\sigma} & \text{else if} \quad ((M=1)\wedge\varphi) \\ \sigma & \text{otherwise} \end{cases}. \tag{6.45}$$

Here $\wedge$ is the logical AND operator.

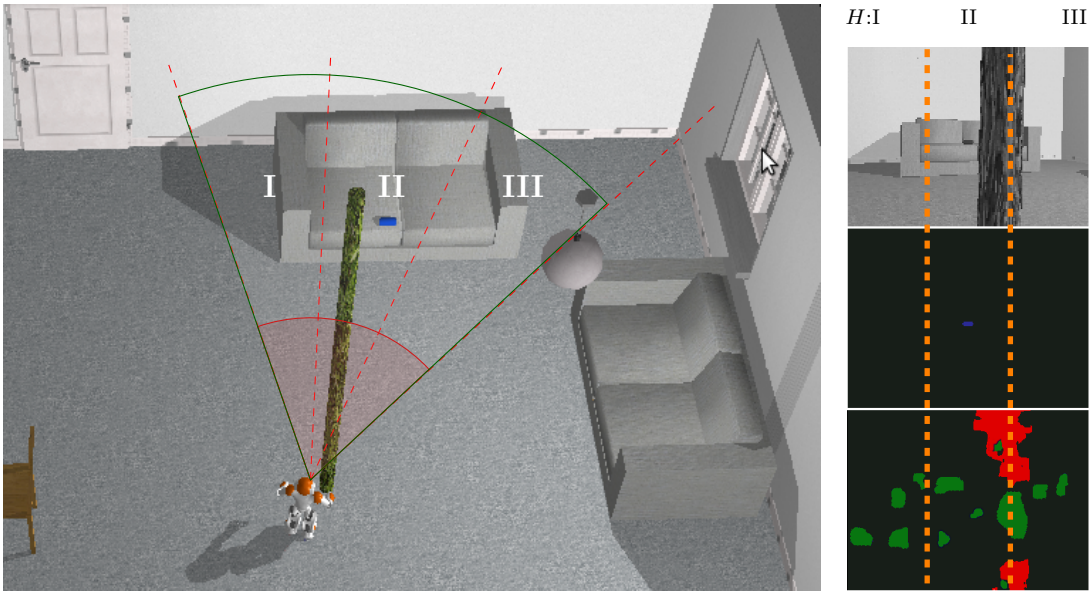


Figure 6.19 – Visual encoding. On the left the external view of the scene. The visualization cone is projected so the regions corresponding to the sectors $H = \{\mathrm{I}, \mathrm{II}, \mathrm{III}\}$ are shown. The region shaded in red would correspond approximately to the space associated to high optic flow. Similarly, the external green arc would delimit the space associated to moderate flow. On the right column the on-board views are given. The image at the middle corresponds to the top-down segmentation based on color, so the can is identified in the retinal area in blue. The image at the bottom is the ternary segmentation of the optic flow. The encoding obtained for the situation depicted are the pairs: "$\mathring{o}\sigma$" (moderate obstacle flow and low object flow in the sector I), "$O\mathring{o}$" (high obstacle flow and moderate object flow in the sector II), and "$O\sigma$" (high obstacle flow and low object flow in the sector III). Thereby, the word encoded is "$\mathring{o}\sigma O\mathring{o}O\sigma$".

To summarize, the output of the behavior is the vector $\mathbf{u}$ that contains, in addition to the active locations ${}^{B}\zeta$, ${}^{B}\tilde{o}$, and the associated forgetting factors γ_ζ and $\gamma_{\tilde{o}}$; the scene encoding produced.

Walk Fusion

In this node the arbitration between the three walk behaviors is produced. Inspired by the weighted fusion scheme described in Proetzsch et al. [152], the node signals are defined, such that

$$\mathbf{u} = \left(\frac{\sum_{j=0}^{k} \mathbf{w}_j a_j \mathbf{u}_j}{\sum_{l=0}^{k} \mathbf{w}_l a_l} \right), \; a = \left(\frac{\sum_{j=0}^{k} a_j^2}{\sum_{l=0}^{k} a_l} \right) \vartheta, \; r = \left(\frac{\sum_{j=0}^{k} a_j r_j}{\sum_{l=0}^{k} a_l} \right). \tag{6.46}$$

In the calculation of $\mathbf{u}$ the input activation vector $\mathbf{a}$ is pre-multiplied by the vector $\mathbf{w}$, that assigns different weight to each walk behavior (i.e. Approach, Reactive, and Search Motion).

As illustrated in Fig. 6.20 and described in Tab. 6.5, at each iteration a state-based arbitration scheme is processed in the state S_0, so a transition is produced to one of three

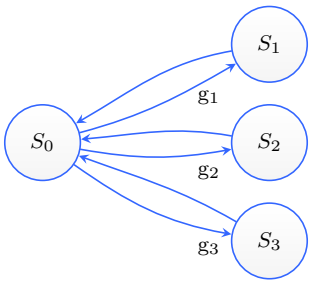


Figure 6.20 – State automate for discrete events arbitration between the motion profiles described in Tab. 6.5. The transition events are denoted by g_i, which are described in Tab. 6.6.

State	$\mathbf{w}_0$	$\mathbf{w}_1$	$\mathbf{w}_2$	Profile
S_0	-	-	-	Arbitration
S_1	0.00	0.75	0.25	Object search
S_2	0.80	0.20	0.00	Fast approaching
S_3	0.25	0.75	0.00	Obstacle avoidance

Table 6.5 – State arbitration profiles. The weights $\mathbf{w}_0$, $\mathbf{w}_1$, $\mathbf{w}_2$ are assigned respectively to the input from Approach Motion, Reactive Motion and Search Motion.

Event	Condition
g_1	The forgetting factor $\gamma_\zeta = 0$ (see Eq. (6.32)), that is to say the object is considered to be lost and not temporarily occluded.
g_2	The probability p of having a free access to the object, given the encoding transition detected, is $p > \epsilon_3$. The estimation of p is done conforming to Eq. (6.47).
g_3	$g_3 = \neg g_2$.

Table 6.6 – Arbitration events.

walk modes. Similarly to the previous definition of this behavior in Sec. 6.5.4, a switch from S_0 to S_1 occurs when Search Motion becomes active, due to the lost of the object. The novelty here is the distinction between walk modes that are adequate for the cases of having a free access and a blocked access to the object of interest. Thus, the idea of the state S_2 is to ensure a fast convergence to the object once there is no obstacle along the path. This is done by assigning less weight to the contribution of Reactive Motion in relation to Approach Motion, so the robot ideally neither reacts to noise nor avoids the location of interest. Contrarily, in the state S_3 more weight is assigned to the reactive component to avoid the obstacles.

As described in Tab. 6.6, the switch event g_2 is triggered based on the probabilistic evaluation of the scene encoding. Let Q be a random variable representing the fact of having a free access to the object of interest, and the event E denote the encoding transition binomial (a, b) occurring in the task. That is, the passage from a word descriptor a in time $t = k$ to a word descriptor b in time $t = k + 1$. From the a priori probability distribution $p(E|Q)$, obtained by kinesthetic demonstration, the desired a posteriori query $p(Q|E)$ is evaluated by applying the Bayes theorem. Thus,

$$p(Q|E) = \frac{p(E|Q)p(Q)}{p(E)}. \tag{6.47}$$

The learning of this probability is treated in the first experiment of the case study.

Learning motion primitives

The robot is given the possibility of choosing from five actions, that consist is starting the motion in a particular direction (see Fig. 6.21). This selection is modeled as a Markov Decision Process (MDP). As illustrated in Fig. 6.22, the transitions to be learned are enclosed in the blue box. The state set S is described in Tab. 6.7. Action selection is obtained based on the observation of the scene encoding. In this case the encoding is done previously to start the locomotion and not during the walk. Ideally an ocular saccade would be produced. Since it is not available to the Nao platform, the robot turns slowly the head while standing up to generate the optic flow. Given a particular scene encoding, the idea is to learn by reinforcement the transition from S_2 that produces the most reward.

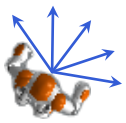


Figure 6.21 – Top view of the action primitives.

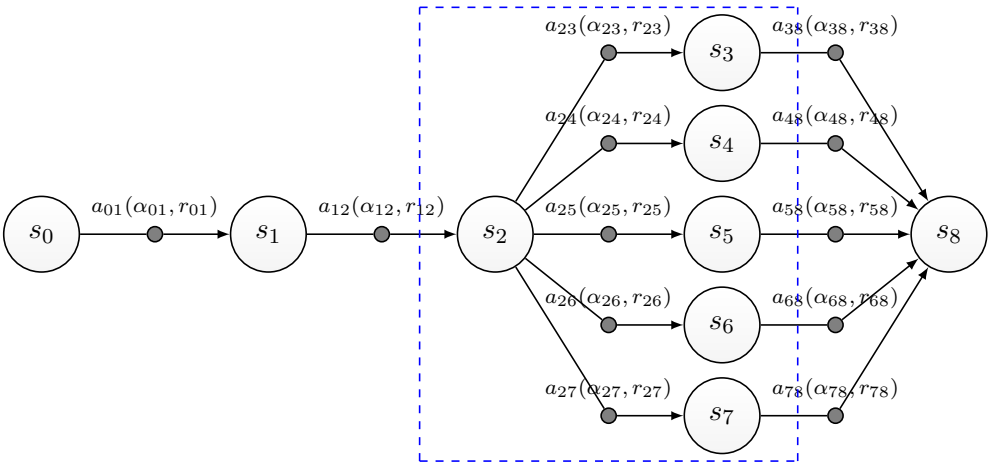


Figure 6.22 – MDP task model. The transition from s_i to s_j when taking the action a_{ij} is denoted by (α_{ij}, r_{ij}), where r_{ij} is the immediate reward and α_{ij} is the transition probability. From all actions there is a transition to s_8 (some are omitted for clarity), which model an abnormal end of the task with probability $1 - \alpha_{ij}$ and reward r_{i8}. The transitions to be learned are those departing from S_2 (which are delimited by the blue box). A detailed description of the states and rewards are provided in Tab. 6.7.

Learned policies from similar encodings can be heuristically used as an initial guess when facing new situations. The idea is to check whether there is a physically similar experience encoded from which a policy can be attempted. In case there is, the learned action is expressed according to the current view perspective, so it is tried in the current situation. The comparison on the similarity between the current encoding w_{t} and a learned encoding w_{r}, is obtained through the function $man(G(a, b))$, that gives the Manhattan distance (see Fig. 6.23) between two nodes a and b in a graph $G(a, b)$. The distance η between two encodings is defined by

State	Description
S_0: Start	Entry to the control program, in case the resources are available, an initialization signal is sent so the joints are activated and the robot stands up.
S_1: Encoding	The scene is encoded by slowly turning the head from left to right.
S_2: Selection	The robot choses a motion primitive to execute.
S_3: Walk-72	The robot turns 72 degrees clockwise and then walks in the saggital plane direction a distance ϵ_4 m.
S_4: Walk-36	The robot turns 36 degrees clockwise and then walks in the saggital plane direction a distance ϵ_4 m.
S_5: Walk-0	The robot walks in the saggital plane direction a distance ϵ_4 m.
S_6: Walk+36	The robot turns 36 degrees counter-clockwise and then walks in the saggital plane direction a distance ϵ_4 m.
S_7: Walk+72	The robot turns 72 degrees counter-clockwise and then walks in the saggital plane direction a distance ϵ_4 m.
S_8: End	Terminal state. Program ending requested by the user are penalized, so in case an interruption κ is produced $\kappa = 0$, otherwise $\kappa = 0$. Reward is also related to the dead reckoning estimate on the linear distance traveled $d = \sum_{t_0}^{t_f} \sqrt{x_t^2 + y_t^2}$, and the number n of times the object was out of the field of view. Therefore, the reward is $r_8 = \kappa\epsilon_5 + d + n\epsilon_6$.

