Developmental approach of perception for a humanoid robot

Natalia Lyubova

THÈSE

PRÉSENTÉE À

ENSTA ParisTech

ÉCOLE DOCTORALE DE L'ÉCOLE POLYTECHNIQUE (EDX)

Par **Natalia Lyubova**

POUR OBTENIR LE GRADE DE

DOCTEUR

SPÉCIALITÉ : INFORMATIQUE

Developmental approach of perception for a humanoid robot

Soutenue le : 30 Octobre 2013

Devant la commission d'examen composée de :

Giorgio Metta	Professeur - IIT	Rapporteur
Peter Ford Dominey ..	Directeur de Recherche - CNRS	Rapporteur
Ryad Benosman	Maitre de Conférence HDR - Institut de la Vision	Examinateur
Jean Christophe Baillie	Directeur de Recherche - Aldebaran Robotics ...	Examinateur
David Filliat	Professeur - ENSTA ParisTech	Directeur de Thèse

– 2013 –

Abstract

Future service robots will need the ability to work in unpredicted human environments. These robots should be able to learn autonomously without constant supervision in order to adapt to the environment, different users, and changing circumstances. Exploration of unstructured environments requires continuous detection of new objects and learning about them, ideally like a child, through curiosity-driven interactive exploration.

Our research work is aimed to design a developmental approach that enables a humanoid robot to perceive its close environment. We take inspiration from human perception in terms of its functionalities and from infant development in terms of the way of learning, and we propose an approach that enables a humanoid robot to explore its environment progressively, like a child through physical actions and social interaction. Following principles of developmental robotics, we focus on incremental, continuous, and autonomous learning that does not require a prior knowledge about the environment or the robot.

The perceptual system starts from segmentation of the visual space into proto-objects as units of attention. The appearance of each proto-object is characterized by low-level features based on color and texture that are considered as complementary features. These low-level features are integrated into more complex features and then, into a multi-view model that is learned incrementally and associated with one physical entity. Entities are then classified into three categories : parts of the robot's body, human parts, and manipulable objects. The categorization approach is based on mutual information between the sensory data and proprioception, and also on motion behavior of physical entities. Once the robot is able to categorize entities, it focuses on interactive object exploration. During interaction, the information acquired about an object's appearance is integrated into its model. Thus, interactive learning enhances the knowledge about objects appearances and improves the informativeness of objects models. The implemented active perceptual system is evaluated on an iCub humanoid robot, learning 20 objects through interaction with a human partner and the robot's own actions. Our system is able to recognize objects with 88.5% success and to create coherent representation models that are further improved by interactive learning.

Résumé

Les robots de service ou d'assistance doivent évoluer dans un environnent humain en constant changement, souvent imprévisible. Ils doivent donc être capables de s'adapter à ces changements, idéalement de manière autonome, afin de ne pas dépendre de la présence constante d'une supervision. Une telle adaptation en environnements non structurés nécessite notamment une détection et un apprentissage continu des nouveaux objets présents, que l'on peut imaginer inspirés des enfants, basés sur l'interaction avec leur parents et la manipulation motivée par la curiosité.

Notre travail vise donc à concevoir une approche développementale permettant à un robot humanoïde de percevoir son environnement. Nous nous inspirons à la fois de la perception humaine en termes de fonctionnalités et du développements cognitifs observé chez les infants. Nous proposons une approche qui permet à un robot humanoïde d'explorer son environnement de manière progressive, comme un enfant, grâce à des interactions physiques et sociales. Suivant les principes de la robotique développementale, nous nous concentrons sur l'apprentissage progressif, continu et autonome qui ne nécessite pas de connaissances a priori des objets.

Notre système de perception débute par la segmentation de l'espace visuel en proto-objets, qui serviront d'unités d'attention. Chaque proto-objet est représenté par des caractéristiques bas-niveaux (la couleur et la texture) et sont eux-mêmes intégrés au sein de caractéristiques de plus haut niveau pour ensuite former un modèle multi-vues. Cet apprentissage s'effectue de manière incrémentale et chaque proto-objet est associé à une ou plusieurs entités physiques distinctes. Les entités physiques sont ensuite classés en trois catégories : parties du robot, parties des humains et objets. La caractérisation est basée sur l'analyse de mouvements des entités physiques provenant de la vision ainsi que sur l'information mutuelle entre la vison et proprioception. Une fois que le robot est capable de catégoriser les entités, il se concentre sur l'interaction active avec les objets permettant ainsi d'acquérir de nouvelles informations sur leur apparence qui sont intégrés dans leurs modèles de représentation. Ainsi, l'interaction améliore les connaissances sur les objets et augmente la quantité

d'information dans leurs modèles.

Notre système de perception actif est évalué avec le robot humanoïde iCub en utilisant une base expérimentale de 20 objets. Le robot apprend par interaction avec un partenaire humain ainsi que par ses propres actions sur les objets. Notre système est capable de créer de manière non supervisée des modèles cohérents des différentes entités et d'améliorer les modèles des objets par apprentissage interactif et au final de reconnaître des objets avec 88.5% de réussite.

Acknowledgments

First and foremost, I would like to express my gratitude to David Filliat for providing me an opportunity to join his teams at ENSTA during these years. I am really glad to work together with him. I very much appreciate his great ideas, and I thank him for giving me an inspiration and providing support.

I gratefully acknowledge the funding sources of the French ANR program that made possible my Ph.D. work as a part of the MACSi project. I thank all the project's partners (INRIA Flowers team, ISIR UPMC, and GOSTAI) for the collaborative work.

I would like to thank all my colleagues for their support. I am really glad to work with them and I am grateful for the interesting discussions, advices, exchange of ideas, and the time spent together. I thank the people helping me with this work and with this thesis. I am especially grateful for Serena Ivaldi for working together with the iCub robot and for all experiments made and published together. I thank all people working on the same project, especially Damien Gérardeaux for his great job and his contribution to the project.

Also, I would like to thank my family for their support, help, encouragement, patience, and understanding. For my parents who raised me with a love of science and supported me in all my pursuits. Also I thank my friends all over the world, being close or far away they fulfill me with the strength to overcome any possible difficulty in my research and in my life. I thank Sebastian Knoedel for his support during my first years of living in Paris and for this beautiful latex template. And I thank Victoria Rudakova for her responsiveness and helpfulness in any situation.

Table des matières

CHAPITRE 1

Introduction

Nowadays, robots are coming into everyday life, not only as factory robots but also as service robots helping people to increase the performance of their work and improve the quality of life. There is no consensus on the definition of a service robot, although they are considered to be semi- or fully autonomous and to perform services useful to well-being of humans. Service robots assist humans, doing a job that is difficult, dull, or repetitive. We imagine these robots working at homes, hospitals, hotels or other service sectors. From the robots working in hospitals and helping elderly or ill people, we expect to fulfill needs of people not just as a mechanical device but also as an attentive and mindful friend. From domestic robots, we expect to perform household chores, like cleaning, cooking or entertainment (Fig.1.1) and it would be advantageous, if these personal robots could adapt to different environments, to different users, and their needs.

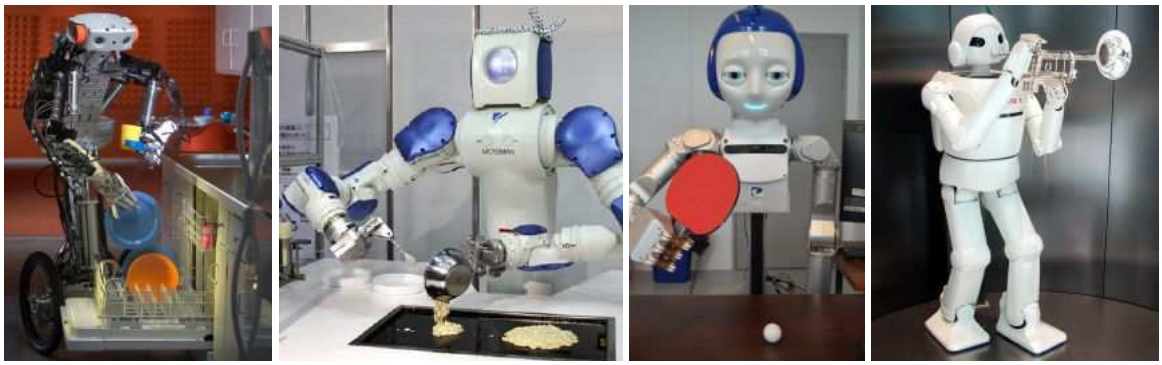


FIGURE 1.1 – Examples of personal robots : Anybots, Motoman, Meka, and Toyota Kaikan

If factory robots work in well-structured environments and perform precise tasks predefined by sequences of processing steps, there is a big difference in the capabilities required from service robots working in human environments that are rarely structured and mostly unpredictable. The understanding of a human environment requires a certain level of intelligence in order to accomplish various non-trivial tasks. We expect a service robot to learn efficiently about its environment in order to understand its meaningful items, main events,

and their actors. A service robot should not depend on constant human supervision, but rather learn autonomously by continuously acquiring new data and synthesizing them with a prior experience into coherent high-level knowledge based on inferences and own decisions. Moreover, it is advantageous if a robot can adapt not only to its environments but also to different users, their desires, behaviors, and characters ; this capabilities would allow robots to turn from purely mechanical devices into intelligent friends of humans.

We consider, that human development is the best example of learning about the environment. Starting from acquisition of basic capabilities, infants progressively learn complex skills and demonstrate understanding of the surrounding world and first intelligence at young ages. Infants learn about the surrounding world through physical contact with the environment and social communication. Infants' experimental behavior, invention and enthusiasm allow them to acquire knowledge that can be unknown to their teachers. The originality of infant development has inspired a variety of research studies on autonomous robots learning. The characteristics of infants learning process, such as being continuous, incremental, and using multimodal exploration of the world, are reflected in different approaches investigating developmental mechanisms, architectures, and constraints. In contrast to traditional robotics, developmental approaches does not focus on a fast achievement of predefined goals, but rather on open-ended learning process, where the performance improves over time, the learning process is flexible and allows the robot to adapt to changing circumstances. From our point of view, taking inspiration from infant perceptual development, the way of acquisition of knowledge and skills, as well as following principles of developmental robotics, is the most appropriate way to learn about the environment.

Perception and learning object

Personal robots working in a real-world human environment should be able to perceive the space in order to identify meaningful elements of the surrounding environment and its actors, like objects and humans. Real-world environments are often unstructured and unpredictable, so, they require the visual segmentation into meaningful entities that can be recognized or learned over time. In order to perceive the space efficiently, like humans do while perceiving the world as organized 3D structures, vision and depth are efficient perceptual modalities. A fast and easy acquisition of 3D data can be achieved using a RGB-D sensor, that moreover, provides data with a high precision compared to stereo vision.

Numerical processing of visual data is investigated in the computer vision field aimed at reproducing human vision capabilities in understanding images. Traditional computer vision approaches achieve reasonable performances for detecting specific objects of particular classes, like human faces, skin, simple colored or textured objects. Most of these approaches are based on a prior knowledge including algorithm choices or labeled samples, which correspond to a visual input with its interpretation. Traditional computer vision approaches usu-

ally require carefully created image databases, where each object is associated with images or extracted visual characteristics and their labels. Other studies include an object learning phase, for example, a turntable is used to rotate an object and to learn its appearance from different viewing angles. Prior knowledge and supervision facilitate object detection, but they are not easily applicable for autonomous robots working in unstructured real-world environments. Traditional computer vision approaches limit robots' adaptabilities, since it is difficult to extend these approaches for non specialists or new objects. However, continuous detection and learning new objects without constant human supervision are crucial capabilities for service robots. These robots should recognize objects independently of categories, properties, or viewing conditions. Therefore, object recognition should be based on general high-level representations and learning methods that are adaptable to the environment.

Approach

We work on a perceptual approach that fulfills the nowadays requirements in the robotics domain and allows a robot to learn about its environment continuously and autonomously. We design our perceptual system taking inspiration from infant development and following principles of developmental robotics.