Table 6.7 – MDP state descriptions. When not specified, the reward $r_i = 0$.

$$\eta = \sum_{h \in H} man(G(w_{\mathrm{t}(h)}, w_{\mathrm{r}(h)})). \tag{6.48}$$

Thus, two encodings are considered to be similar if $\eta < \epsilon_7$. Once a trial is finished, the case-based memory is updated with the learned Q-value.

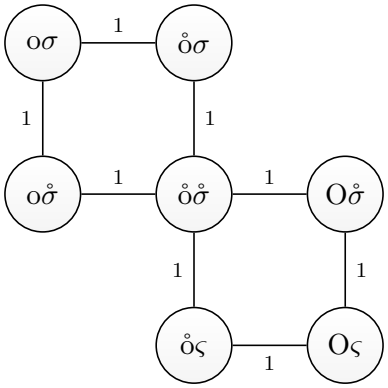


Figure 6.23 – Visual encoding neighborhood. Manhattan distance graph G.

Experiments

Four experiments were designed. As shown in Fig. 6.24, in the first experiment the robot is put to walk toward the object in two conditions: in one the path is cleared from obstacles, in the other it is not. Thus, the encoding transitions are registered from kinesthetic demonstration, in order to calculate the a posteriori probability distribution $p(Q|E)$, as defined in Eq. (6.47). In the second experiment the current implementation of the behavior is compared to the model defined in the precedent case study (see Sec.

6.5.4). For this, the task is repeated under the same conditions. In the third experiment multiple instances of the object of interest are added to increase the difficulty on the task. In the forth experiment, the RL-based approach is evaluated at distinct initial positions. Table 6.8 presents the parameters used in the model.

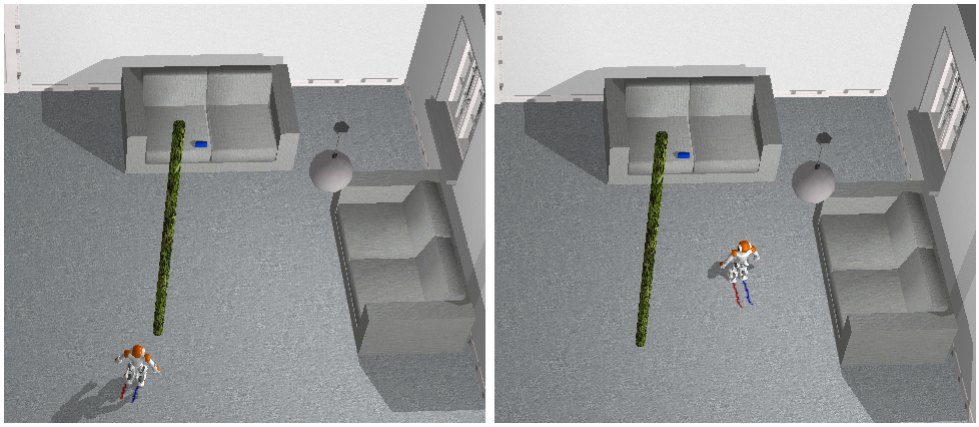


Figure 6.24 – Example of kinesthetic demonstrations for learning the scene encoding transitions. The robot walks towards the object. On the left a column is blocking the path, whereas on the right there is a free access to the object.

Id	Description	Value
ϵ_1	High flow threshold test	12.00
ϵ_2	Moderate flow threshold test	7.0
ϵ_3	Probabilistic test for state arbitration.	0.8
ϵ_4	Motion in the saggital plane induced by RL	0.15 m
ϵ_5	User program interruption cost	200
ϵ_6	Object lost iteration cost	0.1
ϵ_7	Manhattan distance threshold	4

Table 6.8 – CS-III task parameters.

Results

In the first experiment (see Fig. 6.25), the transitions produced were registered, so $p(E|Q)$ was estimated. After training, several tests where performed to assess the recognition of the event Q. The free access to the object was identified at a rate of 65.21% when this condition was tested (with $\epsilon_3 = 0.8$). On the contrary, the robot failed to recognize the condition of blocked access to the object at a rate of 1.07% . As shown in Fig. 6.26, failures can probably be explained by noisy detection of the optic flow from simulated images.

Figure 6.27 illustrates the results obtained for the second experiment. The differences are subtle, though the trajectory followed on the right (i.e. the Walk Fusion behavior is based on learned transitions from the scene encodings), was slightly more efficient. The results for the third experiment is shown in Fig. 6.28. It is noticed that despite the presence of other cans over the sofa, the robot was able to converge to the desired one. The result for the fourth experiment is presented in Fig. 6.29. Table 6.9 gives the roll-out reward obtained for the full set episode. In this case, the most efficient action was *do_Walk+36*, since no occlusions were produced. As shown in Fig. 6.30, the robot was

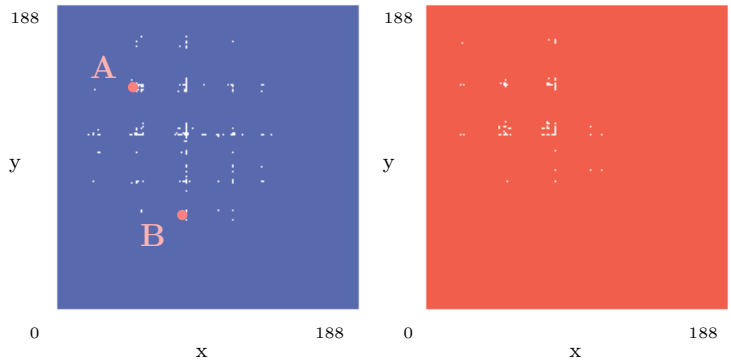


Figure 6.25 – Encoding transition matrix. The encodings are enumerated from 0 to 188, so a white dot in the matrix represents a transition between two indexes. The matrix on the left corresponds to the demonstration of a free access to the object. On the right, the transitions obtained when obstacles blocked the access to the object. Two transitions are shown. The location A corresponds to a transition from the encoding "$o\mathring{\sigma}o\mathring{\sigma}o\mathring{\sigma}$" to the encoding "$o\mathring{\sigma}o\mathring{\sigma}\mathring{o}\mathring{\sigma}$" (probably due to noise), whereas the location B corresponds to a transition from the encoding "$\mathring{o}\mathring{\sigma}o\mathring{\sigma}o\mathring{\sigma}$" to the encoding "$o\mathring{\sigma}o\sigma o\mathring{\sigma}$".

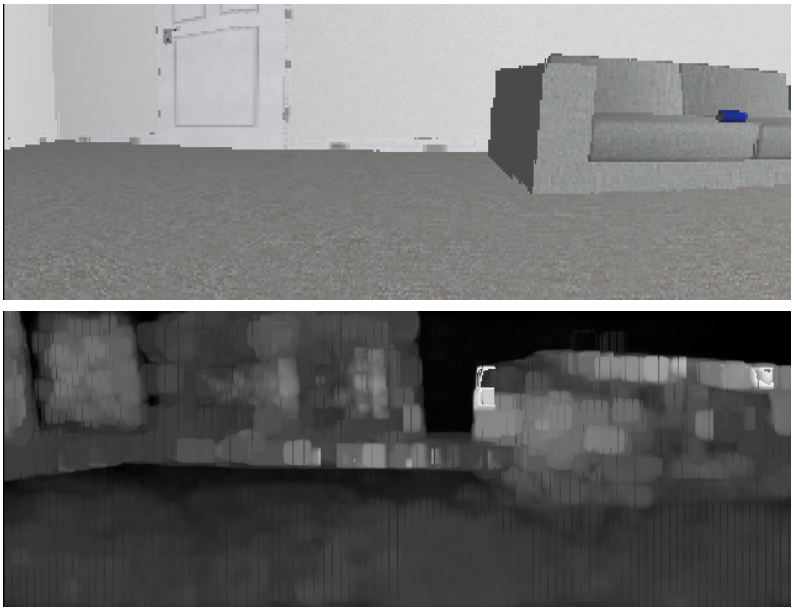


Figure 6.26 – Panoramic flow detection. On the top, the view of the scene at a range of π rad in the frontal ego-space. On the bottom, the normalized magnitude of the flow. Clearer regions should be phisically closer to the sensor than darker regions. As seen, noise from texture mapping in the simulated images affected the computation of the optic flow (e.g. the top-left corner of the sofa is perceived closer than the frontal part).

placed in an adjacent location so it applied the policy previously learned for a similar scene encoding. The trajectories obtained also were very similar. Finally, as shown in

Discussion

The experiments conducted have shown that the model is able to produce the desired behavior. In relation to the process of arbitration based on the scene encoding, smoother and more efficient trajectories were obtained, since the robot reacted less to noise. The fact of estimating the optic flow on simulated images, probably accounts for the relatively

Action	Distance	Occusions	Reward
a_{23}: *do_Walk-72*	4.13	85	12.63
a_{24}: *do_Walk-36*	3.92	83	12.22
a_{25}: *do_Walk-0*	3.81	58	9.61
a_{26}: *do_Walk+36*	3.60	0	3.60
a_{27}: *do_Walk+72*	4.08	62	10.80

Table 6.9 – Reward obtained for the full set of actions illustrated in Fig 6.29. The table shows the estimated distance walked in m and the occlusions registered. No interruptions were generated by the user in the trial.

Figure 6.27 – CS-II vs CS-III implementation. On the left, the result for CS-II where a single walk mode was available. On the right, the results for the current implementation, so a dual walk scheme based on visual encoding arbitration was available.

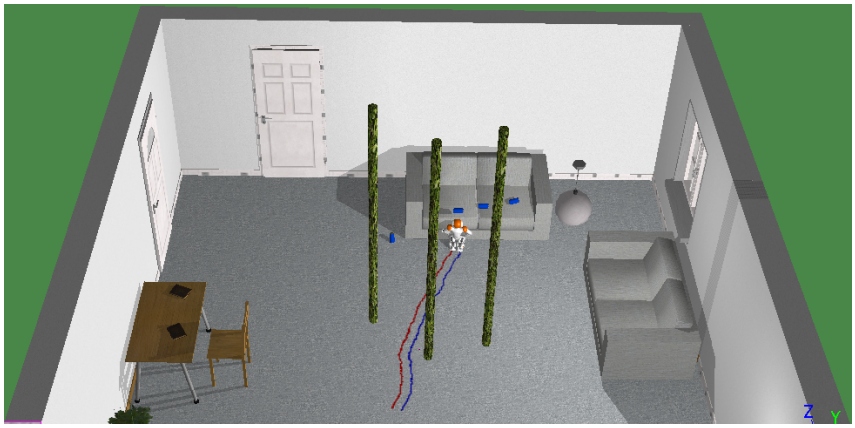


Figure 6.28 – Evaluation of the condition of multiple objects and obstacles. Despite the presence of other cans on the sofa, the robot was able to approach the desired one by relying on the embodied filtering process (described in Sec. 5.4).

low reliability obtained in the visual encoding process. Thus, the verification of these results with a real robot remains for future research.

In relation to the aspect of learning motion primitives by reinforcement, the robot was able to do the task and to efficiently reuse previous experiences in a similar situation. However, some limitations must be addressed. Given the stochastic nature of the behavior, it is possible that for real experiences the same action produces considerably different rewards (e.g. the robot may slip, so the estimate on the reward based on dead reckoning may be noisy). Additionally, the state reward did not consider important aspects such that the safety on the task. That is, a shorter path where the object is fully visible, may

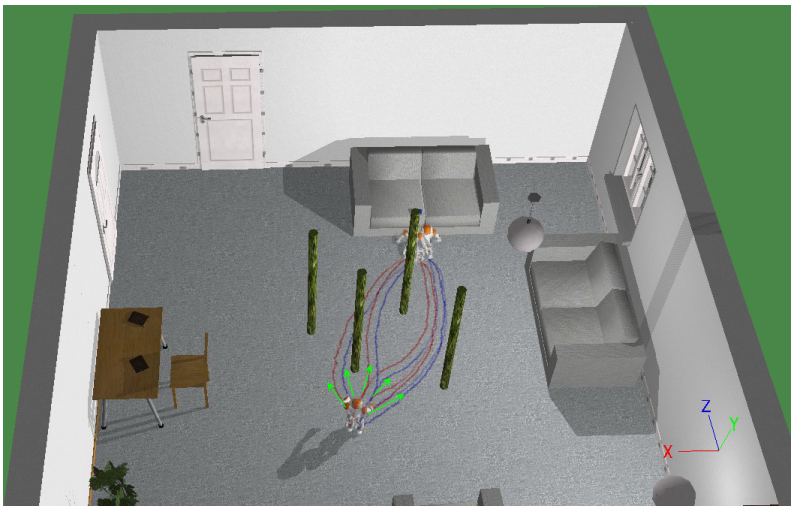


Figure 6.29 – RL policy learning. The available actions are shown in green. The robot walked 0.15 m along each of these directions before activating Approach Motion. The rest of the nodes operated normally, so reactivity to the obstacles and convergence on the task is obtained.

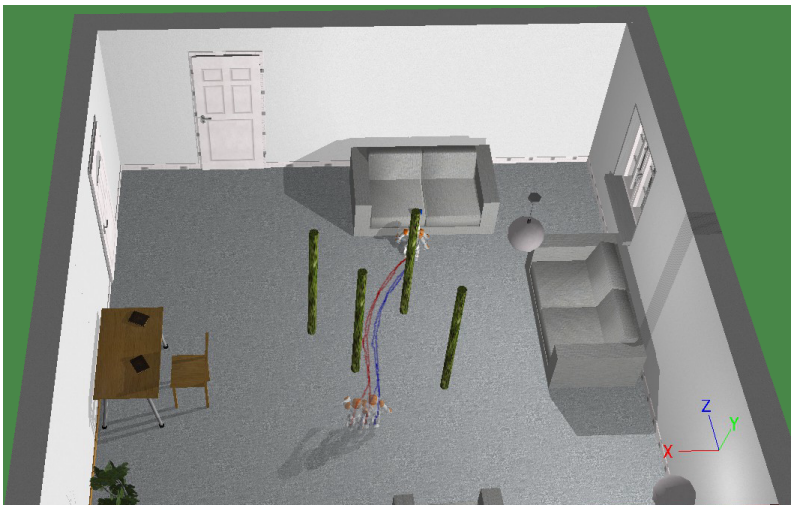


Figure 6.30 – CBR experience retrieval. The policy learned (*do_Walk+36*) at the left-most trial (see Fig. 6.29), was expressed with respect to the perspective of the agent on the right-most trial (*do_Walk-0*), so the knowledge acquired could be reused in the similar situation.

be for instance narrower so there is a greater risk for collision.

Conclusions

In the context of the action selection problem for the control of humanoid locomotion guided by vision, this chapter has started by a review on proposals available in the literature of behavior-based architectures. The choice for the iB2C framework to model the task was justified based on its generality and flexibility. Through a detailed implementation proposal, and several experiments; different topics were investigated, such as embodiment, knowledge representation, learning, and the adaptive aspect of behavior.

The results of the first study have shown that, by combining multiple image features that exploited embodiment and local heuristics, and in case a high acquisition rate is

available in the platform; the reactive model proposed can provide similar trajectories to those obtained in Sec. 5.5.5 (that were based on the observation of the localization from a disembodied representation of the object). This is consistent with the EC statements. Though, the heuristics used to estimate the heading direction of the object (i.e. the bilateral symmetry), did not produce the same results for a less symmetrical object. Thus, alternative representations are to be investigated. This is in fact a distinctive aspect of pure EC models, where the generalization of the solution is not ensured, even to small changes in the task specifications (which can be disadvantageous for service robotics applications). Therefore, although the methodology can produce efficient solutions from reduced modeling (a relatively rudimentary model was proposed), that comes with the price of poor generalization.

In the models proposed the philosophy adopted was to define individual behaviors in charge of specific aspects of the response. The operation of these nodes were mostly determined from kinesthetic demonstrations of the task, and relied on a local representation context. The agent accomplished the task efficiently without building a global representation, so the knowledge available was distributed in the architecture. The second and third case studies showed that behavior emerged and persisted enough in the absence of stimuli. The consistency on the behavior was provided by the perceptive ego-sensory that temporarily stored the localization of stimuli.

As the third case study has shown, visual information obtained from a first-person perspective was used to refine the behavior repertory, so the agent was able to learn actions and arbitrate between walk policies. Thus, the visual encoding proposed led to improvements in the execution of the task. Some aspects remain for future research. Although the stimuli representation chosen was efficient and convenient to ensure the reactivity on the task, it did not guarantee optimal behavior. This is particularly significant when representing the obstacles as particles in space. Alternative representations (e.g. a volume area in the ego-space) should be explored to study optimal behavior when avoiding obstacles, so the robot can decide between adopting a frontal or a saggital walking style, as human eventually do. It would also be interesting to evaluate the bottom-up processing proposed with a real robot capable of providing the required acquisition rate, so the arbitration based on visual encoding could be assessed.

7

Conclusions

In the study of service robotics applications for human-centered scenarios, the review of the state of the art of the research in humanoid robotics has suggested that, despite important achievements in the control of locomotion, manipulation, or adaptation; these robots have not reached a sufficient level of maturity as to become a viable technological solution. Although designed with an anthropomorphic body, there are important physical differences related to the kinematic properties, the sense organs, and the actuation system; that impose restrictions to humanoids. Thus, the field is still waiting for technological and scientific breakthroughs, to meet the requirements of reliable operation under unstructured scenarios.

In this work the architectural aspect of behavior was studied in the context of the action selection problem. Through a general case study, which is the fundamental skill of approaching and positioning in relation to stimuli guided by vision, several topics were explored, including: embodiment, visual attention, knowledge representation, egocentric localization and learning. The study focused on the processing of information from the visual and the proprioceptive sensory modalities, acquired on-board.

The study of visual attention showed that this process can be driven endogenously (by goals or top-down), or exogenously (by novelty or bottom-up). From a multidisciplinary perspective, the spotlight metaphor, the FIT and GS theories motivated the design of visual features and the structure of some of the behavior models proposed. The literature on machine vision was reviewed and relevant techniques were explored to extract information from images. The results of the case studies developed have shown that, although some structure may be recovered by heuristic clustering, the segmentation obtained may not be physically plausible. A supervised segmentation technique defined within a MRF framework was adapted for top-down saliency processing in a continuous image inflow. The evaluation suggested that it is a plausible approach for unstructured scenes, though the performance is degraded with metallic or reflective objects, or under excessive artificial illumination. The exploration of dense optical flow based on polynomial expansion has showed that some structure of the scene can be recovered (an estimate on the depth with respect to the sensor), by measuring the magnitude of the detected flow. Thus, unsupervised segmentation was obtained for textured objects.

A visual servoing scheme was studied for modeling the behavior of the robot. The PBVS and IBVS control approaches were employed simultaneously, in order to maintain the object of interest in the field of view, and to steer the robot to the desired 2D pose in relation to the object. The solution considered the motion primitives of walking and directing the head. The processing of the localization was based on the design of a sensory ego-cylinder, where the 3D position of the center of the object and its heading direction on the plane were represented. This information was obtained from a binary image and a rough 3D model of the object. Differently from previous contributions in the field, that relied on the principle of Verification Vision, the MRF segmentation technique was used for localizing the object, so the top-down saliency processing does not require of knowledge on the spatial motion of the sensor. This is advantageous since the solution operates at a low acquisition rate. The model was tested both in simulation and a real experiment. The results showed that it is a plausible strategy to approaching convex colored objects on the environment.

The computational complexity and the reliability on the localization parameters are related to the definition of the measurement and the representation frames of reference. Thus, different locations for the sensory ego-cylinder were studied. Given the lack of consensus in the literature about the placement of the ego-sensory structure, body- and eye-centered locations were investigated. The results of the experiments suggested that converge can be obtained for body-centered locations (i.e. the measurement and the representation frame are defined at different placements). The fact that the robot walked in vertical posture constrained the mobility of the reference system, so heuristic placements also provided convergence. This was not the case for eye-centered placements (i.e. the measurement and the representation frame are defined in the same location). The visual system is constantly redirected toward the object of interest by the look-at task, so the local context of the body posture during the task is not preserved. A hybrid solution was proposed, so the correction in the position is determined eye-centered, but the regulation on the angular motion is calculated body-centered. This combination provided correct results, it was computationally more efficient, and less affected by noise in the proprioceptive measurements.

From the analysis of the dynamic aspects of human locomotion guided by vision, a first-order description HMW was proposed for approaching the object. HMW allowed the agent to mimic the human walking style. That is, non-holonomic motion is used when the individual is far from the object, but holonomic motion is preferred when the individual is close enough to the goal. A contribution of HMW is to consider a desired 2D pose in relation to the object, whereas previous studies focused only on the control of the position component. This is of crucial importance, since the operational face of the stimulus is taken into account in the motion, so the path followed is more efficient and aesthetic. These are valued aspects for the acceptance of the solution in the context of human-machine interaction and service robotics applications.

The six-steps methodology developed to design reliable solutions illustrated an interesting combination between the cognitivist and the EC research. As mentioned previously, the visual selection mechanism proposed was inspired by the information processing models of attention. It was also based on a Bayesian Network structure, which is usually employed for information fusion and knowledge representation in the context of the cognitivist research in AI. However, in the BN multi-sensory information is fused from features that exploited embodiment, so they were carefully defined from the EC perspective. Furthermore, the anticipative aspect of the behavior scheme was an interesting

opportunity to study the effect of the statistical regularities induced by the coupling, and the information redundancy in the sensory-motor coordination.