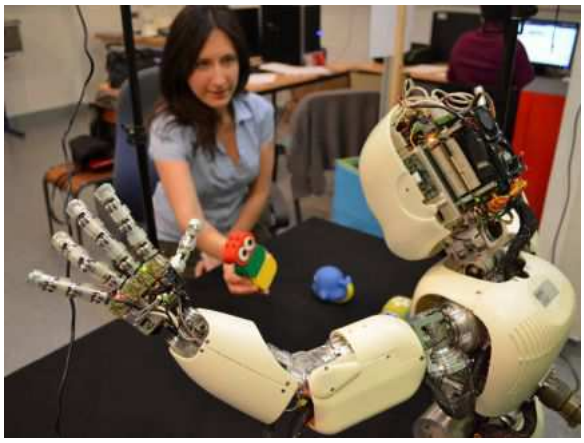
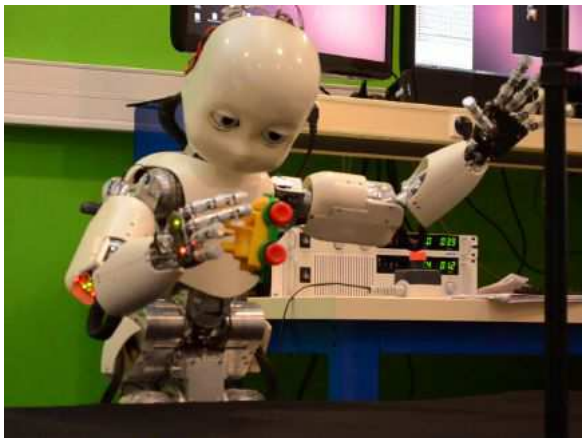
Our approach requires limited prior knowledge, it needs neither a predefined environment, nor predefined objects. We use none of image databases and none of specified detectors, such as markers or human face/skin/skeleton detectors. In our approach, the robot incrementally explores its close environment through autonomous segmentation of the visual space into physical entities and learning them by continuously processing the visual data and synthesizing the gathered information into high-level representations of physical entities.

All information about the entities appearances and behavior is acquired in interactive scenarios within the following contexts (Fig.1.2) :

- learning through observation, while a human partner interacts with the robot and demonstrates different objects,
- learning through interaction, while the robot is free to move its hands, torso, and head, to perform object-oriented interactive actions, and to explore objects manually.

The perception of the environment starts with the robot's attention based on saliency of the visual space, like in human vision. In our approach, the robot's attention is attracted mostly by motion. The visual space is segmented into proto-objects as units of attention defined from coherent motion, appearance, and continuous 3D shape. The proto-object concept is inspired from human vision, where a proto-object is considered as a unit of visual information with closely localized features acquired during the pre-attentive stage and integrated into a coherent object notion during the focused attention.

In our algorithm, the appearance of each proto-object is incrementally analyzed in order

Learning through observation Learning through interaction

FIGURE 1.2 – Main contexts of our experiments

to learn or recognize it as a physical entity. The proto-object's appearance is characterized by complementary low-level features encoding color and texture. The low-level features are grouped into mid-features integrating local geometry. The mid-features are used to encode a view that characterizes an entity's appearance from one perspective. The overall appearance of the entity is represented as a multi-view model characterizing its different perspectives. The entity's representation model is learned while a corresponding object is manipulated and tracked in the visual space. The identification of an entity is based either on tracking across images, or on appearance-based recognition with a Bayesian filter.

Since our work is based on interaction with objects, objects often move together with a human or a robot hand, and a grasped object composes a single proto-object with the hand. In this case, in order to achieve robust recognition during manipulation, our perceptual system recognizes each proto-object either as a single entity, or two connected entities.

Each physical entity is then classified into one of the following categories : a part of the robot's body, a human part or a manipulable object. The categorization approach is independent on the robot's appearance. The robot self-identification is achieved during motors activity, while both sensory and proprioceptive data are acquired and used for estimation of mutual information. The entities with high mutual information are identified as the robot's parts. Among the remaining entities, object are considered to be mostly static and independent on the robot's motors ; the object category is identified based on the statistics on entities motion simultaneously with human and robot's parts.

Once the robot is able to identify its own body and to categorize other entities, it focuses on interactive exploration of objects. The robot performs simple interactive actions and complex manipulations aimed to acquire maximum information about an object's appearance. The ability to categorize entities is used to distinguish between a view corresponding to the object and a view corresponding to the hand manipulating it. Thus, during the interaction

with the object, its representation model is updated with recognized non-robot views and newly created views.

Overview of Contributions

Our main contribution is the integration of perception, self- and others- identification, and interactive exploration of the environment into an active perceptual system. The following robot's capabilities have been implemented :

- a perceptual system that enables to detect physical entities based on visual attention and to learn entities appearances incrementally by organizing the gathered knowledge into multi-view representation models,
- active categorization that classifies all detected entities into parts of the robot's body, parts of human partners, and manipulable objects,
- active object exploration accomplished through interactive actions and manipulations aimed to acquire maximum information about an overall appearance of the object and to improve its multi-view representation model acquired through observation.

The important aspect of our approach is incremental, continuous, and autonomous learning by following principles of developmental robotics, without a need of a prior knowledge about the environment, its objects, or the robot. The implemented system is inspired by human perception and development, and allows the robot to learn about its close environment, like a child, through exploratory actions and social interaction.

Publications

The work presented in this document was published in 1 journal paper, 5 conference papers/posters, and 3 workshops.

Journal

– Object learning through active exploration, Ivaldi, S., Nguyen, S.M, Lyubova, N., Droniou, A., Padois, V., Filliat, D., Oudeyer, P-Y., Sigaud, O., In IEEE Transactions on Autonomous Mental Development, 2013

Conferences

– Improving object learning through manipulation and robot self-identification, Lyubova, N., Filliat, D., and Ivaldi, S., In IEEE Proc. of the International Conference on Robotics and Biomimetics (ROBIO), 2013

– Learning to recognize objects through curiosity-driven manipulation with the iCub humanoid robot, Nguyen, S.M, Ivaldi, S., Lyubova, N., Droniou, A., Gérardeaux-Viret, D., Filliat, D., Padois, V., Sigaud, O., and Oudeyer, P-Y., In Proc. of the International Conference on Development and Learning (ICDL), 2013

– Perception and human interaction for developmental learning of objects and affordances, Ivaldi, S., Lyubova, N., Gérardeaux-Viret, D., Droniou, A., Anzalone, S., Chetouani, M., Filliat, D., and Sigaud, O., In Proc. of the IEEE-RAS International Conference on Humanoid Robots (HUMANOIDS), 2012

– Developmental approach for interactive object discovery, Lyubova, N. and Filliat, D., In IEEE Proc. of the International Joint Conference on Neural Networks (IJCNN), 2012, pp. 1-7

– Interactive object learning using developmental approach, Lyubova, N. and Filliat, D., Poster for the International Conference on Cognitive Systems (CogSys), 2012

Workshops

– Developmental object learning through manipulation and human demonstration, Lyubova, N., Ivaldi, S., Filliat, D., In Interactive Perception Workshop - IEEE International Conference on Robotics and Automation (ICRA), 2013

– A cognitive architecture for developmental learning of objects and affordances : perception and human interaction aspects, Ivaldi, S., Lyubova, N., Gérardeaux-Viret, D., Droniou, A., Anzalone, S. M., Chetouani, M., Filliat, D., and Sigaud, O., Poster for the IEEE Ro-man Workshop on Developmental and bio-inspired approaches for social cognitive robotics, 2012

– Developmental Learning for Object Perception, Lyubova, N. and Filliat, D., In Workshop on Deep Hierarchies in Vision, 2012

Thesis Organization

This thesis presents the development of a perceptual system for a humanoid robot learning about its close environment. Looking for the appropriate way to learn continuously and autonomously, we take inspiration from infant development and follow principles of developmental robotics. In Chapter 2, we give an overview of functionalities of the human perceptual system, and we describe the main stages of its development. We are interested in infant perceptual development, the way of learning about the world, and its reflection in developmental robotics. We give a short description of general concepts of developmental robotics and its application to learning objects.

Our learning approach for a robot exploring its environment is presented in the two parts of this thesis : learning through observation and learning through interaction. Each part has a similar structure with an overview of the state of art, the description of the proposed approach and its implementation, and the experimental evaluation.

Part I of this thesis is devoted to learning about the robot's environment based on pure observation. In Chapter 3, we review existing computer vision algorithms aimed at detecting and learning objects. In Chapter 4, we present the proposed perceptual system and its implementation including detection and segmentation of proto-objects, the choice of visual features, the entity representation model, and the learning method. The experimental evaluation of the implemented system is reported in Chapter 5, where we describe the organization of experiments, the methodology of evaluation, and the achieved results.

Part II covers the active exploration of the environment, when the robot learns through its interactive actions and observation of their effects. In Chapter 6, we review existing approaches on interactive perception focusing on the robot's self-identification and the use of interaction for learning objects. In Chapter 7, we present the proposed active perceptual system and its implementation. We describe the categorization of physical entities into parts of the robot's body, parts of a human partner, and manipulable objects, and then the integration of categorization together with the perceptual approach proposed in Part I into interactive object exploration method. The experimental evaluation of the implemented system is reported in Chapter 8, where we report the organization of experiments, the evaluation of categorization and interactive object learning. Finally, the results obtained during interactive learning are compared to the results obtained during learning through observation.

Last Chapter 9 is devoted to conclusions and discussions about the importance of our work, our achievements, and limitations of our approach that can be improved in future work.

CHAPITRE 2

Developmental approach

Humans provide a very good source of inspiration for the development of a system capable of continuous and efficient learning, since they learn progressively, and already from an early age in life demonstrate an understanding of the surrounding environment, occurring events, as well as involved actors.

Human vision displays a fast understanding of the scene and very strong ability to recognize and differentiate objects. However, vision and signal processing performed by the human brain is quite complex, so we are not going to study and model them precisely, but rather take inspiration from their main functionalities. In Section 2.1 we give an overview of visual attention, perception of features, and their integration into high-level semantic representations, such as proto-objects which are later recognized as objects. We focus on these functionalities, since they are the ones we are going to adapt into our perceptual approach designed for a humanoid robot.

In order to understand how the notion of an object appears within perception, in Section 2.2 we study the different stages of infant perceptual development. We describe the acquisition of abilities used to perceive objects' properties and to segregate objects in a scene. We also discuss the infant's exploration of own body and the exploration of the surrounding world, including its objects, through physical and social interaction. Since infants development is used as an inspiration for continuous and incremental learning within the developmental robotics field, in Section 2.3 we describe general principles of developmental approaches and their application to object learning. The overall architecture of our system and its relation to human vision, infants development and developmental robotics are described in Section 2.4.

2.1 Human perception

Human eyes are well tuned sensor elements that allow one to perceive the surrounded world, understand the environment, gain knowledge, and to communicate thoughts

through the visual expression [Paternoster, 2007]. The human vision system is a complex mechanism with a complex structure and multiple layers of neural cells that processes the acquired light and leads to identification of elements of the environment building up perception of the surrounding world. Human perception results in an image that humans have in mind, but this image is far from actual representation of the observed environment [Goldstein, 2010]. Due to the limited resolution of eye receptors, most of seeing information is obtained not directly from the eyes, but through the processing of that acquired data along the visual pathways among several cortex areas shown in Fig.2.1 [Paternoster, 2007]. We give a brief description of the low-level processing that happens in human eyes, and focus on the higher-level processing that is important for perception of the environment.