The BN structure designed provided reliable information about the degree of confidence and the discriminative power of the attention selection mechanism. This became a significant contribution to the autonomy of the agent, through the efficient use of available resources. Thus, the solution was operational at a low acquisition rate in a low-cost robot. The static policies for the BN were adequate to real tasks, whereas the advantages of the dynamic policies were noted only in simulations, given the observation noise and the lack or redundancy in the information provided by the features. The BN also grounded the implementation of the hybrid architecture proposed to ensure the safety when accessing remote resources.

In the behavior-based models proposed the philosophy adopted was to define individual behaviors in charge of specific aspects of the response. The operation of the nodes were mostly determined from kinesthetic demonstrations of the task, which is very convenient to robotic service applications. The nodes rely on a local representation context, so the task is efficiently accomplished without building a global representation. In the study of reactive walk, by defining action-oriented representations of the object, similar trajectories were obtained to the scheme using a disembodied representation of the object; which is consistent with the EC statements. Though, the heuristics considered did not produce the same results for a less symmetrical object. This illustrated a distinctive aspect of pure EC models, where the generalization of the solution is not ensured, even to small changes in the task specifications (which can be disadvantageous for service robotics applications).

In the action selection problem, visual information was used to refine the behavior repertory of the robot, so it was able to learn actions and arbitrate between walk policies. Thus, the visual encoding proposed led to improvements in the execution of the task. In the models behavior emerged and persisted enough in the absence of stimuli, given the temporal storage of information in the perceptive ego-sensory. The models have illustrated a potential and feasible strategy that can be adopted for prototyping and exploring more complex sensory-motor coordinations. Thus, the fact of counting on modular motion primitives that are already available to the agent, handles much of the security aspects involved in the task, as for example, maintaining the body balance.

Research perspectives

In the studies conducted several aspects remained for future research. The fact of considering a static scene is restrictive to applications in service robotics, so the approach to moving objects should be studied. Time constrained tasks could also be explored (e.g. approaching an object in motion, or avoiding a moving obstacle). The models considered the case of walking at a constant velocity profile, so a deterministic predictive model was sufficient to obtain the desired results. It is important to notice that this assumption may not hold when the objects move, or when the robot walks faster or runs. Thereby, stochastic models of motion could be considered in the task, notably when the acquisition rate is low.

The processing of top-down saliency was based on color features. Depending on the surface of the object, disturbances, or illumination noise; momentary degradations where produced in the segmentation. Thus, features redundancy (e.g. the image con-

tours, edges, among others) could be integrated to the model to increase the reliability of the task. Likewise, other sensory modalities available in humanoids (e.g. sonars, laser range, binocular vision) could also be included. In a context of feature redundancy, the advantages of the dynamic policies for the BN could be observed.

Although the stimuli representation chosen was efficient and convenient to ensure the reactivity on the task, it did not guarantee optimal behavior. This is particularly significant when representing the obstacles as particles in the ego-space. Alternative representations (e.g. a volume area) should be explored to study optimal behavior when avoiding obstacles, so the robot can for instance decide between adopting a frontal or a saggital walking style, as human eventually do. It would also be interesting to evaluate the bottom-up processing proposed with a real robot capable of providing the required acquisition rate, so the arbitration based on visual encoding could be assessed.

List of Publications

International Journals

- H F Chame, C Chevallereau. *Grounding humanoid visually guided walking: from action-independent to action-oriented knowledge.* DOI 10.1016/j.ins.2016.02.053. Information Sciences, ElSevier, 2016 [38].

Book Chapters

- H F Chame, C Chevallereau. *Sensory-motor anticipation and local information fusion for reliable humanoid approach.* New Trends in Medical and Service Robots of the series Mechanisms and Machine Science, 39. DOI: 10.1007/978-3-319-30674-2_10. Springer International Publishing Switzerland 2016 [35].
- H F Chame, P Martinet. *Cognitive modeling for automating learning in visually-guided manipulative tasks.* Informatics in Control, Automation and Robotics, Lecture Notes in Electrical Engineering 325, DOI: 10.1007/978-3-319-10891-9_2. Springer International Publishing Switzerland 2015 [40].

International Conferences

- H F Chame, C Chevallereau. *A Top-Down and Bottom-Up Visual Attention Model for Humanoid Object Approaching and Obstacle Avoidance.* XIII Latin American Robotics Symposium and IV Brazilian Robotics Symposium (LARS/SBR), Recife, Brazil, 2016. DOI: 10.1109/LARS-SBR.2016.12, IEEE [37].
- H F Chame, C Chevallereau. *Sensory-motor anticipation and local information fusion for reliable humanoid approach.* 4th international Workshop on Medical and Service Robots (Mesrob2015). Nantes, France.
- H F Chame, C Chevallereau. *Embodied localization in visually-guided walk of humanoid robots.* 11th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2014). DOI: 10.5220/0005063001650174 [36].
- H F Chame, P Martinet. *A computational cognition and visual servoing based methodology to design automatic manipulative tasks.* 10th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2013). DOI: 10.5220/0004480802130220 [39].

National Conferences

- H F Chame, C Chevallereau. *Towards assistance robotics through embodiment.* Journées Nationales de la Robotique Humanoïde et des Architectures de Contrôle en Robotique, 23-24 Juin 2014 à la Cité Internationale de Paris.

Regional Conferences

- H F Chame, C Chevallereau. *On-board visually-guided walk of humanoid robots.* Journée de doctorants (JDoc). l'École des mines de Nantes, 20 mars 2014.

Bibliography

[1] Ackerman, E. G. a. E. DARPA Robotics Challenge Finals: Know Your Robots, June 2015. 14

[2] Agüero, C., Cañas, J., Martín, F., and Perdices, E. Behavior-based iterative component architecture for soccer applications with the nao humanoid. In *5th Workshop on Humanoids Soccer Robots. Nashville, TN, USA* (2010). 127

[3] Albus, J. S., and Proctor, F. G. A reference model architecture for intelligent hybrid control systems. In *Proceedings of the International Federation of Automatic Control* (San Francisco, CA, 1996). 27

[4] Allen, B. F., Picon, F., Dalibard, S., Magnenat-Thalmann, N., and Thalmann, D. Localizing a mobile robot with intrinsic noise. In *3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), 2012* (Oct 2012), pp. 1–4. 19

[5] Ambrose, R., Bekey, G., Kumar, V., Sanderson, A., and Wilcox, B. Wtec panel report on international assessment of research and development in robotics. Technical report, National Science Foundation, Washington, DC. World Technology Evaluation Center, Baltimore, MD, 2007. 12

[6] Anderson, M. Embodied cognition: A field guide. *Artificial Intelligence 149*, 1 (2003), 91–130. 94

[7] Arbib, M. A., Metta, G., and van der Smagt, P. Neurorobotics: From vision to action. In *Springer Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Springer Berlin Heidelberg, Jan. 2008, pp. 1453–1480. 100

[8] Arkin, R. Motor schema based navigation for a mobile robot: An approach to programming by behavior. In *Robotics and Automation. Proceedings. 1987 IEEE International Conference on* (Mar 1987), vol. 4, pp. 264–271. 29

[9] Arkin, R. C., Fujita, M., Takagi, T., and Hasegawa, R. Ethological modeling and architecture for an entertainment robot. In *Proceedings of the 2001 IEEE International Conference on Robotics and Automation, ICRA 2001, May 21-26, 2001, Seoul, Korea* (2001), pp. 453–458. 126

[10] Armbrust, C., Kiekbusch, L., and Berns, K. Using behaviour activity sequences for motion generation and situation recognition. In *ICINCO (2)* (2011), J.-L. Ferrier, A. Bernard, O. Y. Gusikhin, and K. Madani, Eds., SciTePress, pp. 120–127. 131

[11] Armbrust, C., Proetzsch, M., and Berns, K. Behaviour-based off-road robot navigation. *KI - Künstliche Intelligenz 25*, 2 (May 2011), 155–160. 128

[12] Arons, B. A review of the cocktail party effect. *Journal of the American Voice I/O Society 12* (1992), 35–50. 35

[13] Asimov, I. *I, Robot*, 1 ed. Gnome Press, New York, 1950. 1, 2

[14] Baier-Lowenstein, T., and Zhang, J. Learning to grasp everyday objects using reinforcement-learning with automatic value cut-off. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on* (Oct 2007), pp. 1551–1556. 133

[15] Barto, A. G., and Mahadevan, S. Recent advances in hierarchical reinforcement learning. *Discrete Event Dynamic Systems 13*, 1-2 (Jan. 2003), 41–77. 133

[16] Beauchemin, S. S., and Barron, J. L. The computation of optical flow. *ACM Comput. Surv. 27*, 3 (Sept. 1995), 433–466. 46, 47

[17] Behnke, S. Humanoid robots - from fiction to reality? *KI 22*, 4 (2008), 5–9. 2, 10, 18, 20

[18] Berns, K., and Hirth, J. Control of facial expressions of the humanoid robot head roman. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on* (Oct 2006), pp. 3119–3124. 128

[19] Besag, J. On the Statistical Analysis of Dirty Pictures. *Journal of the Royal Statistical Society. Series B (Methodological) 48*, 3 (1986), 259–302. 53

[20] Bodiroza, S., Schillaci, G., and Hafner, V. Robot ego-sphere: An approach for saliency detection and attention manipulation in humanoid robots for intuitive interaction. In *2011 11th IEEE-RAS International Conference on Humanoid Robots (Humanoids)* (2011), pp. 689–694. 73

[21] Bolles, R. C. Verification vision for programmable assembly. In *Proceedings of the 5th International Joint Conference on Artificial Intelligence - Volume 2* (San Francisco, CA, USA, 1977), IJCAI'77, Morgan Kaufmann Publishers Inc., pp. 569–575. 46, 101

[22] Bonasso, R. P., Kortenkamp, D., and Whitney, T. Using a robot control architecture to automate space shuttle operations. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence and Ninth Conference on Innovative Applications of Artificial Intelligence* (1997), AAAI'97/IAAI'97, pp. 949–956. 30

[23] Bradski, G., and Kaehler, A. *Learning OpenCV: Computer Vision in C++ with the OpenCV Library*, 2nd ed. O'Reilly Media, Inc., 2013. 46, 47

[24] Bradski, G. R. Computer Vision Face Tracking For Use in a Perceptual User Interface. *Intel Technology Journal* (1998), 214–219. 46

[25] Breazeal, C. *Designing Sociable Robots*. MIT Press, Cambridge, MA, USA, 2002. 20

[26] Broadbent, D. E. *Perception and communication*. Pergamon Press, New York, NY, 1958. 34

[27] Brooks, R. A robust layered control system for a mobile robot. *Robotics and Automation, IEEE Journal of 2*, 1 (Mar 1986), 14–23. 28, 126, 131

[28] Brooks, R., Breazeal, C., Marjanović, M., Scassellati, B., and Williamson, M. The cog project: Building a humanoid robot. In *Computation for Metaphors, Analogy, and Agents*, C. Nehaniv, Ed., vol. 1562 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 1999, pp. 52–87. 10

[29] Brooks, R. A. *Cambrian Intelligence: The Early History of the New AI*, 1 edition ed. A Bradford Book, Cambridge, Mass., July 1999. 4, 27, 29, 94, 99, 101, 144

[30] Burgess, N. Spatial cognition and the brain. *Annals of the New York Academy of Sciences 1124* (Mar. 2008), 77–97. 24

[31] Burghart, C., Mikut, R., Stiefelhagen, R., Asfour, T., Holzapfel, H., Steinhaus, P., and Dillmann, R. A cognitive architecture for a humanoid robot: a first approach. In *Humanoid Robots, 2005 5th IEEE-RAS International Conference on* (Dec 2005), pp. 357–362. 126

[32] Canny, J. A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on PAMI-8*, 6 (Nov 1986), 679–698. 45

[33] Carone, G., and Costello, D. ¿Llega Europa a la tercera edad?: la UE debe tomar en serio las proyecciones recientes que indican que el envejecimiento de la población tendrá un gran impacto económico y presupuestario. *Finanzas y desarrollo: publicación trimestral del Fondo Monetario Internacional y del Banco Mundial 43*, 3 (2006), 28–31. 2

[34] Cave, K. R., and Bichot, N. P. Visuospatial attention: beyond a spotlight model. *Psychonomic Bulletin & Review 6*, 2 (1999), 204–23. 36

[35] Chame, H. F., and Chevallereau, C. Sensory-motor anticipation and local information fusion for reliable humanoid approach. In *New Trends in Medical and Service Robots: Human Centered Analysis, Control and Design*, P. Wenger, C. Chevallereau, D. Pisla, H. Bleuler, and A. Rodić, Eds. Springer International Publishing. 167

[36] Chame, H. F., and Chevallereau, C. Embodied localization in visually-guided walk of humanoid robots. In *ICINCO (2)* (2014), pp. 165–174. 167

[37] Chame, H. F., and Chevallereau, C. A Top-Down and Bottom-Up Visual Attention Model for Humanoid Object Approaching and Obstacle Avoidance. In *2016 XIII Latin American Robotics Symposium and IV Brazilian Robotics Symposium (LARS/SBR)* (Recife, Brazil, Oct 2016). 167

[38] Chame, H. F., and Chevallereau, C. Grounding humanoid visually guided walking: From action-independent to action-oriented knowledge. *Information Sciences 352–353* (2016), 79 – 97. 167

[39] Chame, H. F., and Martinet, P. A computational cognition and visual servoing based methodology to design automatic manipulative tasks. In *Proceedings of the 10th International Conference on Informatics in Control, Automation and Robotics - Volume 1: ICINCO,* (2013), INSTICC, SciTePress, pp. 213–220. 167

[40] Chame, H. F., and Martinet, P. Cognitive modeling for automating learning in visually-guided manipulative tasks. In *Informatics in Control, Automation and Robotics*, J.-L. Ferrier, O. Gusikhin, K. Madani, and J. Sasiadek, Eds., vol. 325 of *Lecture Notes in Electrical Engineering*. 2015. 133, 167

[41] Chapman, D. Planning for conjunctive goals. *Artif. Intell. 32*, 3 (July 1987), 333–377. 26

[42] Chaumette, F., and Hutchinson, S. Visual servo control, part i: Basic approaches. *IEEE Robotics and Automation Magazine 13* (2006), 82–90. 67, 69

[43] Chaumette, F., and Hutchinson, S. Visual servo control, part ii: Advanced approaches. *IEEE Robotics and Automation Magazine 14*, 1 (March 2007), 109–118. 69

[44] Cherry, C. E. Some Experiments on the Recognition of Speech, with One and with Two Ears. *Journal of the Acoustical Society of America 25*, 5 (1953), 975–979. 35

[45] Chevallereau, C., Abba, G., Aoustin, Y., Plestan, F., Westervelt, E., Canudas-de Wit, C., and Grizzle, J. Rabbit: a testbed for advanced control theory. *Control Systems, IEEE 23*, 5 (Oct 2003), 57–79. 15

[46] Cho, H., Ed. *Opto-Mechatronic Systems Handbook: Techniques and Applications*. CRC Press, 2002. 38, 43, 44

[47] Clark, A. *Being There: Putting Brain, Body, and World Together Again*, reprint edition ed. A Bradford Book, Cambridge, Mass., Jan. 1998. 95, 99

[48] Collins, S., Ruina, A., Tedrake, R., and Wisse, M. Efficient bipedal robots based on passive-dynamic walkers. *Science (New York, N.Y.) 307*, 5712 (Feb. 2005), 1082–5. 19

[49] Comport, A., Marchand, E., Pressigout, M., and Chaumette, F. Real-time markerless tracking for augmented reality: the virtual visual servoing framework. *Visualization and Computer Graphics, IEEE Transactions on 12*, 4 (July 2006), 615–628. 46, 66

[50] Conde, L., Gabriel, B., and Nunes, P. U. A behavior based fuzzy control architecture for path tracking and obstacle avoidance. In *Proceedings of the 5th Portuguese Conference on Automatic Control (Controlo 2002)* (Aveiro, Portugal, 2002), pp. 341–346. 126

[51] Corke, P. *Robotics, Vision and Control: Fundamental Algorithms in MATLAB*. Springer Science & Business Media, Nov. 2011. 10

[52] Corke, P. I. *Visual control of robots : high-performance visual servoing*. Robotics and mechatronics series. Research Studies Press ; Wiley, 1996. 38, 39, 40, 43, 44, 45

[53] Cypher, A., Ed. *Watch What I Do – Programming by Demonstration*. MIT Press, Cambridge, MA, USA, 1993. 21

[54] Deneve, S., and Pouget, A. Bayesian multisensory integration and cross-modal spatial links. *Journal of physiology, Paris 98*, 1-3 (2004), 249–258. 105

[55] Deng, H., and Clausi, D. Unsupervised image segmentation using a simple MRF model with a new implementation scheme. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004* (2004), vol. 2, pp. 691–694 Vol.2. 53

[56] Doggaz, N., and Ferjani, I. Image segmentation using normalized cuts and efficient graph-based segmentation. In *Proceedings of the 16th International Conference on Image Analysis and Processing - Volume Part II* (Berlin, Heidelberg, 2011), ICIAP'11, Springer-Verlag, pp. 229–240. 49

[57] Duda, R. O., and Hart, P. E. Use of the hough transformation to detect lines and curves in pictures. *Commun. ACM 15*, 1 (Jan. 1972), 11–15. 46

[58] Duda, R. O., Hart, P. E., and Stork, D. G. *Pattern Classification (2Nd Edition)*. Wiley-Interscience, 2000. 49

[59] Dune, C., Herdt, A., March, E., Stasse, O., Wieber, P.-b., and Yoshida, E. Vision based control for humanoid robots. In *in "IROS Workshop on Visual Control of Mobile Robots (ViCoMoR* (2011). 66

[60] Dune, C., Herdt, A., Stasse, O., Wieber, P. B., Yokoi, K., and Yoshida, E. Cancelling the sway motion of dynamic walking in visual servoing. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2010), pp. 3175–3180. 84

[61] El-Bagoury, B. M., Salem, A.-B. M., and Burkhard, H.-D. Hierarchical case-based reasoning behavior control for humanoid robot. *An. Univ. Craiova, Ser. Mat. Inf. 36*, 2 (2009), 131–140. 127

[62] Elkmann, N., Fritzsche, M., and Schulenburg, E. Tactile sensing for safe physical human-robot interaction. In *4th International Conference on Advances in Computer-Human Interactions* (February 23- 2011), vol. 3, pp. 212–217. 19

[63] Ertel, W. *Introduction To Artificial Intelligence*. Springer London Ltd, Mar. 2011. 48, 105, 106, 109, 134

[64] Exner, D., Bruns, E., Kurz, D., Grundhofer, A., and Bimber, O. Fast and robust CAMShift tracking. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (2010), pp. 9–16. 46

[65] Fajen, B. R., and Warren, W. H. Behavioral dynamics of steering, obstacle avoidance, and route selection. *Journal of Experimental Psychology. Human Perception and Performance 29*, 2 (Apr. 2003), 343–362. 96

[66] Fanello, S. R., Pattacini, U., Gori, I., Tikhanoff, V., Randazzo, M., Roncone, A., Odone, F., and Metta, G. 3d stereo estimation and fully automated learning of eye-hand coordination in humanoid robots. In *14th IEEE-RAS International Conference on Humanoid Robots, Humanoids 2014, Madrid, Spain, November 18-20, 2014* (2014), pp. 1028–1035. 22

[67] Farnebäck, G. *Polynomial Expansion for Orientation and Motion Estimation*. PhD thesis, Linköping University, Sweden, SE-581 83 Linköping, Sweden, 2002. Dissertation No 790, ISBN 91-7373-475-6. 56

[68] Farnebäck, G. Two-frame motion estimation based on polynomial expansion. In *Proceedings of the 13th Scandinavian Conference on Image Analysis* (Berlin, Heidelberg, 2003), SCIA'03, Springer-Verlag, pp. 363–370. 47, 55, 57, 59

[69] Ferland, F., Cruz-Maya, A., and Tapus, A. Adapting an hybrid behavior-based architecture with episodic memory to different humanoid robots. In *Proceedings of the 24th IEEE International Symposium on Robot and Human Interactive Communication* (Kobe, Japan, 2015), IEEE Computer Society. 127

[70] Ferreira, H., Casañas, R., Ramos, E., Fernándes, V., and Núñez, H. Segmentación y descripción automática de espermatozoides humanos a partir de imágenes. In *XXXV Conferencia Latinoamericana de Informática. CLEI 2009. Pelotas, Brasil* (2009), Actas XXXV CLEI. 45

[71] Fujita, M., Kuroki, Y., Ishida, T., and Doi, T. Autonomous behavior control architecture of entertainment humanoid robot sdr-4x. In *Intelligent Robots and Systems, 2003. (IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on* (Oct 2003), vol. 1, pp. 960–967 vol.1. 126, 146

[72] García, J. F., Rodríguez, F. J., Martín, F., and Matellán, V. Using visual attention in a nao humanoid to face the robocup any-ball challenge. In *5th Workshop on Humanoids Soccer Robots @ Humanoids 2010* (Nashville, TN, USA, 2010), pp. 1–6. 37

[73] Gibson, J. J. *The Perception of the Visual World*. Houghton Mifflin, Boston, MA, 1950. 46

[74] Gonzalez, R. C., and Woods, R. E. *Digital Image Processing (3rd Edition)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2006. 38, 40, 41, 46, 47

[75] Gouaillier, D., Hugel, V., Blazevic, P., Kilner, C., Monceaux, J., Lafourcade, P., Marnier, B., Serre, J., and Maisonnier, B. The nao humanoid: a combination of performance and affordability. *CoRR* (2008). 15, 16, 17, 18

[76] Graziano, M. S. Is reaching eye-centered, body-centered, hand-centered, or a combination? *Reviews in the neurosciences 12*, 2 (2001), 175–185. 24

[77] Halámek, J., Viščor, I., and Kasal, M. Dynamic range and acquisition system. *Measurement Science Review 1*, 1 (2001), 71–74. 40