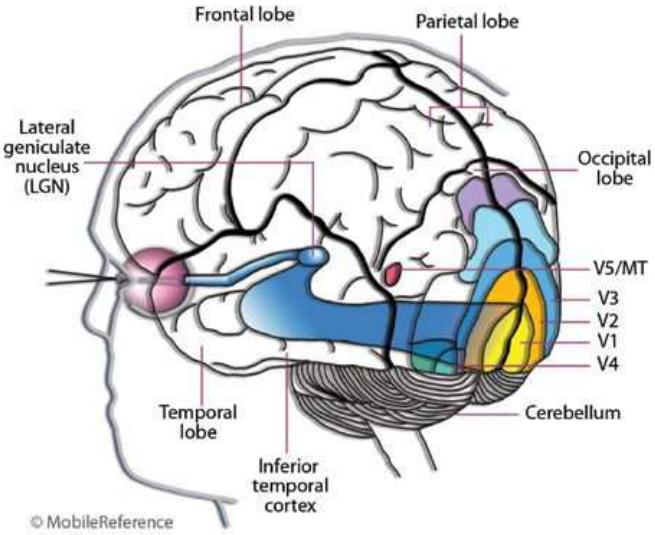


FIGURE 2.1 – The areas of a human brain that participate in perception [Vyshedskiy, 2009]

2.1.1 Signal processing

The physical nature of perception includes three main components interacting between each other : a light source, a surface, and a visual sensor. The light from the source reflects off the surface and incidents upon the visual sensor ; the light reflected off the surface and acquired by the sensor integrates properties of both the surface and the light source [Goldstein, 2010]. In human vision, the eyes play a role of a visual sensor, while in robot vision, the visual sensor is an integrated or external camera.

The light entering a human eye is processed by the sensitive neurons called photoreceptors [Goldstein, 2010]. The photoreceptors that support daytime vision are responsible for perception of luminance, contrast, and color, and they come in three spectral classes :

– cones mostly sensitive to nearly "red" colors,

– cones mostly sensitive to nearly "green" colors,
– cones mostly sensitive to nearly "blue-violet" colors.

The information captured by photoreceptors is processed through multiple layers of neuron cells. Most of these cells withdraw signals from several receptors and generalize them into a single output. They can have an excitatory response to some inputs and an inhibitory response to others. For example, color opponent cells analyze the signals from two types of photoreceptors and have excitatory response to wavelengths in one part of the spectrum and an inhibitory response in other part of the spectrum (like red-green opponent cells) [Goldstein, 2010]. The functionalities of these neuron cells processing opponent colors, as well as cells processing oriented gradients are integrated in computer vision algorithms aimed at detecting objects in biologically motivated architectures, like [Siagian and Itti, 2007], [Orabona et al., 2005] and [Walther and Koch, 2006] described in Section 4.1. In the primary visual cortex, among neurons that respond to oriented edges, there are simple cells that are sensitive to different frequencies and orientations, and complex cells that have a degree of spatial invariance [Hérault, 2010]. It is interesting to note that both simple and complex cell behavior can be learned by Deep Learning approaches that are able to capture invariance and discover non-local structure in data distribution, while learning hierarchies of visual features in computer vision algorithms [Lee et al., 2009].

A great part of information processing leading to perception of objects is accomplished by the primary visual cortex (V1) that transfers information to two primary visual pathways [Goldstein, 2010] :

– the dorsal stream called "Where Pathway", goes through V2 and the middle temporal area (MT) and is associated with motion and object locations, responsible for sensorimotor processing,
– the ventral stream called "What Pathway", goes through V2 and V4) and is associated with a shape recognition, object representation and long-term memory responsible for cognitive processing.

2.1.2 Visual attention

The most photoreceptors that are sensitive to the daylight are localized in a tiny part of a central retina called fovea [Goldstein, 2010]. Thus, during perception of a scene, only a small area of the scene around the focus can be perceived with a high resolution. The perception of a whole scene is possible due to eye movements called saccades [Hérault, 2010]. Saccades shift the gaze that allows to trace a scene and acquire its visual details. The selective visual attention directs a human gaze towards regions of interest in the visual field. There are two different aspects of human attention [Goldstein, 2010] :

– bottom-up processing also known as a stimulus-driven attention,
– top-down processing also known as a goal-driven attention.

2.1.2.1 Bottom-up visual attention

During bottom-up processing, attention is driven by characteristics of the scene, like color, contrast, orientation, etc. Among other factors, motion also attracts human attention in a pre-conscious way [Goldstein, 2010]. The visual property that turns some areas of a scene to stand out from their neighborhood and to attract attention, is called visual saliency [Itti and Koch, 2001]. Visual saliency and bottom-up attention mechanism inspire modeling of attention in computer vision (for example, saliency concept [Itti and Koch, 2001]) that are further used for object detection, like in [Siagian and Itti, 2007] and [Walther and Koch, 2006].

2.1.2.2 Gist of a scene and top-down attention

Selective visual attention, as well as a perception of "gist", provide people with an ability to perceive the surrounding environment efficiently, even in case of limited cognitive resources [Itti and Koch, 2001], [Fei-Fei et al., 2007]. The term gist is defined as the amount of information about the scene gathered by a human within a single glance that lasts about 200 ms [Oliva and Torralba, 2006]. The captured gist provides a brief understanding of a scene and its abstract meaning ; for example, a type of a scene [Rensink, 2000]. Fast gist perception is possible due to rapid acquisition of global image features that can be associated with scenes types. Among global image features, the degree of naturalness, roughness, expansion, and color are reported in [Oliva and Torralba, 2006].

Selective visual attention is influenced by the top-down attention mechanism affected by the observer's task, expectations, knowledge about the scene, past experience, etc. [Goldstein, 2010]. Top-down attention is important for high-level behavior, but we will not use it in our system.

2.1.3 Interpretation of sensory information into objects and proto-objects as high-level semantic representations

The efficient perception of the environment is possible due to the organization of the perceptual process, when at first, humans perceive an overall gist with a general description of a scene, and then the selective visual attention guides the gaze to visual details [Goldstein, 2010]. All information acquired by processing a sensory input by eyes and the visual cortex is integrated into a coherent object identity over the following stages [Treisman and Gormican, 1988] :

– during the pre-attentive stage, low-level processing including perception of general visual features, like color and orientation, is performed in parallel across the visual field,

– during focused attention, humans perceive details of the scene, and isolated visual features are combined into a coherent object identity.

According to the theory of feature integration shown in Fig.2.2, the attention links together visual features acquired at a close location into an integrated meaning of a whole that combines both "what" and "where" streams of the visual cortex.

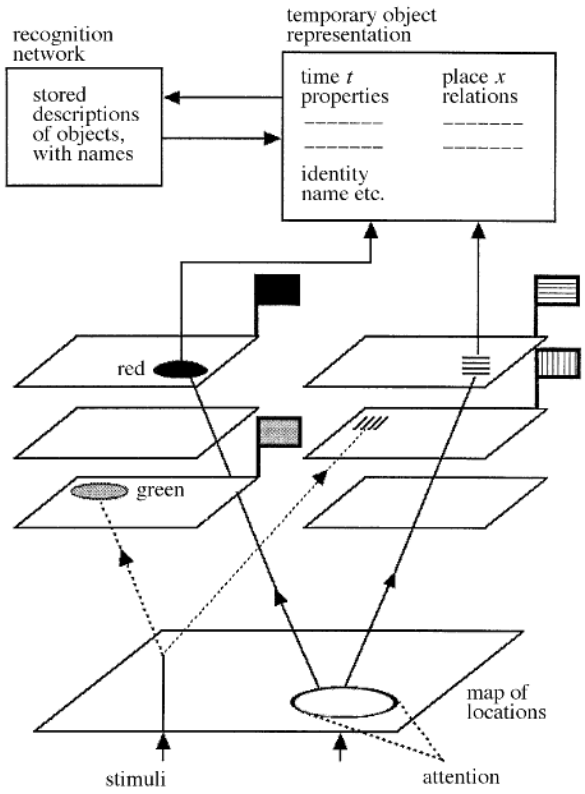


FIGURE 2.2 – A theory of integrating features [Treisman and Gormican, 1988]

According to [Rensink, 2000], among the visual features acquired during the preattentive stage, features at a close location build up a local description of a scene that determines a proto-object. The term proto-object is defined as "a volatile unit of visual information that can be bound into a coherent and stable object when accessed by focused attention". A proto-object contains more information than a single visual feature, but it is not yet a physical object. As an example, a proto-object can be a visual area with certain characteristics at one location in the visual space, but this area is not yet recognized as an object.

The temporal coherence of proto-objects is limited, since they are constantly regenerated, when new stimuli appear at the same location on a human eye retina. During focused attention, the information about all proto-objects from the visual scene is analyzed and used for high-level decisions about coherent identities of possible objects, as shown in Fig.2.3. After releasing the focused attention, the information processing is performed again on the level of proto-objects [Rensink, 2000]. The proto-object concept is used in some com-

puter vision studies aimed at detecting and learning objects, like [Walther and Koch, 2006], [Orabona et al., 2005], [Zhou et al., 2011], and [Natale et al., 2005].

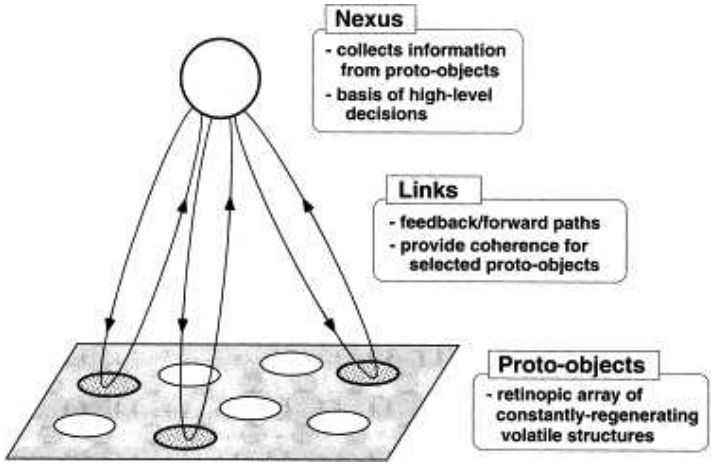


FIGURE 2.3 – The coherence field including proto-objects and links between them and a nexus collecting low-level information for higher-level decisions [Rensink, 2000]

The human perceptual process is already complex, but it is just a part of cognition, with its great ability to learn incrementally and to use the acquired experience continuously throughout life [Mohan et al., 2013]. The acquisition of cognitive skills over different stages of infant development is described in the following section.

2.2 Infants perceptual development

As the most appropriate way of learning about the environment, we consider human development. Humans show an impressive ability of cumulative lifelong and open-ended learning. From birth, infants continuously develop over time and experience by acquiring various skills, gathering knowledge, and progressively improving them [Haith, 1968]. At early age of life, infants begin to show first signs of their intelligence, the first understanding of the surrounding environment, its own body, and its impact to the physical world [Smith and Gasser, 2005]. The stages of the development of infant intelligence, incremental acquisition, use, and synthesis of knowledge are analyzed in the Piaget's theory of cognitive development [Piaget, 1999]. We focus on specific infant capabilities and give a brief overview of infant perceptual development, including the perception of objects and their properties, sensorimotor development with exploration of own body and exploration of the world through physical actions and social interaction (Fig.2.4).

a) b) c) d)

FIGURE 2.4 – Infant development : a) perception of objects, b) own body control, c) learning about objects through physical actions, d) learning through social interaction

2.2.1 Learning about visual properties

Prior to individuation of objects, infants acquire basic skills of perceiving object properties, such as color, shape, size, motion, etc. The perception of these properties and the ability to make inferences about changes in these properties are needed to understand a general and more complex concept of an object's identity [Cohen and Cashon, 2003].

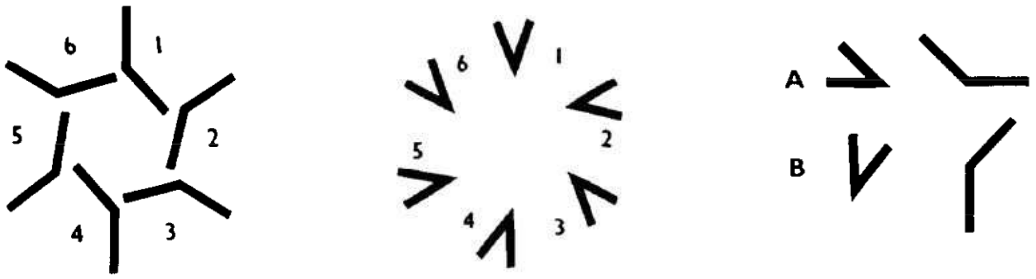
Some perceptual skills infants have from their birth ; the acquisition of other skills is investigated by various experimental studies based on demonstration of a specific stimulus and analyzing the following reaction of children. The ability of an infant to learn and recognize a stimulus is often evaluated based on the habituation paradigm, like it is done in the research studies [Bornstein et al., 1976], [Slater et al., 1991], and [Oakes and Baumgartner, 2012]. The recognition of a stimulus is determined based on the duration of an infant's gazing, and the inference about a novel stimulus is made based on dishabituation, when a novel stimulus is gazed significantly longer than a known stimulus.