[78] Harris, C., and Stephens, M. A combined corner and edge detector. In *In Proc. of Fourth Alvey Vision Conference* (1988), pp. 147–151. 47

[79] Hashimoto, S., Narita, S., Kasahara, H., Shirai, K., Kobayashi, T., Takanishi, A., Sugano, S., Yamaguchi, J., Sawada, H., Takanobu, H., Shibuya, K., Morita, T., Kurata, T., Onoe, N., Ouchi, K., Noguchi, T., Niwa, Y., Nagayama, S., Tabayashi, H., Matsui, I., Obata, M., Matsuzaki, H., Murasugi, A., Kobayashi, T., Haruyama, S., Okada, T., Hidaki, Y., Taguchi, Y., Hoashi, K., Morikawa, E., Iwano, Y., Araki, D., Suzuki, J., Yokoyama, M., Dawa, I., Nishino, D., Inoue, S., Hirano, T., Soga, E., Gen, S., Yanada, T., Kato, K., Sakamoto, S., Ishii, Y., Matsuo, S., Yamamoto, Y., Sato, K., Hagiwara, T., Ueda, T., Honda, N., Hashimoto, K., Hanamoto, T., Kayaba, S., Kojima, T., Iwata, H., Kubodera, H., Matsuki, R., Nakajima, T., Nitto, K., Yamamoto, D., Kamizaki, Y., Nagaike, S., Kunitake, Y., and Morita, S. Humanoid robots in waseda university—hadaly-2 and wabian. *Autonomous Robots 12*, 1 (2002), 25–38. 11

[80] Hauagge, D. C. Image matching using local symmetry features. In *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Washington, DC, USA, 2012), CVPR '12, IEEE Computer Society, pp. 206–213. 139

[81] Hawes, N., and Wyatt, J. Engineering intelligent information-processing systems with {CAST}. *Advanced Engineering Informatics 24*, 1 (2010), 27–39. Informatics for cognitive robots. 127

[82] Hirose, M., and Ogawa, K. Honda humanoid robots development. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences 365*, 1850 (Jan. 2007), 11–19. 10, 11

[83] Hoffmann, F. An overview on soft computing in behavior based robotics. In *Fuzzy Sets and Systems — IFSA 2003*, T. Bilgiç, B. De Baets, and O. Kaynak, Eds., vol. 2715 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2003, pp. 544–551. 130

[84] Hoffmann, M., and Pfeifer, R. The implications of embodiment for behavior and cognition: animal and robotic case studies. *CoRR abs/1202.0440* (2012). 94

[85] Horn, B. K. P., and Schunck, B. G. Determining optical flow. *Artificial Intelligence 17* (1981), 185–203. 47

[86] Hornung, A., Wurm, K., and Bennewitz, M. Humanoid robot localization in complex indoor environments. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on* (Oct 2010), pp. 1690–1695. 66

[87] Horton, T. E., Chakraborty, A., and Amant, R. S. Affordances for robots: a brief survey. *Avant : Journal of Philosophical-Interdisciplinary Vanguard 3*, 2 (2012). 122

[88] Ichbiah, D., and Adams, P. B. *Robots : Genèse d'un peuple artificiel*. Minerva, Genève, Suisse, Mar. 2005. 1

[89] Itti, L., Koch, C., and Niebur, E. A model of saliency-based visual attention for rapid scene analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 20*, 11 (Nov 1998), 1254–1259. 37, 38

[90] Jähne, B. *Digital Image Processing*, 5th ed. Springer-Verlag Berlin Heidelberg New York, 2002. 38, 45

[91] Jang, W., Kim, K., Chung, M., and Bien, Z. Concepts of augmented image space and transformed feature space for efficient visual servoing of an "eye-in-hand robot". *Robotica 9* (4 1991), 203–212. 42

[92] Kaelbling, L., Littman, M., and Moore, A. Reinforcement learning: A survey. *Journal of Artificial Intelligence Research 4* (1996), 237–285. 21, 132

[93] Kaneko, K., Kanehiro, F., Kajita, S., Hirukawa, H., Kawasaki, T., Hirata, M., Akachi, K., and Isozumi, T. Humanoid robot hrp-2. In *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on* (April 2004), vol. 2, pp. 1083–1090 Vol.2. 12

[94] Kasim, M. A. A., Aqilah, A., Jaffar, A., Low, C. Y., Jaafar, R., Bahari, M. S., and Armansyah. 896 – 901. 20

[95] Kato, Z., Pong, T.-C., and Chung-Mong Lee, J. Color image segmentation and parameter estimation in a markovian framework. *Pattern Recognition Letters 22*, 3–4 (Mar. 2001), 309–321. 45, 51, 52

[96] Khalil, W., and Dombre, E. *Modeling, Identification and Control of Robots*, 3rd ed. Taylor & Francis, Inc., Bristol, PA, USA, 2002. 100

[97] Khalil, W., and Kleinfinger, J. A new geometric notation for open and closed-loop robots. In *Robotics and Automation. Proceedings. 1986 IEEE International Conference on* (Apr 1986), vol. 3, pp. 1174–1179. 80

[98] Klatzky, R. L. Allocentric and egocentric spatial representations: Definitions, distinctions, and interconnections. In *Spatial Cognition, An Interdisciplinary Approach to Representing and Processing Spatial Knowledge* (London, UK, UK, 1998), Springer-Verlag, pp. 1–18. 24, 25

[99] Koch, C., and Ullman, S. Shifts in Selective Visual Attention: Towards the Underlying Neural Circuitry. *Human Neurobiology 4* (Jan. 1985), 219–227. 37

[100] Kolodner, J. L. An introduction to case-based reasoning. *Artif. Intell. Rev. 6*, 1 (1992), 3–34. 127

[101] Konolige, K., and Myers, K. The saphira architecture for autonomous mobile robots. In *Artificial Intelligence and Mobile Robots*, D. Kortenkamp, R. P. Bonasso, and R. Murphy, Eds. 1998, pp. 211–242. 30

[102] Kormushev, P., Nenchev, D. N., Calinon, S., and Caldwell, D. G. Upper-body kinesthetic teaching of a free-standing humanoid robot. In *Proc. IEEE Intl Conf. on Robotics and Automation (ICRA)* (Shanghai, China, 2011), pp. 3970–3975. 21

[103] Kosecka, J., and Bajcsy, R. Discrete event systems for autonomous mobile agents. *Robotics and Autonomous Systems 12* (1993), 187–198. 131

[104] Kristensen, S. Sensor planning with bayesian decision theory. In *Reasoning with Uncertainty in Robotics*, L. Dorst, M. van Lambalgen, and F. Voorbraak, Eds., vol. 1093 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 1996, pp. 353–367. 131

[105] Laumond, J., Sekhavat, S., and Lamiraux, F. Guidelines in nonholonomic motion planning for mobile robots. In *Robot Motion Planning and Control*, J.-P. Laumond, Ed., vol. 229 of *Lecture Notes in Control and Information Sciences*. Springer Berlin Heidelberg, 1998, pp. 1–53. 22

[106] Legrand, L., Bordier, C., Lalande, A., Walker, P., Brunotte, F., and Quantin, C. Magnetic resonance image segmentation and heart motion tracking with an active mesh based system. In *Computers in Cardiology, 2002* (Sept 2002), pp. 177–180. 45

[107] Lenser, S., Bruce, J., and Veloso, M. A modular hierarchical behavior-based architecture. In *RoboCup 2001: Robot Soccer World Cup V*, A. Birk, S. Coradeschi, and S. Tadokoro, Eds., vol. 2377 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2002, pp. 423–428. 126

[108] Lewis M.A., and Simo L.S. Elegant stepping: A model of visually triggered gait adaptation. *Connection Science 11*, 3-4 (1999), 331–344. 66

[109] Li, Z., and Canny, J. *Nonholonomic Motion Planning*. Springer Science & Business Media, Jan. 1993. 22

[110] Liu, H., and Iba, H. A layered control architecture for humanoid robot. In *Proceedings of International Conferences on Autonomous Robots and Agents* (Palmerston North, New Zealand, 2004), pp. 424–439. 127

[111] Low, T., and Wyeth, G. Obstacle detection using optical flow. In *Australasian Conference on Robotics and Automation 2005* (Sydney, N.S.W, 2005), C. Sammut, Ed., Australian Robotics and Automation Association Inc. 144

[112] Lozano-Pérez, T. Spatial planning: A configuration space approach. *IEEE Transactions on Computers C-32* (1983), 108–120. 22

[113] Lucas, B. D., and Kanade, T. An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence - Volume 2* (San Francisco, CA, USA, 1981), IJCAI'81, Morgan Kaufmann Publishers Inc., pp. 674–679. 47

[114] Lungarella, M., and Sporns, O. Information self-structuring: Key principle for learning and development. In *Development and Learning, 2005. Proceedings., The 4th International Conference on* (July 2005), pp. 25–30. 100

[115] MacQueen, J. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics* (Berkeley, Calif., 1967), University of California Press, pp. 281–297. 45, 48

[116] Mataric, M. J. Behavior-based control: Examples from navigation, learning, and group behavior. *Journal of Experimental and Theoretical Artificial Intelligence 9* (1997), 323–336. 27, 30, 126

[117] Matsui, D., Minato, T., MacDorman, K. F., and Ishiguro, H. Generating natural motion in an android by mapping human motion. In *IROS* (2005), IEEE, pp. 3301–3308. 20

[118] McGeer, T. Passive dynamic walking. *The International Journal of Robotics Research 9*, 2 (1990), 62–82. 18

[119] Metta, G., Sandini, G., Vernon, D., Natale, L., and Nori, F. The icub humanoid robot: An open platform for research in embodied cognition. In *Proceedings of the 8th Workshop on Performance Metrics for Intelligent Systems* (New York, NY, USA, 2008), PerMIS '08, ACM, pp. 50–56. 11

[120] Meystel, A. Knowledge based nested hierarchical control. *Advances in Automation and Robotics 2* (1990), 63–152. 27

[121] Michel, P., Chestnutt, J., Kuffner, J., and Kanade, T. Vision-guided humanoid footstep planning for dynamic environments. In *Proceedings of the IEEE-RAS Conference on Humanoid Robots (Humanoids'05)* (December 2005), pp. 13 – 18. 66

[122] Ming Luo, Yu-Fei Ma, and Hong-Jiang Zhang. A Spatial Constrained K-Means Approach to Image Segmentation. *Fourth IEEE Pacific-Rim Conference On Multimedia* (December 2003), 738 – 742. 50

[123] Minsky, M. *The Society of the Mind*. Simon and Schuster, 1985. 126

[124] Mori, M. Bukimi no tani [The uncanny valley]. *Energy 7*, 4 (1970), 33–35. 20

[125] Morse, A. F., Belpaeme, T., Cangelosi, A., and Smith, L. B. Thinking with your body: modelling spatial biases in categorization using a real humanoid robot. In *In Cognitive Science* (2010). 22

[126] Mou, W., McNamara, T. P., Rump, B., and Xiao, C. Roles of egocentric and allocentric spatial representations in locomotion and reorientation. *Journal of Experimental Psychology. Learning, Memory, and Cognition 32*, 6 (Nov. 2006), 1274–1290. 24