Shape understanding is studied in terms of infants' reaction to the different spatial organization of object features. The early evidences of the shape constancy and size constancy are found within the perception of newborns. If the dishabituation to different line orientations (see Fig.2.5) is found among newborns, the dishabituation to different angle amplitudes is found among infants at the age of two months [Slater et al., 1991].

At the same age about two-three months, infants show the ability to discriminate between some colors produced by different wavelengths [Kellman and Arterberry, 2000]. At the age of four months infants show the ability to organize different colors into color categories that are similar to color categories perceived by adults [Bornstein et al., 1976].

2.2.2 Learning about objects

The visual saliency plays an important role in infants' understanding of objects. As one factor of saliency, motion attracts infants attention from the age of two months [Volkmann and Dobson, 1976], and it helps to understand the object unity [Johnson and Náñez Sr, 1995]. Under the object unity, we consider the perception of



a) b) c)

FIGURE 2.5 – Form perception at birth : a)first familiarization trial with six stimuli with the same angle but different orientations, b)second familiarization trial with six stimuli with the same angle but different orientations, c)two test trials, where *A* and *B* are pairs of stimuli of same orientation but different angles [Slater et al., 1991]

object parts as integrated into a single whole. Infants' understanding of object unity is based on the motion behavior of an object's parts, and the simultaneously moving parts are perceived as an integral object [Kellman and Arterberry, 2000].

Object individuation is the process of determining how many distinct objects are involved in a given scene or an event [Van de Walle et al., 2000]. Studies on object individuation usually concern the infant's ability to segregate objects as isolated units and to discriminate different objects. Under the object segregation, we consider detection of boundaries between adjacent objects. There are several opinions about the way in which infants acquire the ability to segregate objects. From one point of view, this ability is trained up during an infant's development. Early experiments with infants at the age of three months show, that they interpret all touching surfaces as a single object [Kestenbaum et al., 1987] ; in the following development, infants show to parse surfaces into units based on boundaries and visual characteristics of surfaces [Needham and Baillargeon, 1998]. There is another point of view, which asserts that the ability to segregate objects comes from a prior experience, like an exposure of an infant to so-called "key events", including perception of isolated objects or relative object motion. In experiments performed with infants at the age of 4.5 months, a prior experience takes a form of a test display showing one object as isolated, and the use of this prior experience for the object segregation task is analyzed based on the duration of the infant's gazing. The experiments have shown, that motion of several objects as a single unit was unexpected for infants, and that separate motion of objects was expected. This results in the inference that infants use a prior experience for object segregation [Baillargeon, 1999].

2.2.3 Sensorimotor development

Infant sensorimotor development starts from birth and takes about two years based on the Piaget's theory of cognitive development [Piaget, 1999]. The sensorimotor stage can be

divided into the following sub-stages characterizing the development of different skills :

- reflexes (0-1 month), when the understanding of the environment is limited by inborn reflexes,
- primary circular reactions (1-4 months) including continuously repeated actions produced by accident,
- secondary circular reactions (4-8 months) including repeated object-oriented actions that are aimed to initiate a response in the environment,
- coordination of reactions (8-12 months) including intentional actions and early goal orientation, like planning and combining actions in order to achieve a desired effect,
- tertiary circular reactions (12-18 months) including experimentation with new behavior, when a child discovers different ways of achieving goals,
- early representational thought (18-24 months), when a child begins to develop primitive symbols representing objects or events and starts to understand the world through mental representations.

We are mostly interested in the infant capacities that appear before 8-12 months, and we concentrate on the perception part leaving out high-level planning capabilities. However, we also participate in experiments on action selection using curiosity that is related to coordination of reactions.

2.2.4 Infant self-recognition and control

Self-awareness of body, the ability to localize oneself with respect to the surrounding physical world and to control own movements are important for the understanding of the environment and for interacting with it. The ability to sense and control one's body is gradually acquired by infants during several first months of their life, over several stages of the Piaget's theory [Piaget, 1999]. From one point of view, the actions of newborns are accident and self-centered [Mahler, 2000], that fits with the infant's inability to differentiate themselves within the environment due to the lack of knowledge, although, at early stages of their development, infants already show a precursor of self-understanding in terms of sensorimotor capabilities and the ability to impact to the environment [Hart and Scassellati, 2010].

The model of an infant's body and its relation with the environment comes from coordination between the senses determined as the "ecological self" [Rochat and Rochat, 2009]. The existence of this inter-sensory model at birth has been proven by experiments. For example, infants show to open their mouth while receiving their fist, rather than placing their fist in the mouth occasionally. According to the Piaget's primary circular reactions, infants produce simple accidental movements and then repeat them for a pleasure, that eventually develops an early sensation of their body.

In the following stage of development including the Piaget's secondary circular reactions, infants begin interaction with the environment through simple repetitive object-oriented ac-

tions. The infant's knowledge about his own capability to create an affect upon the environment is determined as "self-efficacy" including the relation between his own actions and body, and objects in the environment [Rochat and Rochat, 2009]. Experiments show, that infants learn a link between their own actions and the consequences of those actions, and they understand it from few months into their life. Over a course of time, while developing the sense of self, infants acquire new capabilities through actions, continuous observation of their own actions, and their related impacts to the environment.

As a criteria for infants' self-recognition, several characteristic behaviors emerge within different stages of their development, for example observation of own movements in a mirror, or the ability to detect a red spot on their own noise [Bertenthal and Fischer, 1978].

2.2.5 Learning through interaction

Perception is only one of the sources by which information is gathered about the surrounding world. The efficiency of infant learning grows significantly through multimodal exploration, when different kinds of sensations, like visual, haptic, sound, etc, are available at the same time [Rohlfing et al., 2006]. Infants are born as active explorers of the environment, they "combine perception and action oriented to sounds they hear" [Rochat and Rochat, 2009]. At early stages of development, infants acquire basic knowledge about the surrounding world through physical interaction with objects, but at the same time and especially at later stages, infants learn in a social world through social interaction and guidance of adults [Asada et al., 2009].

2.2.5.1 Physical interaction

Babies develop through "interactions and experiences in the physical world", and their intelligence come out within the interaction with an environment and sensorimotor activity [Smith and Gasser, 2005]. If at the age of two months, an infant's exploration of an object is limited to bringing the object into the field of view and placing it into his mouths, later, infants start to explore objects manually [Rochat and Rochat, 2009]. The infant's ability to reach an objects that he see is acquired at the age of four months ; infants start to use their hands to support an object, while exploring it visually [Rochat and Rochat, 2009]. Among manual actions, infants show to transfer an object from one hand to another or to hold an object by one hand and to finger it with another hand. The fingering action is performed under visual control, and it is related to the development of coordination between visual and manual actions.

Infants continue further exploration of the world through self-initiated actions directed to elements of the environment and through observation of their own actions and the effects of those actions. The experience of action allows an infant to discover affordances

or the variety of actions that can be performed with a given object in the given environment [Gibson, 1988].

The process of infant development results in changing the visual interpretation and their sensitivity to different visual cues [Fitzpatrick et al., 2008]. The development of manipulation skills results in changing saliency of object features [Oakes and Baumgartner, 2012] ; for example, the experience of grasping a mug would result in increasing saliency of a handle. While training manipulation skills, infants learn to control their attention and to estimate dimensions of objects, including not only manipulating objects, but also objects that are not reachable. The interaction with one object helps them to make inferences about other objects, that finally, allows infants "to learn how to learn about objects" [Oakes and Baumgartner, 2012].

Knowledge explored about an object depends on the amount and the type of manual exploration performed by a child [Oakes and Baumgartner, 2012]. The amount of physical contact with objects is related to the infants' sensitivity to objects' appearance features, and infants who are more sophisticated in holding objects, are more sophisticated in learning object properties [Oakes and Baumgartner, 2012]. The variety of exploratory hand movements performed during learning about different object properties are studied in [Lederman and Klatzky, 1987], and it is reported that learning about each object property can be associated with a certain exploratory procedure.

2.2.5.2 Social interaction

Infants develop in a social world where social partners often guide them and help them to learn [Smith and Gasser, 2005]. The behavior of an adult interacting with an infant is different from the behavior of an adult interacting with another adult. Infant-directed speech usually has a high pitch and simple grammar. Adults provide a lot of supervision during infants development, for example, they repetitively name objects and properties. However, this supervision is not that direct and detailed compared to supervision used in machine learning approaches.

During their interaction with infants, the movements of adults have some motionese that attracts infants' attention and helps to identify the structure of actions and meaningful units. Comparing an infant-directed action (IDA) and an adult-directed action (ADA), the actions of adults change over a variety of characteristics, like proximity, interactiveness, enthusiasm, range of motion, repetition, and simplification [Brand et al., 2002].

2.3 Developmental robotics

Future robot will need to learn autonomously by continuously gathering new information and synthesizing it with prior experience into coherent knowledge. This continuous

lifelong and open-ended learning, acquisition of knowledge and development of skills is investigated in the field of developmental robotics [Weng et al., 2001].

Traditional research limits robots capabilities in unknown uncontrolled environments. In contrast to traditional robotics, developmental approaches take inspiration from humans that are "autonomous throughout its lifelong mental development" [Weng et al., 2001]. Developmental approaches are focused on the learning process, but not on the final performance, however, the final performance can be continuously improved during learning. Learning is expected to be incremental, the development starts from acquisition of basic capabilities and progress continuously by acquiring more complex capabilities, like in human children, through exploration, interaction with the environment, and social interaction.

The developmental approaches take inspiration from human development, formalize its theories, and adapt them to robots. The learning approaches inspired from infant development require understanding of how infants learn about the world, how infants acquire and synthesize their information into coherent knowledge, and how infants integrate their experience.

2.3.1 General approach

The "flexible and inventive intelligence" can be developed by learning, like a child, and following the six principles from developmental psychology [Smith and Gasser, 2005] :

- multimodal learning, since babies learn most efficiently, when different kinds of sensory data, like vision, touch, and sound are available at the same time,
- incremental development, since children acquire intelligence over time and throughout their development, while choosing at each stage "what to learn and in which order",
- learning through physical interaction, since babies develop through "interactions and experiences in the physical world",
- exploratory behavior, since babies explore the environment not in a goal-oriented way, but rather by playing and discovering new problems and new solutions,
- social interaction, since children develop in a society, where parents often help to learn about the world,
- symbolic communicative system, since babies use a language grounded on perception, sensorimotor and social processes.

Taking inspiration from children in choosing constraints and a priori for learning, like what to learn and in which order, autonomous learning in robots can be based on artificial curiosity [Oudeyer et al., 2007]. In psychology, curiosity is defined as spontaneous attraction toward different activities for pleasure or "a need, thirst or desire for knowledge" [Edelman, 1997], and curiosity is considered as "a motivational prerequisite for exploratory behavior" [Berlyne, 1960]. In robotics, the curiosity-driven exploration is used to

improve the learning performance with a certain notion of interest [Oudeyer et al., 2007]. The progressive development based on exploratory behavior and continuous interaction with the physical world is a promising way to design autonomous robots actively learning about its environment.

2.3.2 Application to object learning

In robotics, various research studies on autonomous object learning are inspired by different stages of infants cognitive development.

In computer vision, object perception often starts from visual features that are grouped together into objects based on common characteristics or motion, that is similar to object perception in infants at early ages, when they individuate objects based on relative motion.

Some computer vision approaches aimed at segmenting objects take inspiration from object segregation performed by infants. For example, object segmentation based on key-events, as the robot's prior experience, is described in [Fitzpatrick et al., 2008]. Key-event can be determined in different ways. Key-events can be based on object observation at a close scale [Natale et al., 2005], that is similar to infants using previously perceived isolated objects as key-events for future objects segregation. Another example of a key-event is a hitting an object by a robot hand, that is similar to an infant using previously perceived relative object motion as key-events for future objects segregation [Fitzpatrick and Metta, 2003].