[127] Moughlbay, A., Cervera, E., and Martinet, P. Error regulation strategies for model based visual servoing tasks: Application to autonomous object grasping with nao robot. In *Control Automation Robotics Vision (ICARCV), 2012 12th International Conference on* (Dec 2012), pp. 1311–1316. 19

[128] Moughlbay, A., Cervera, E., and Martinet, P. Model based visual servoing tasks with an autonomous humanoid robot. In *Frontiers of Intelligent Autonomous Systems*, S. Lee, K.-J. Yoon, and J. Lee, Eds., vol. 466 of *Studies in Computational Intelligence*. Springer Berlin Heidelberg, 2013, pp. 149–162. 66

[129] Moussa, M., and Kamel, M. An experimental approach to robotic grasping using a connectionist architecture and generic grasping functions. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on 28*, 2 (May 1998), 239–253. 133

[130] Munirathinam, K., Sakka, S., and Chevallereau, C. Dynamic motion imitation of two articulated systems using nonlinear time scaling of joint trajectories. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on* (Oct 2012), pp. 3700–3705. 21

[131] Murphy, R. R. *Introduction to AI Robotics*, 1st ed. MIT Press, Cambridge, MA, USA, 2000. 23, 27, 28, 29, 126

[132] Musto, A., Stein, K., Schill, K., Eisenkolb, A., and Brauer, W. Qualitative motion representation in egocentric and allocentric frames of reference. In *Spatial Information Theory. Cognitive and Computational Foundations of Geographic Information Science*, C. Freksa and D. Mark, Eds., vol. 1661 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 1999, pp. 461–476. 24, 82

[133] Ngau, C., Ang, L.-M., and Seng, K. P. A survey of bottom-up visual saliency methods in wireless multimedia sensor networks. In *Intelligent Human-Machine Systems and Cybernetics, 2009. IHMSC '09. International Conference on* (Aug 2009), vol. 2, pp. 335–338. 47, 48

[134] Niemüller, T., Ferrein, A., and Lakemeyer, G. A lua-based behavior engine for controlling the humanoid robot nao. In *RoboCup* (2009), J. Baltes, M. G. Lagoudakis, T. Naruse, and S. S. Ghidary, Eds., vol. 5949 of *Lecture Notes in Computer Science*, Springer, pp. 240–251. 127

[135] Nilsson, N. J. A mobile automaton: An application of artificial intelligence techniques. In *Proc. of the 1st IJCAI* (Washington, DC, 1969), pp. 509–520. 22

[136] Ogino, M., Kikuchi, M., Ooga, J., Aono, M., and Asada, M. Optic flow based skill learning for a humanoid to trap, approach to, and pass a ball. In *RoboCup 2004: Robot Soccer World Cup VIII*, D. Nardi, M. Riedmiller, C. Sammut, and J. Santos-Victor, Eds., vol. 3276 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2005, pp. 323–334. 145

[137] Olshausen, B. A., Anderson, C. H., and Van Essen, D. C. A neurobiological model of visual attention and invariant pattern recognition based on dynamic routing of information. *The Journal of neuroscience : the official journal of the Society for Neuroscience 13*, 11 (Nov. 1993), 4700–4719. 36

[138] Orebäck, A., and Christensen, H. I. Evaluation of architectures for mobile robotics. *Auton. Robots 14*, 1 (2003), 33–49. 29

[139] Oriolo, G., Paolillo, A., Rosa, L., and Vendittelli, M. Vision-based odometric localization for humanoids using a kinematic ekf. In *Humanoid Robots (Humanoids), 2012 12th IEEE-RAS International Conference on* (Nov 2012), pp. 153–158. 67

[140] Otsu, N. A Threshold Selection Method from Gray-level Histograms. *IEEE Transactions on Systems, Man and Cybernetics 9*, 1 (1979), 62–66. 45

[141] Peters, J., Vijayakumar, S., and Schaal, S. Reinforcement learning for humanoid robotics. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids2003)* (Karlsruhe, Germany, Sept.29-30, 2003). clmc. 133

[142] Peters, R. A., II, Hambuchen, K. E., Kawamura, K., and Wilkes, D. M. *The Sensory Ego-Sphere as a Short-Term Memory for Humanoids*. 2001. 72

[143] Pfeifer, R., and Pitti, A. *La révolution de l'intelligence du corps*. Manuella Editions, Nov. 2012. 94

[144] Pinto, Y., Leij, A. R. v. d., Sligte, I. G., Lamme, V. A. F., and Scholte, H. S. Bottom-up and top-down attention are independent. *Journal of Vision 13*, 3 (jul 2013), 16. 34

[145] Pirjanian, P. An Overview of System Architectures for Action Selection in Mobile Robotics. Technical report, Laboratory of Image Analysis, Aalborg University, Aalborg, Denmark, 1997. 26, 27, 28

[146] Pirjanian, P. Behavior coordination mechanisms - state-of-the-art. Technical report, Institute for Robotics and Intelligent Systems, School of Engineering, University of Southern California, 1999. 130

[147] Posner, M. I., Snyder, C. R., and Davidson, B. J. Attention and the detection of signals. *Journal of Experimental Psychology 109*, 2 (June 1980), 160–174. 36

[148] Pouget, A., and Sejnowski, T. J. Simulating a lesion in a basis function model of spatial representations: Comparison with hemineglect. *Psychological Review 108* (2001), 653–673. 106

[149] Prescott, T. J. Forced moves or good tricks in design space? landmarks in the evolution of neural mechanisms for action selection. *Adaptive Behavior - Animals, Animats, Software Agents, Robots, Adaptive Systems 15*, 1 (Mar. 2007), 9–31. 26

[150] Pretto, A., Menegatti, E., Bennewitz, M., Burgard, W., and Pagello, E. A visual odometry framework robust to motion blur. In *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on* (May 2009), pp. 2250–2257. 39

[151] Proetzsch, M., Luksch, T., and Berns, K. Fault-tolerant behavior-based motion control for offroad navigation. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on* (April 2005), pp. 4697–4702. 128

[152] Proetzsch, M., Luksch, T., and Berns, K. Development of complex robotic systems using the behavior-based control architecture ib2c. *Robotics and Autonomous Systems 58*, 1 (2010), 46 – 67. 31, 128, 130, 154

[153] Quinlan, P. T., and Dyson, B. *Cognitive psychology*. Pearson Prentice Hall, New York, 2008. 34, 35

[154] Reichel, M. Transformation of shadow dextrous hand and shadow finger test unit from prototype to product for intelligent manipulation and grasping. In *Intelligent Manipulation and Grasping, International Conference, 1-2 July, Genova, Italy* (2004), University of Genova, pp. 123–124. 20

[155] Rosenblatt, J. Damn: A distributed architecture for mobile navigation - thesis summary. In *Journal of Experimental and Theoretical Artificial Intelligence* (1995), AAAI Press, pp. 339–360. 130

[156] Rother, C., Kolmogorov, V., and Blake, A. GrabCut -Interactive Foreground Extraction using Iterated Graph Cuts. *ACM Transactions on Graphics (SIGGRAPH)* (Aug. 2004). 53

[157] Ruesch, J., Lopes, M., Bernardino, A., Hornstein, J., Santos-Victor, J., and Pfeifer, R. Multimodal saliency-based bottom-up attention a framework for the humanoid robot icub. In *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on* (May 2008), pp. 962–967. 73

[158] Russell, J., and Cohn, R. *Moravec's Paradox*. Book on Demand, 2012. 3

[159] Russell, S. J., and Norvig, P. *Artificial Intelligence: A Modern Approach*, 3rd ed. Prentice Hall, 2009. 21, 133

[160] Saffiotti, A., Ruspini, E., and Konolige, K. Using fuzzy logic for mobile robot control. In *Practical Applications of Fuzzy Technologies*, H.-J. Zimmermann, Ed., vol. 6 of *Handbooks of Fuzzy Sets*. Kluwer Academic, MA, 1999, ch. 5, pp. 185–205. 130

[161] Schaal, S. Is imitation learning the route to humanoid robots? *Trends in Cognitive Sciences 3*, 6 (1999), 233–242. 21

[162] Schmidt, D., Luksch, T., Wettach, J., and Berns, K. Autonomous behavior-based exploration of office environments. In *3rd International Conference on Informatics in Control, Automation and Robotics (ICINCO)* (Setubal, Portugal, August 1-5 2006), pp. 235–240. 128

[163] Sekkati, H., and Mitiche, A. Joint optical flow estimation, segmentation, and 3d interpretation with level sets. *Computer Vision and Image Understanding 103*, 2 (2006), 89–100. 62

[164] Shapiro, L. The embodied cognition research programme. *Philosophy Compass 2*, 2 (2007), 338–346. 67

[165] Shapiro, L. *Embodied Cognition*. Routledge, New York, sep 2010. 94

[166] Siagian, C., Chang, C. K., and Itti, L. Autonomous mobile robot localization and navigation using a hierarchical map representation primarily guided by vision. *Journal of Field Robotics 31*, 3 (2014), 408–440. 37

[167] Siciliano, B., and Khatib, O., Eds. *Springer Handbook of Robotics*. Springer, 2008. 22, 38

[168] Smith, E. E., and Kosslyn, S. M. *Cognitive psychology: Mind and Brain*. 2009. 34, 36, 37

[169] Souhila, K., and Karim, A. Optical flow based robot obstacle avoidance. *International Journal of Advanced Robotic Systems 4*, 1 (2007), 13–16. 144

[170] Stellin, G., Cappiello, G., Roccella, S., Carrozza, M. C., Dario, P., Metta, G., Sandini, G., and Becchi, F. Preliminary design of an anthropomorphic dexterous hand for a 2-years-old humanoid: towards cognition. *IEEE RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics 1* (2006), 73–78. 20

[171] Stiehl, W. D., and Breazeal, C. A sensitive skin for robotic companions featuring temperature, force, and electric field sensors. In *IROS* (2006), IEEE, pp. 1952–1959. 19

[172] Stockman, G., and Shapiro, L. G. *Computer Vision*, 1st ed. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2001. 38, 44, 53

[173] Stulp, F., and Sigaud, O. Policy Improvement: Between Black-Box Optimization and Episodic Reinforcement Learning. In *Journées Francophones Planification, Décision, et Apprentissage pour la conduite de systèmes* (France, 2013). 133

[174] Testart, J., Ruiz Del Solar, J., Schulz, R., Guerrero, P., and Palma-Amestoy, R. A real-time hybrid architecture for biped humanoids with active vision mechanisms. *J. Intell. Robotics Syst. 63*, 2 (Aug. 2011), 233–255. 127

[175] Thelen, E., and Smith, L. B. *A Dynamic Systems Approach to the Development of Cognition and Action*, reprint edition ed. A Bradford Book, Cambridge, Mass., Jan. 1996. 95

[176] Theodorou, E., Buchli, J., and Schaal, S. A generalized path integral control approach to reinforcement learning. *J. Mach. Learn. Res. 11* (Dec. 2010), 3137–3181. 133

[177] Thompson, A. M. The navigation system of the jpl robot. In *IJCAI* (1977), R. Reddy, Ed., William Kaufmann, pp. 749–757. 22