Interactive object exploration performed by robots is often inspired by infants learning through continuous interaction with the physical world. The manual object exploration, like children do, is widely used in robotics society to learn about objects, their appearance, and other properties [Modayil and Kuipers, 2008], [Kraft et al., 2010], and [Rudinac et al., 2012]. Furthermore, in some research studies robots learn about objects using artificial curiosity-driven behavior [Chandrashekhariah et al., 2013] and [Nguyen et al., 2013], that is similar to intrinsic motivation of infants continuously looking for new information. Sometimes, this curiosity-driven behavior is used in combination with social guidance [Nguyen et al., 2013].

Interactive exploration of the environment enables a robot to learn not only about objects, but also about its own body. Developmental approaches aimed at learning about the robot's body, like [Natale et al., 2005], [Saegusa et al., 2012], take inspiration from infants sensori-motor development and self-recognition. Some studies go further to learn actions and affordances (like [Kraft et al., 2010] and [Natale et al., 2005]) through interaction with objects in the surrounding environment, like infants do.

2.4 Overall architecture of our work

Taking inspiration from the work presented in the previous sections, we present a developmental approach that enables a humanoid robot to learn about its close environment

by detecting and learning its meaningful elements, that we call physical entities. A physical entity can be an object, a part of the robot's body or a human part.

2.4.1 General principles

The perceptual system of the robot is designed by analogy with the human vision in terms of selective visual attention and integration of visual features into high-level visual representations building the concept of an object or in our case, a physical entity. Autonomous and incremental learning as basic principles of developmental robotics, enables the robot to continuously synthesize all newly acquired information into the coherent knowledge about physical entities, as described in Part I of this thesis. Infants learning about the world through a physical contact and a social interaction inspires active exploration of objects through interaction with them, as described in Part II of this thesis.

2.4.2 Learning through observation

In Part I of this thesis, we present the implemented perceptual approach that enables the robot to learn about its environment through observation, while a human partner interacts with a robot, like with a child. The perceptual system of the robot is designed by analogy with human perception, though we do not try to precisely reproduce human vision or inner brain functioning. Rather, we take its basic concepts described in Section 2.1 as an inspiration. We do not focus on low-level signal processing over human visual pathways, and we move further to higher-level representation, such as objects and preceding them proto-objects described in Section 2.1.3. Our perceptual approach starts from segmentation of the robot's visual space into proto-objects as units of visual attention, and we assume that the visual attention of the robot is attracted by motion that is one of components of human visual attention as described in Section 2.1.2. The segmentation of proto-objects is based on coherent motion similar to the object unity concept used by infants as described in Section 2.2.2. In our algorithm, each proto-object indicates a possibility of a presence of a physical entity. Therefore, the visual appearance of proto-objects is analyzed in order to learn or recognize physical entities. All information about the entities appearance is acquired incrementally by extracting low-level features, like colors, textures, and their spatial relations and synthesizing them into higher-level visual representations.

2.4.3 Learning through interaction

In Part II of this thesis, the robot is free to move its hands and to perform various interactive actions. The robot learns to identify parts of its own body in the visual space through motor activity, similar to self-recognition in infants as described in Section 2.2.4. The implemented self-identification algorithm is based on the mutual information between sensory

data and proprioception, and it does not require the pre-defined appearance of the robot's body. The ability to identify parts of its own body and to discriminate objects from human parts allows the robot to focus on learning objects through interaction. The interactive object learning is accomplished through manual exploration that is typical for children as discussed in Section 2.2.5. The learning algorithm incrementally acquires new information about an object appearance while it is manipulated, and this information is continuously synthesized within the already acquired object representation.

Première partie

Exploration of the robot's environment based on observation

CHAPITRE 3

State of the art : object learning

Our approach is designed for a humanoid robot that explores its close environment using mainly vision. Artificial perception based on processing of visual data has been studied for a long time in the computer vision field. There is a great number of computer vision approaches aimed at detecting and learning object in order to understand images. A short and partial review of existing object detection and segmentation algorithms is presented in Section 3.1. Objects' appearances are often analyzed using feature detection and description algorithms. The visual features that are widely used in computer vision, are described in Section 3.2. The integration of visual features into objects representations, such as appearance-based, part-based, or combined models, are presented in Section 3.3. Object learning can be performed using different methods, like generative and discriminative, described in Section 3.4. The discussion about advantages and limitations of reviewed algorithms, and the reasoning leading to our system's design choices are provided in Section 3.5.

3.1 Detection and localization of objects and proto-objects

The perception of the environment begins from localization of meaningful elements in the visual space and their segmentation from the background. Most traditional object detection approaches are based on a prior knowledge or narrow-purpose algorithms providing robust detection of specific objects of particular categories, like human faces, skin, skeleton. More generic approaches segment a scene into coherent image regions based on consistency of visual characteristics, and further group these regions into integral objects based on motion behavior or other properties. Unsupervised approaches are aimed to detect not a concrete object, but an evidence of an object existence or a proto-object, such as a region of a scene that represents a possible object or its part. Biologically motivated approaches take inspiration from human vision and detect objects or proto-objects using mechanisms of selective attention based on visual saliency.

3.1.1 Specialized detectors

Fast and robust real-time object detection can be achieved based on artificial markers used to estimate the object position relative to a camera. Methods, like the ARTag system presented in [Fiala, 2005], allow to create markers and to detect them in the environment. However, these systems require objects tagging.

Numerous narrow-purpose detectors provide efficient identification of specific object categories, like human faces, skin, hands, or skeletons. Detection of human faces based on low-level Haar-like features is proposed in [Viola and Jones, 2004]. Human skin is detected based on its color [Zhu et al., 2000], [Fritsch et al., 2002]. Detection of human skin is used to improve object segmentation, for example in [Wersing et al., 2007], objects are detected in the peripersonal space while subtracting image regions with human hands holding objects.

An efficient object detection can be achieved by using predefined object positions, for example, when objects are localized in the center of images, like in [Browatzki et al., 2012], or when the objects positions are provided by users through dedicated interfaces, like proposed in [Rouanet et al., 2009]. Other approaches take advantage of localizing objects on a plane, like a ground, a floor, or a table. In these cases, the object detection consists of estimation of a plane and using it to localize an object in a position sticking out from the plane, like performed in [Zhou et al., 2011].

3.1.2 Saliency

Unsupervised object detection is based on coherence of visual properties. For example, an object can be detected based on homogeneity of visual information, like consistency of color, texture, or 3D shape. Using these visual properties, an object can be segmented from the background, like in [Southey and Little, 2006]. In case of dynamic scenarios, an object can be detected using motion information, like in [Prest, 2012], [Beale et al., 2011], and [Katz et al., 2010].

Biologically motivated object detection is based on visual saliency that guides attention and allows to detect objects without supervision. Salient image regions differ from their neighborhood by their physical properties, like color, intensity, texture, spatial orientation, or shape. In addition, saliency also proceeds from dynamic properties, such as motion, trajectory, speed, changing size or appearance [Goldstein, 2010]. A widely used saliency-based attention model proposed in [Itti and Koch, 2001] is shown in Fig.3.1. This saliency map is obtained through combination of several image properties, and the locations with the highest saliency values are defined by a winner-take-all (WTA) algorithm. This saliency model is adapted in many research works, like [Walther and Koch, 2006], [Siagian and Itti, 2007], [Rudinac et al., 2012], and [Chandrashekhariah et al., 2013].

Object detection method proposed in [Siagian and Itti, 2007] takes inspiration from a human visual attention mechanism providing a brief understanding of a scene in a short

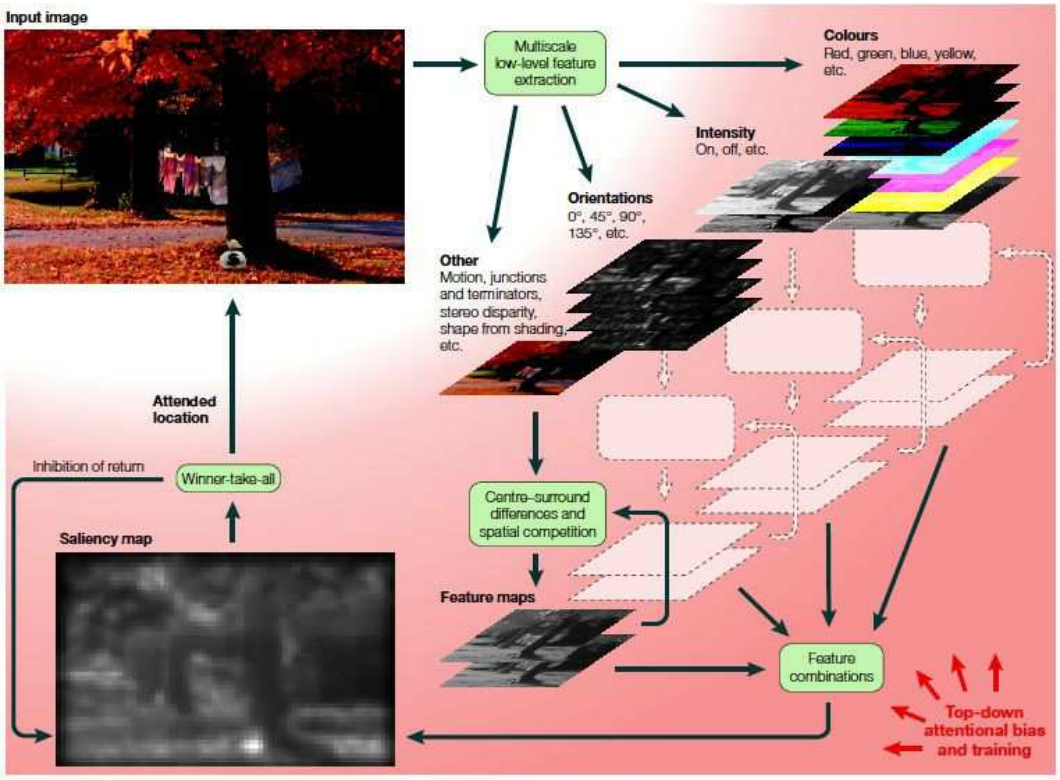


FIGURE 3.1 – Visual attention modeling [Itti and Koch, 2001]

time [Goldstein, 2010]. As discussed in Section 2.1.2, while looking at a scene, humans instantly capture "gist" of the scene that contains the most important information about the scene. The gist concept can be used for object detection, like in [Siagian and Itti, 2007], where gist is estimated from the spatial competition of early-visual features throughout the visual field. As the early-visual features, the algorithm uses the orientation estimated by the Gabor filter, color, and intensity.

The object detection approach inspired by signal processing in human vision is proposed in [Rudinac et al., 2012]. In this work, saliency guides attention in a way that allows to detect objects in unstructured environments. During the saliency computation, the visual data are processed similar to signals treated by color-opponent cells in a human eye. The spectral residuals are computed in three color channels : red-green, blue-yellow, and illumination, and they are used to estimate, how much each pixel stands out from its background. The MSER blob detector [Matas et al., 2002] is applied to the saliency map in order to precise salient regions. Then, the salient regions are clustered into visual areas corresponding to objects based on the Parzen-window density estimation [Tax, 2001] and mean-shift clustering [Comaniciu and Meer, 2002].

3.1.3 Detecting proto-objects

Research studies aimed at biologically plausible modeling of visual attention often use the concept of proto-objects, like in [Orabona et al., 2005], [Natale et al., 2005], and [Walther and Koch, 2006]. The proto-object concept is inspired by the human vision mechanism that performs perceptual grouping of information about the viewing scene into "pre-attentive" objects, as discussed in Section 2.1.3. Each proto-objects is defined as a unit of visual information, that can be bound into a coherent object, when accessed by focused attention [Rensink, 2000]. A proto-objects can be also described as a region of the visual space with "objecthood" characteristics [Pylyshyn, 2001], and this region can be segmented similar to perceptual grouping in human vision.

In [Orabona et al., 2005], images are processed in three color-opponent channels, like in human vision, as described in Section 2.1.1. Then, each color channel is used to extract edges, and a watershed transform on the edge map is used to generate proto-objects.