[178] Thrun, S., Burgard, W., and Fox, D. *Probabilistic Robotics*. The MIT Press, Cambridge, Mass., Aug. 2005. 23, 24, 66, 100, 122, 133, 134

[179] Treisman, A. M., and Gelade, G. A feature-integration theory of attention. *Cognitive Psychology 12* (1980), 97–136. 37

[180] UN. *World Population Ageing, 1950-2050*. No. 207 in Population Studies. United Nations. Dept. of Economic and Social Affairs. Population Division, 2002. 2, 3

[181] Vasiliu, L., Trochidis, I., Bussler, C., and Koumpis, A. Robobrain: A software architecture mapping the human brain. In *Humanoid Robots (Humanoids), 2014 14th IEEE-RAS International Conference on* (Nov 2014), pp. 160–165. 102

[182] Vernon, D. Enaction as a conceptual framework for developmental cognitive robotics. *Paladyn 1* (2010), 89–98. 22, 95

[183] Vukobratović, M., and Borovac, B. Zero-moment point — thirty five years of its life. *International Journal of Humanoid Robotics 01*, 01 (2004), 157–173. 18

[184] Waibel, M., Beetz, M., Civera, J., D'Andrea, R., Elfring, J., Galvez-Lopez, D., Haussermann, K., Janssen, R., Montiel, J., Perzylo, A., Schiessle, B., Tenorth, M., Zweigle, O., and van de Molengraft, R. Roboearth. *Robotics Automation Magazine, IEEE 18*, 2 (2011), 69–82. 102

[185] Ward, J. The loneliest humanoid in america. *Popular Science* (August 2010), 38–45. 12

[186] Watkins, C. J. C. H. *Learning from Delayed Rewards*. PhD thesis, King's College, Cambridge, UK, 1989. 134

[187] Wilson, A. D., and Golonka, S. Embodied cognition is not what you think it is. *Frontiers in Psychology 4*, 58 (2013). 95, 97

[188] Wilson, M. Six views of embodied cognition. *Psychonomic Bulletin and Review 9* (2002), 625–636. 94

[189] Wolfe, J. M. Guided Search 4.0: Current Progress with a model of visual search. In *Integrated Models of Cognitive Systems*, W. Gray, Ed. Oxford, New York, 2007, pp. 99–119. 37

[190] Yoo, D.-W., Won, D.-Y., and Tahk, M.-J. Optical flow based collision avoidance of multi-rotor uavs in urban environments. *International Journal of Aeronautical and Space Sciences 3*, 3 (Sep 2011). 144

[191] Zhao, L. Cognitive modeling for computer animation: A comparative review. Technical report, Computer and Information Department, University of Pennsylvania, PA 19104-6383, USA, 2001. 27

[192] Zhu, W., and Levinson, S. Vision-based reinforcement learning for robot navigation. In *Proceedings, International Joint Conference on Neural Networks* (2001), IEEE, pp. 1025–1030. 133

Appendixes

Bayesian Network probability observation

This appendix details the calculation of Eq. (5.8), which is the query to determine the probability that the current observations O_i correspond to the object of interest. Thus, Fig. A.1 recalls the structure of the BN proposed, which is a naive Bayes classifier. That is, the information provided by the features on the branches are assumed to be independent from each other.

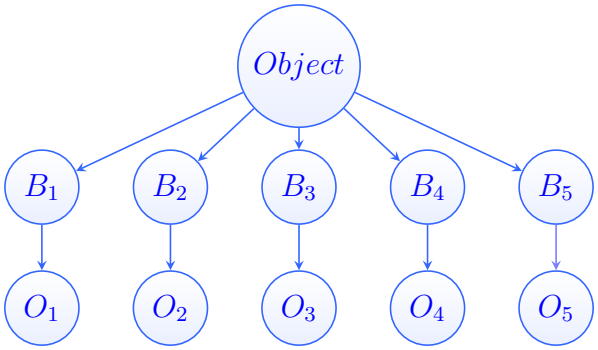


Figure A.1 – Bayesian network for contextual information fusion.

In the query the particular observation of the leaf features are propagated recursively until the root node. Given the assumption of statistical independence between the branches, only the case of the left-most branch is going to be developed here, in order to illustrate the calculations required. For this, let us assume that the following a priori knowledge is available:

1. The probability distribution $p(Object)$ of observing the object in Tab. A.1.

Object	Probability
True	0.5
False	0.5

Table A.1 – Object's a priori probability.

2. The probability distribution $p(B_1|Object)$ in Tab. A.2. It is the discriminative power of the feature B_1, given the observation of the object.

$B_1/Object$	True	False
True	0.9	0.1
False	0.1	0.9

Table A.2 – A priori knowledge on the discriminative power of feature B_1.

3. The probability distribution $p(O_1|B_1)$ in Tab. A.3. It is the discriminative power of the leaf O_1, given the observation of feature B_1.

O_1/B_1	True	False
0	0.05	0.333
1	0.15	0.333
2	0.80	0.333

Table A.3 – A priori knowledge on the discriminative power of leaf O_1.

Let an observation be described by three levels of intensity, so $O_1 \in \{0, 1, 2\}$. Assuming a value $O_1 = 2$ is observed, the query to be stated has the form

$$p(Object = \text{True}|B_1, O_1 = 2) \tag{A.1}$$

$Case$	$Object$	B_1	O_1
1	False	False	0
2	False	False	1
3	False	False	2
4	False	True	0
5	False	True	1
6	False	True	2
7	True	False	0
8	True	False	1
9	True	False	2
10	True	True	0
11	True	True	1
12	True	True	2

Table A.4 – All possible queries related to the left-most branch of the network

Before proceeding to solve the query notice that all the possible observations on the network are enumerated in Tab. A.4. Let the notation be simplified so F:False and T:True. The cases that match the clues ($Object$ = T, and $O_1 = 2$) are 9 and 12. The likelihood l_{q} of the match (which is not a probability distribution) is obtained by

$$l_{\text{q}} = p(Case = 9) + p(Case = 12). \tag{A.2}$$

By definition, in the network a node is independent from others given its parents (see Eq. (5.7)), thus

$$p(Object = \text{T}|B_1, O_1 = 2) = p(Object = \text{T})p(B_1|Object = \text{T})p(O_1 = 2|B_1). \tag{A.3}$$

Consequently,

$$p(Case = 9) = p(Object = \mathrm{T})p(B_1 = \mathrm{F}|Object = \mathrm{T})p(O_1 = 2|B_1 = \mathrm{F}), \tag{A.4}$$

and

$$p(Case = 12) = p(Object = \mathrm{T})p(B_1 = \mathrm{T}|Object = \mathrm{T})p(O_1 = 2|B_1 = \mathrm{T}). \tag{A.5}$$

By consulting Tabs. A.1, A.2, A.3, the likelihood of the query is

$$l_{\mathrm{q}} = (0.5)(0.1)(0.333) + (0.5)(0.9)(0.8) = 0.37655 \tag{A.6}$$

In order to get a full probability distribution, the likelihood of the complementary event $l_{\neg \mathrm{q}}$ must be estimated. Which is done according to the expression

$$l_{\neg \mathrm{q}} = p(Case = 3) + p(Case = 6). \tag{A.7}$$

Thus,

$$p(Case = 3) = p(Object = \mathrm{F})p(B_1 = \mathrm{F}|Object = \mathrm{F})p(O_1 = 2|B_1 = \mathrm{F}), \tag{A.8}$$

and

$$p(Case = 6) = p(Object = \mathrm{F})p(B_1 = \mathrm{T}|Object = \mathrm{F})p(O_1 = 2|B_1 = \mathrm{T}). \tag{A.9}$$

The likelihood of the complementary query is

$$l_{\neg \mathrm{q}} = (0.5)(0.9)(0.333) + (0.5)(0.1)(0.8) = 0.18985 \tag{A.10}$$

Finally, the answer to the original query is

$$p(Object = \mathrm{T}|B_1, O_1 = 2) = \frac{l_{\mathrm{q}}}{(l_{\mathrm{q}} + l_{\neg \mathrm{q}})} = 0.6647. \tag{A.11}$$

Notice that if the network is used without considering the discriminative power, it is not necessary to calculate the normalization term Z in Eq. (5.8) (which in this case is $Z = (l_{\mathrm{q}} + l_{\neg \mathrm{q}})$), but to keep the biggest likelihood (see Eq. (5.9)). Finally, it can be handy to recall at this point that the Eqs. (5.10) and (5.11) are used in case the probability distribution of Tab. A.2 is estimated in runtime.

l'université nantes angers le mans
COMMUNAUTÉ D'UNIVERSITÉS ET ÉTABLISSEMENTS

Centrale Nantes

Thèse de Doctorat

Hendry FERREIRA CHAME

Représentations Ego-centrées pour la Navigation Autonome d'un Robot Humanoïde

Egocentric Representations for Autonomous Navigation of Humanoid Robots

Résumé
Les applications en robotique de service nécessitent d'être capable de réaliser des tâches d'approches et de positionnement vis à vis d'un objet d'intérêt à partir d'informations visuelles. Les scénarios envisagés conçus pour les activités humaines (e. g. au bureau ou à la maison) sont naturellement stochastiques, il est donc important que la solution proposée permette de réagir face à l'imprévu et garantisse l'autonomie décisionnelle du robot. L'approche connectiviste traditionnelle en intelligence artificielle (IA) n'a pas réussi à produire de résultats fiables car elle est basée sur une sélection d'action centralisée introduisant un retard et des représentations non contextualisées des tâches. En revanche les modèles de comportements émergents ont permis d'obtenir des réponses rapides mais au prix d'une faible possibilité de généralisation à des scénarios proches. Notre travail s'est appuyé sur un point de vue intermédiaire entre la méthodologie connectiviste et la cognition incarnée (Embodied Cognition (EC)). On adopte simultanément des représentation indépendantes-de-l'action pour faire la reconnaissance visuelle de la cible et des représentations locales sous la forme de sensations corporelles afin d'anticiper les conséquences de l'action, de discriminer les objets, de réagir à des circonstances imprévues, et d'évaluer le progrès et le succès de la mission.

Mots clés
Architectures des comportements, Attention visuelle ascendante et descendante, Vision par ordinateur, Localisation égocentrique embarquée, Sélection d'action, Intégration multisensorielle, Apprentissage par renforcement.

Abstract
The skill of visually approaching and positioning in relation to objects on the scene is of crucial importance for service robotics applications. Furthermore, the autonomy of the solution is essential, since human-centered scenarios, where these robots are expected to operate (e.g. at the office of home), are stochastic; so is important that the agent can react to unforseen situations. The traditional approach of AI has not produced reliable results since they are based on extensive context-free models of the tasks, so action selection is a centralized and delayed process. Emergent models have in contrast produced fast-response systems at the cost of poor generalization power even to very similar scenarios. This research has taken an intermediate perspective between the cognitivist and the EC research. It employs simultaneously action-independent knowledge for visually recognizing the stimuli of interest, and local representations in the form of bodily sensations, in order to anticipate the consequences of action, to discriminate the object, to react to unexpected circumstances and to assess the progress and success on the mission.

Key Words
Behavior architectures, Bottom-up and top-down visual attention, Machine vision, Egocentric on-board localization, Action-selection, Multisensory-integration, and Reinforcement learning.