The attention model presented in [Walther and Koch, 2006] is based on the saliency map [Itti and Koch, 2001] described in Section 3.1.2. The regions around salient locations (see Fig.3.2) are processed as units of visual information or proto-objects that will be validated as actual objects.

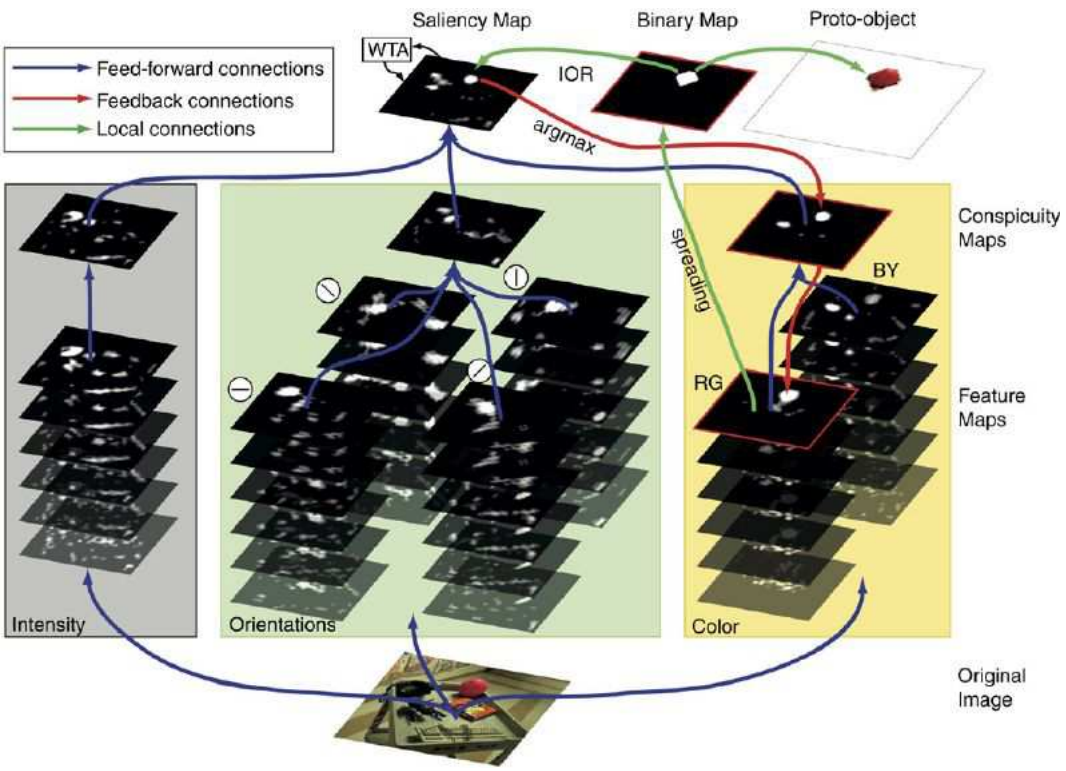


FIGURE 3.2 – Proto-objects accessed by visual attention [Walther and Koch, 2006]

The proto-object idea is used in research studies aimed at learning and recognizing

objects. In these studies, a proto-object is viewed either as a possible object, or a group of features that corresponds to an object or its part. For example, in [Natale et al., 2005], proto-objects correspond to clusters of image points grouped together based on their visual characteristics, and these proto-objects with their spatial relations are used to construct objects models. In [Walther and Koch, 2006], proto-objects are analyzed as possible objects, and attention modeling to salient proto-objects is used to recognize multiple objects in a biologically-inspired way.

3.2 Visual features

Once an object is detected, its appearance is analyzed by processing the visual data. Since the visual data are high dimensional and often contain redundant information, the visual content is characterized by more compact descriptors that allow to process the data faster and more efficiently than operating by image pixels. Each descriptor summarizes the information in a vector encoding the visual content into a significantly smaller amount of data. The descriptor computation usually implies a measure of statistical, geometric, algebraic, differential, or spatial properties of the visual data [Benois-Pineau et al., 2012].

The variety of existing descriptors characterizes not only general content information, like color, texture, shape, and motion, but also local image features representing fine visual details [Burger and Burge, 2008]. The local feature extraction has often a high computational cost, and it consists of extraction of image patches and their description. Image patches can be determined in terms of regular sampling or salient image positions. The salient positions can be localized by feature detection algorithms abstracting the image into a subset of isolated keypoints, curves, or regions. However, the density of extracted image patches should be reasonable relative to the distribution of information in the image. Images can be also characterized regularly, by analyzing them on the level of segmented regions grouping similar adjacent pixels to relatively homogeneous areas.

A good visual feature should be balanced between robustness, sparseness, speed, and completeness as the ability to preserve information. Visual features should allow to discriminate different objects and to accommodate the intra-object variation. Moreover, image matching and recognition tasks require feature detectors and descriptors that are repeatable and invariant to viewing conditions, like lighting, viewing point and orientation [Csurka et al., 2004].

3.2.1 Local descriptors

Most of local feature detectors have an associated feature descriptors [Dickscheid et al., 2011]. Local descriptors can characterize image details in a neighborhood of extracted blobs, like scale-invariant SIFT [Lowe, 2004] and SURF [Bay et al., 2008],

edges, like straight edge segment EDGE [Förstner, 1994], junctions and corners, like scale-invariant SFOP [Förstner et al., 2009] (shown in Fig.3.3). Due to the variety of existing feature detectors, we will only focus on these few that are representative and relevant for our work.

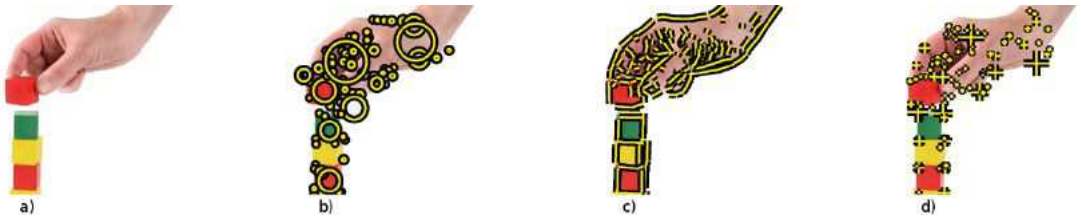


FIGURE 3.3 – Examples of feature detectors : a)the original image, b)SIFT, c)EDGE, d)SFOP junctions [Dickscheid et al., 2011]

3.2.1.1 SIFT

SIFT is one of the most popular feature detectors based on key-points used for object recognition and image matching. SIFT keypoint detection is based on the Difference of Gaussians (DoG) blob detector that is an approximation of the Laplacian operator. It achieves the scale space representation by computing the difference between two Gaussian smoothed images :

$$\nabla^2_{norm} L(u, v; \sigma) \approx \frac{\sigma}{\Delta\sigma}(L(u, v; \sigma + \Delta\sigma) - L(u, v; \sigma - \Delta\sigma)). \tag{3.1}$$

SIFT descriptors are histograms of gradient locations and orientations obtained through the following stages : scale-space extrema detection, keypoint localization, orientation assignment, and descriptor generation [Lowe, 2004]. During the first stage, the interest points invariant to scale and orientation are localized using DoG filters at different scales. Then, interest points with low contrast are removed, and the responses along edges are eliminated. The Hessian matrix is used to compute the principal curvatures and to eliminate some of keypoints based on a ratio between the principal curvatures. Around each keypoint, a descriptor is computed as a set of orientation histograms on 4x4 pixel neighborhoods. Each histogram contains 8 bins, quantizing gradient angles into 8 orientations and resulting in a 128-dimensional descriptor. SIFT algorithm is reasonably invariant to changes in illumination, rotation, scaling, and small changes in a viewpoint, but it has a hight computational cost that limits its use in real-time processing.

3.2.1.2 SURF

SURF is similar but several times faster than SIFT, since SURF relies on integral images that reduce the processing time and allow fast computation of approximate LoG images [Bay et al., 2008]. SURF feature detection is based on a scale-normalized determinant of

the Hessian (DoH) blob detector :

$$\det HL(u, v; \sigma) = \sigma^2 (L_{uu} L_{vv} - L_{uv}^2). \tag{3.2}$$

SURF descriptor characterizes 4x4 sub-regions centered around interest points by a distribution of Haar wavelet responses stored in 2x2 subdivisions. The final 64-dimensional vector results in scale- and rotation-invariant SURF descriptor.

3.2.1.3 EDGE and SFOP

As an example of a feature detector based not only on keypoints, the EDGE detector characterizes images locally by measuring the average squared gradient and the regularity of the intensity function with respect to junctions and circular symmetric features [Förstner, 1994]. This detector is based on a matrix related to the auto-correlation function computed by averaging derivatives around the image pixels in a window W :

$$A(u, v) = \begin{bmatrix} \sum_W I_u(u_k, v_k)^2 & \sum_W I_u(u_k, v_k) I_v(u_k, v_k) \\ \sum_W I_u(u_k, v_k) I_v(u_k, v_k) & \sum_W I_v(u_k, v_k)^2 \end{bmatrix}, \tag{3.3}$$

where (u, v) is the coordinates of the pixel, I is the input image, and (u_k, v_k) are the coordinates of image points in the window W.

The auto-correlation matrix determines the similarity of a patch with its neighborhood. In the case of a matrix of rank zero, a homogeneous region is detected ; in the case of a matrix of rank one, an edge is detected ; in the case of a matrix of rank two, an interest point is detected and classified into junctions or circles based on the local gradient field.

Scale-invariant SFOP detector integrates the detector [Förstner, 1994] with the spiral feature model [Bigün, 1990] that allows to obtain features, like corners, junctions, and circles.

3.2.2 Color

The color is another important characteristic of visual data. The image color content can be described by a dominant color or a distribution of colors. The computation of Color Histograms (HC) includes the quantization of a color space into bins with discrete ranges. Each bin of the histogram accumulates the number of image pixels that belong to a particular color range. HC is robust to small object deformation and scale variation, especially when the appearances of the object and the background are relatively stable [Han et al., 2011]. Although, HC does not assume color spatial distribution.

In contrast to HC, Color Autocorrelogram describes the spatial correlation of colors, and Color Layout characterizes the spatial distribution of representative colors [Benois-Pineau et al., 2012]. The color distribution can be computed based on image pix-

els or image regions.

Color descriptors can be based on different color spaces (examples are shown in Fig.3.4), like RGB used in digital devices (cameras and displays), HSV with dimensions that are similar to human color interpretation, or CIELab where Euclidean distances between colors coordinates correspond to perceived differences between colors [Burger and Burge, 2008]. The choice of a color space depends on the application and needed properties.

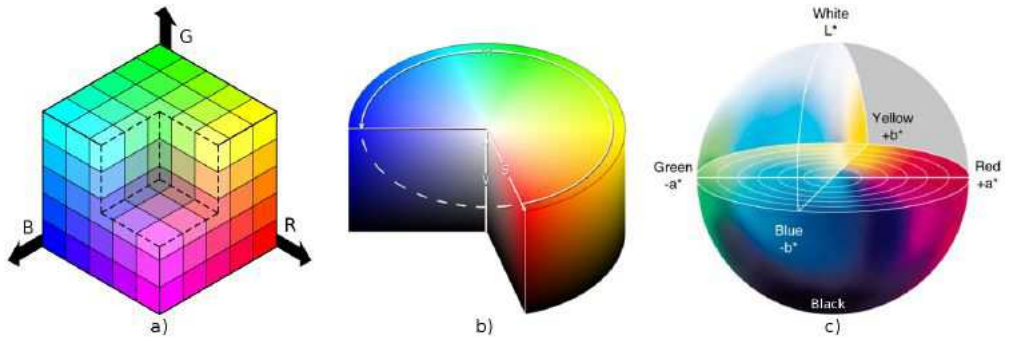


FIGURE 3.4 – Color spaces : a)RGB, b)HSV, and c)CIELab

Since color descriptors can not work well, if objects have similar colors between each other or the background [Han et al., 2011], color invariant features, like texture and contours, are widely used in the characterization of an object appearance.

3.2.3 Texture descriptors

Texture descriptors characterize the homogeneity of the visual information. Texture representation can be obtained using different approaches, such as simple means and variances of a filter bank output, wavelet coefficients, wave-packets, or model-based methods [Benois-Pineau et al., 2012].

Another interesting approach for describing the image content is the Histogram of oriented gradients (HoG) that captures edge structures characterizing local contours and shapes. However, this method is sensitive to orientation, and it can not describe well an object with large smooth areas [Han et al., 2011]. Therefore, it is well adapted to detect pedestrians that have constant orientations, but less adapted to detect objects that can have arbitrary orientations.

3.2.4 Regular image characterization

In order to describe well images with homogeneous regions, a regular image characterization is needed. A regular image characterization can be performed by segmenting an image into elementary regions and characterizing these regions.

An efficient graph-based image segmentation algorithm is proposed in [Felzenszwalb and Huttenlocher, 2004]. In this algorithm, an image is represented as a graph of elements and edges between them, where each element is a pixel, and a weight of an edge is measured based on the dissimilarity between two pixels connected by the edge. Authors define a predicate for measuring the evidence for a boundary between pairs of regions. This predicate is based on the dissimilarity between pixels along the region's boundary compared to the dissimilarity among neighboring pixels within the region. The pixel dissimilarity can be based on difference of intensity, color, motion, location or other local attributes. Being very fast, this method is often used to over-segment images in superpixels, i.e. small uniform regions, that are used as the basis for further processing.

Image segmentation based on superpixels becomes widely used in computer vision. An algorithm that segments images into high quality, compact, nearly uniform superpixels, is proposed in [Achanta et al., 2010] and known as Simple linear iterative clustering (SLIC). The algorithm performs a local clustering of pixels in a combined 5D space defined by color (L, a, b values of the CIELAB color space) and pixel coordinates (x, y).

An example of image characterization based on superpixels is presented in [Micusik and Kosecka, 2009]. In this algorithm, images are segmented into a set of superpixels that correspond to semantically meaningful objects or scene parts. These elementary regions are computed by watershed segmentation on LoG interest points as seeds. At the centers of segmented superpixels, SIFT descriptors are computed and used as appearance features characterizing image regions.

3.2.5 Feature combination

The study on features combination [Dickscheid et al., 2011] aimed at achieving maximally efficient object learning, shows, that several different descriptors provide better recognition than any descriptor used alone. The study proposes a scheme analyzing complementary features, as a minimum set of features that allows to characterize well different kinds of visual data and to avoid redundancy. Feature combination is widely used in image processing. For example, the object tracking algorithm [Han et al., 2011] is based on HC and HoG ; the object learning method [Rudinac et al., 2012] uses HC and HoG with a texture descriptor ; the image classification approach [Carbonetto et al., 2008] integrates the visual cues, like interest regions and low-level segmentation into superpixels. The concepts of several descriptors can be also integrated to construct a new descriptor that is more efficient, like Integral Color descriptor [Aldavert et al., 2010] integrating color information with a gradient-based local descriptor.

Taking inspiration from human vision, the information from several features can be integrated within an intermediate layer between low-level features and high-level visual entities, like objects. Indeed, feature extraction is a low-level image processing, that can be associ-

ated with the Early Vision stage in human vision, where the scene description is derived from simple primitives captured across the visual field [Treisman and Gormican, 1988]. However, the object concept appears in the Cognitive Vision stage. An intermediate layer that connects the Early vision and the Cognitive vision is proposed in [Krüger et al., 2010] and called Early Cognitive Vision (ECV). ECV concept allows to improve low-level processing using the assumptions from high-level reasoning. Since ECV operates contextually embedded representations of the visual information, whose level is higher than features, but lower than the object concept, we consider it as a variety of a feature combination or a features structure.

The transformation of low-level descriptors into mid-level features with richer representations and intermediate complexity can be achieved in different ways. For example, this transformation can be achieved based on coding and pooling, like in [Boureau et al., 2010]. During the coding stage, descriptors are transformed into better representations adapted to the task. Then, during the pooling stage, coded features are summarized over larger neighborhoods. The study performs the cross-evaluation of the coding methods, such as vector quantization and sparse coding, with the pooling techniques, such as taking the average (average pooling) or the maximum (max pooling).

3.3 Representation of an object appearance

Based on features and descriptors extracted from a segmented object region, an efficient representation of visual information should be able to characterize the significant content in a short description. We describe several ways of representing visual data, like appearance-based methods and part-based models that combine both appearance and geometry information. Among the variety of representation approaches, we also distinguish the Hierarchical feature model as a separate topic due to its importance in this research study.

3.3.1 Bag of Words

The representation of visual data can be performed on a local or a global level. The local representation is based on extraction of local patches (for example, by pooling techniques) or patterns of patches with a spatial order. Since matching patch-based representations can be difficult due to their spatial constraints, local patches or other extracted features can be encoded and used in global representations, like a Bag of Words (BoW).

BoW is one of widely-used methods of representing visual information. It is inspired by the approach originally invented for text processing. In text retrieval, a document is represented as an orderless collection of words. Since the occurrence of words is sparse across documents, an index maps words between documents and a dictionary, so that each document is encoded by the frequencies of its words associated with the dictionary entries. In computer vision, BoW represents images as collections of unordered features, and each im-

age is encoded by the frequencies of the corresponding visual words from the dictionary. The BoW algorithm consists of the following steps : feature extraction, feature quantization into dictionaries of visual words (also called codebooks), and training a classifier on visual words. The recognition procedure consists of extraction of features, searching the corresponding visual words in dictionaries, and applying the classifier on a set of visual words [Sivic and Zisserman, 2003].

Visual features can be obtained by any method presented in Section 3.2.1, for example, using keypoints [Sivic and Zisserman, 2003], [Filliat, 2007] (see Fig.3.5), edges [Kokkinos and Yuille, 2006], regions [Russell et al., 2006], [Borenstein and Ullman, 2002], based on image patches [Csurka et al., 2004], [Shotton et al., 2008] or based on a pixel-level [Aldavert et al., 2010]. Image patches can be obtained by applying detectors, like Harris affine detector used in [Csurka et al., 2004], or based on regular sampling used in [Shotton et al., 2008]. The image patches can be characterized based on different descriptors, like SIFT used in [Csurka et al., 2004], or even without descriptors, like in [Shotton et al., 2008] by using semantic texton forests. The approach using BoW on a pixel-level [Aldavert et al., 2010] is based on the Integral Color descriptor described in Section 3.2.5, and the main advantage of this approach is the simplicity of parallelization, since the BoW is applied locally.

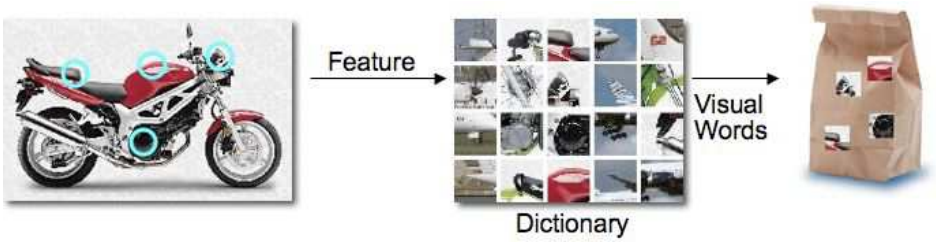


FIGURE 3.5 – The illustration of the Bag of visual Words approach

Extracted features are quantized into dictionaries of visual words, in order to reduce data dimensionality. The quantization can be achieved by iterative square-error partitioning or hierarchical techniques. The hierarchical techniques organize data into a hierarchy of clusters, like a dendrogram or a tree. These quantization techniques are not frequently used, because they require some heuristics to form clusters [Csurka et al., 2004]. The quantization based on square-error partitioning, like a K-Means algorithm, seeks for a partition minimizing the intra-cluster scatter or maximizing the inter-cluster scatter. Methods based on recursively applied K-Means algorithm are known as Hierarchical K-Means (HKM) [Nister and Stewenius, 2006]. The weakness of K-Means quantization is related to misclassification errors occurred, when a descriptor lying on a border between several clusters is assigned to a wrong visual word. This mis-clustering problem can be compensated by the Vector of Locally Aggregated Descriptors (VLAD) [Jégou et al., 2010], that accumulates

the difference between a descriptor and the corresponding visual word instead of direct descriptor quantization. The VLAD method increases the amount of information stored in a codebook compared to the HKM method, since it accumulates not only the amount of visual words but also the distribution of descriptors with respect to centers of visual words. However, both VLAD and HKM perform a hard assignment without considering visual words uncertainty and plausibility. In contrast, the spherical soft assignment [Ai et al., 2012] adaptively associates features with close visual words defined by a hyper-sphere with a radius denoted as the distance between the word and the feature.

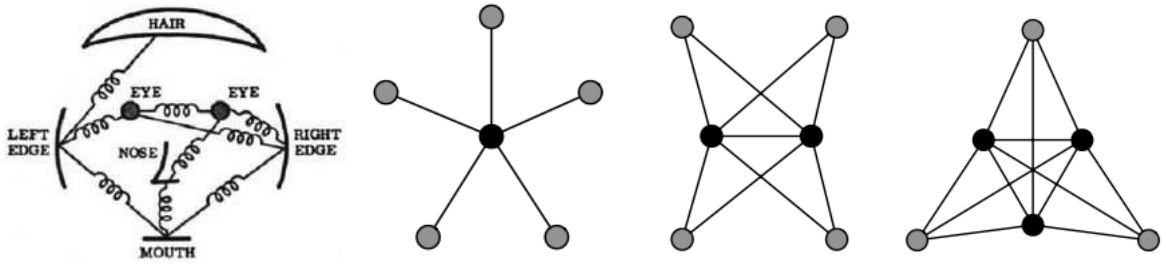
Instead of using just a list of visual words, the importance of each visual word can be characterized using Term Frequency-Inverse document frequency (TF-IDF) approach. This approach was initially used in text retrieval, where each document is described by a set of words from a dictionary and the frequencies of these words [Sivic and Zisserman, 2003]. In image recognition, it is often used to represent an image by a set of visual words from the dictionary and the frequencies of these words. TF-IDF approach is aimed to evaluate the importance of words with respect to images and to give higher weights to distinctive visual words. The inverted index allows to quickly compare an image with all memorized objects.

3.3.2 Part-based models

The main weakness of BoW approaches is the absence of a spatial relation between visual words inside images. This limitation is resolved by variations of BoW, like part-based models, Constellation models, k-fans models, etc. Part-based models combine appearance-based and geometrical models. An early example of a part-based model proposed in [Fischler and Elschlager, 1973] is shown in Fig.3.6a. Each part represents local visual properties, and the spatial configuration between parts is characterized by a statistical model or spring-connections representing "deformable" relative locations between parts. Constellation models are based on learning the geometrical relations between image parts or features, for example, local features used in [Fergus et al., 2003] or edges used in [Fergus et al., 2005] and [Shotton et al., 2005].

A family of spatial priors for part-based recognition is provided by graphical models defined by a class of graphs called k-fans and proposed in [Crandall and Huttenlocher, 2006]. The method learns both local parts appearances and a model of spatial relations between the parts. The parameter k controls the dependence between the locations of object's parts. If $k = 0$, there is no dependence between parts ; if $k = 1$, the structure corresponds to a star graph, if $k = n - 1$ (where n is the number of parts), there are dependencies between all pairs of parts. The family of k-fans models (shown in Fig.3.6b) allows to investigate how the number of spatial constraints influences the recognition performance and the computational cost. The results show, that the recognition performance depends on a particular object class.

The computation of spatial features, like in Constellation models, requires a long process-



a) b)

FIGURE 3.6 – Examples of part-based models : a) a part-based model representing a face as a collection of individual parts [Fischler and Elschlager, 1973] ; b) k-fans models (1-fan, 2-fan, and 3-fan models) based on 6 parts with the reference part shown in black [Crandall and Huttenlocher, 2006]

ing time. In contrast, a method extracting spatial features without exhaustive computation is proposed in [Yang et al., 2008]. The higher-order spatial features are obtained from lower-order features based on an additive feature selection algorithm, like boosting.

3.3.3 Hierarchical feature models

In order to construct an object representation that is more comprehensive than a collection of its features, the features are grouped into hierarchical models, like it is done in [Bouchard and Triggs, 2005] and [Kokkinos and Yuille, 2011]. A hierarchical feature model presented in [Bouchard and Triggs, 2005] accumulates both object's appearance and geometry. This model outperforms constellation models in its ability to handle many redundant features by grouping them into parts and representing an object by a hierarchy of parts, like shown in Fig.3.7. The approach evaluates the hierarchical representation build on SIFT features. However, hierarchical object representations can be based on different features, for example edges, like it is done in [Kokkinos and Yuille, 2011].

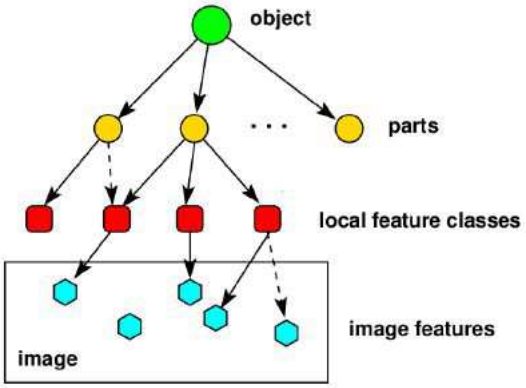


FIGURE 3.7 – Object hierarchical representation [Bouchard and Triggs, 2005]

3.3.4 View-based models

Since in the real world most of objects are 3-dimensional, they result in a variety of 2D projections in images [Ullman, 1998]. In order to overcome this issue, an overall object appearance can be characterized by a view-based model accumulating the object's appearance from different viewpoints. For example, a view-based model considering spatial relations between views is presented in [Paletta and Pinz, 2000]. However, in this approach, object's viewpoints are learned in a supervised way using a controlled turn-table that rotates the object each time on 30°.

On the other hand, views can be similar between objects and significantly different for a single object depending on a viewing angle, distance from the object, and lightning conditions. A computational approach using combinations of views in order to deal with a varying viewing position and illumination direction, is presented in [Ullman, 1998]. In this approach, a 3D object is represented by a linear or non-linear combination of 2D views.

3.4 Learning methods

As soon as available visual data are characterized by visual features or higher level representations, one of learning methods can be used to associate each representation with an object label. Among machine learning algorithms, we give a short description of two major philosophies : generative and discriminative. Also we describe incremental learning methods, which provide an ability of online learning for autonomous robots.

3.4.1 Generative methods

Generative approaches describe data by structured probabilistic models or estimate the joint probability distribution over observations and labels, where the observations are random variables, whose distributions depend on the model's parameters. In the field of object learning, generative approaches are used to estimate the distribution of data samples which belong to each data class. Generative models include three main stages : evaluation of the probability of an observation, estimation of the model's parameters from the observed data, and running the model forward to generate new data.

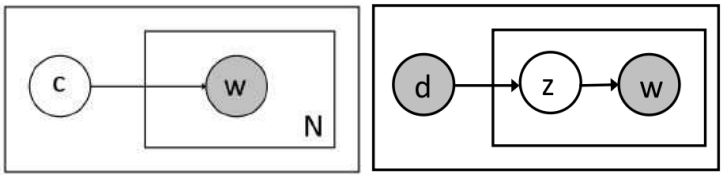
Naive Bayes (used in [Csurka et al., 2004]) is a generative probabilistic classifier based on the assumption, that observed variables have own distributions, and these distributions are significantly different between data classes. In the context of object learning, it means, that visual representations, like a collection of features, are significantly different between objects. The graphical model of this approach is shown in Fig.3.8a. During training, the classifier learns the distribution of visual representations from objects samples. The recognition is achieved by applying the Bayes' theorem with strong independence assumptions and

choosing a maximum posteriori decision :

$$c* = \arg\max_{c} p(c) \prod_{j=1}^{N} p(w_j|c)^{n(w_j)}, \tag{3.4}$$

where c is the object label, w_j is a visual representation of the object (can be a collection of features), and $n(w_j)$ is the number of times the visual representation was seen.

Among Hierarchical Bayesian models applied to unsupervised object categorization is the Probabilistic Latent Semantic Analysis (pLSA) (used in [Sivic et al., 2005]). pLSA is a two-level generative model originally developed for the statistical text literature, where each document corresponds to a mixture of topics, and each topic has its own distribution of words. The graphical model of this approach is shown in Fig.3.8b.



a) b)

FIGURE 3.8 – The graphical models : a) the Naive Bayes classifier, where c is an object label, w is a visual object representation among N representations ; b)Probabilistic Latent Semantic Analysis (pLSA), where d is an object label, z is topic, and w is a visual representation

3.4.2 Discriminative methods

Discriminative methods are aimed at learning the differences between several data classes based on a decision rule called a classifier, that associates each data sample with one of the possible classes. In object learning, the classification is performed on data samples that correspond to collections of visual features or higher level representations. In this case, the classifier associates an available representation mapped as a point to some space, like a feature space, with one of objects. Decision rule divides the input space into regions separated by decision boundaries.

Nearest Neighbor Classifier (used in [Lowe, 2004]) is one of discriminative methods, that assigns a label of the nearest data sample from the training set to each test sample. K-Nearest Neighbors method searches the k closest data samples from the training set, which vote to classify a new data sample. The method works well in the case of many data samples and an appropriate distance function. Among the distance functions widely-used for comparison of feature histograms are heuristic distances, like Minkowski-norm distance (ℓ_p norm), nonparametric statistics testing the hypothesis that two empirical distributions have been generated from the same true distribution, like χ^2 distance, or ground distance measure, like

Earth Movers Distance [Burger and Burge, 2008].

Support Vector Machine or SVM (used in [Csurka et al., 2004], [Yang et al., 2008], and [Ude et al., 2008]) is an example of discriminative non-probabilistic classifiers. This classifier is used to distinguish between two classes of data or between positive and negative samples of a particular data class. During training, SVM searches a hyperplane separating data classes with a maximal margin between positive and negative samples and maximizing the distance between the hyperplane and the closest data sample. The recognition of a data sample mapped to the feature space is based on its position relative to the hyperplane. Since not all data classes are linearly separable, several modifications can be introduced to the model : misclassified samples can be penalized proportionally to their distance to the decision boundary, or the visual representations can be mapped from the original space to another space that can have a higher dimensionality [Csurka et al., 2004]. Multi-class problems can be solved by training several SVM classifiers.

Considering the concept of discriminative models aimed at learning differences between classes, these models are especially useful, when data classes are similar. Though, generative models performs better, when data samples are rare. The comparison of SVM and Naive Bayes classifies applied to image classification reports the superiority of SVM results over the results obtained with a Naive Bayes classifier [Csurka et al., 2004].

3.4.3 Incremental learning

In general, it is easy to update a generative method with a new example, since we just need to update the class statistics. It is also possible to learn incrementally using discriminative methods [Cauwenberghs and Poggio, 2001] or using gradient descent as in Neural Networks. A voting method, like the one proposed in [Filliat, 2007], is also easy to use incrementally. The algorithm computes the statistics of visual words seen among images, and this statistic is used during recognition. Updating statistics during learning is very fast, as it simply entails adding counts to the numbers of features viewed for each object. While incremental learning is an active research area, we will rely on this simple method that is well adapted to the representation of objects using the Bag of Words approaches.

3.5 Conclusion

There is a variety of computer vision approaches aimed at detecting objects in images and learning their visual appearances. Our work is focused on a perceptual approach that requires a minimum prior knowledge and minimum supervision. We are interested in a generic method that works in an unstructured environment and allows to detect different types of objects without predefined objects appearances.

Among object detection algorithms, we find interesting the proto-object concept used

often in biologically-motivated architectures. In our approach, we are going to detect proto-objects based on saliency in the visual space.

The studies on complementary features inspire us to choose a set of features that would maximize the encoded information while characterizing different types of objects. We choose SURF and colored superpixels as complementary features that should characterize well both simple homogeneous objects and complex textured objects. In order to incorporate local object geometry, we analyze relative features positions and group them into more complex features.

As an object representation, we are going to use Bag of Words with TF-IDF. However, each object will be characterized not by a collection of features, but rather by a multi-view model, where each view, encoded by its features, describes the object's appearance from one perspective. As a recognition/learning approach, we will use voting that is fast and well adapted to incremental learning.

CHAPITRE 4

Perceptual system implementation

In this chapter, we describe the proposed perceptual approach that allows the robot to learn about its close environment through observation, while a human partner interacts with the robot and demonstrates different objects. The robot's perception starts from segmentation of the visual space into proto-objects defined as salient units of attention. The procedure of proto-object detection and segmentation is presented in Section 4.1. The visual appearances of each proto-object is analyzed as described in Section 4.2, and it is learned or recognized as one of physical entities that can be an object, a part of a human partner or a part of the robot's body. The appearance of each entity is characterized by a multi-view representation model, where each view describes the entity's appearance from one perspective. The learning and recognition procedures are detailed in Section 4.3. The main modules of the implemented perceptual system are shown in Fig.4.1.

Our approach is based on online incremental learning, and it does not require image databases or specialized face/skin/skeleton detectors. All knowledges are iteratively gathered by analyzing the visual input and integrating extracted low-level information into hierarchical representation models of physical entities. The visual input is acquired from a RGB-D sensor (Kinect) using the OpenNI [1] library. The RGB-D sensor is chosen as a source of visual information due to its efficiency and precision of 3D data in comparison with stereo vision based on the robot's cameras.

4.1 Detection and segmentation of physical entities as proto-objects

In this part of the thesis, the exploration of the robot's environment is performed through pure observation, so we analyze the robot's environment within its visual space. The visual space covers a part of the surrounding environment that falls into the field of view of the visual sensor. Given our setup, the position of the visual sensor allows to observe the interaction area (shown in Fig.4.2b) including the table placed in front of the robot, some parts of

1. http ://openni.org

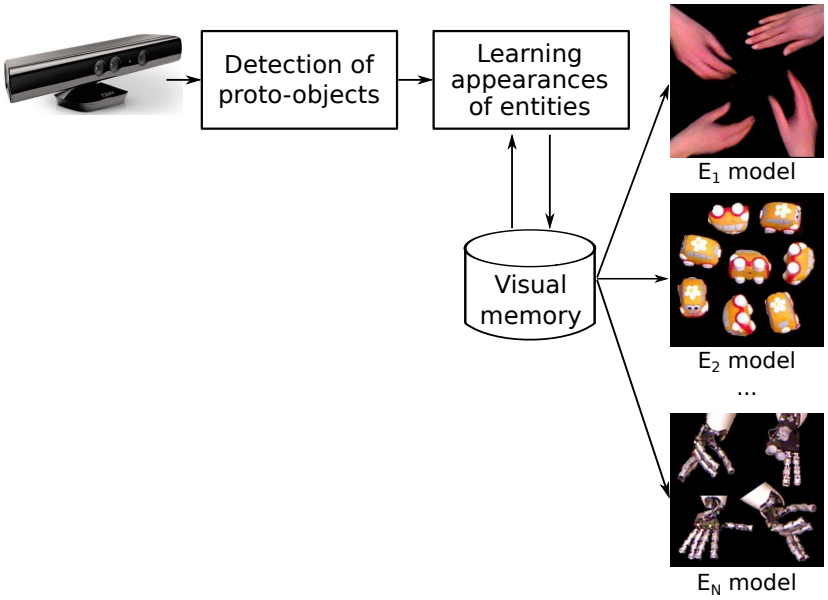


FIGURE 4.1 – The main modules of the perceptual system developed for a robot learning about its environment through observation, where E_1, E_2, ..., E_N are the physical entities detected in the visual space, learned, and stored in the memory

a) b)

FIGURE 4.2 – The visual space of the robot : a)the position of the robot relative to its interaction area, b)the visual field

a human partner, and some parts of the robot's body.

The visual space is segmented into proto-objects as units of visual attention that can be later identified as possible physical entities. The main steps towards segmentation of the visual space into proto-objects and the intermediate results of image processing are shown in Fig.4.3.

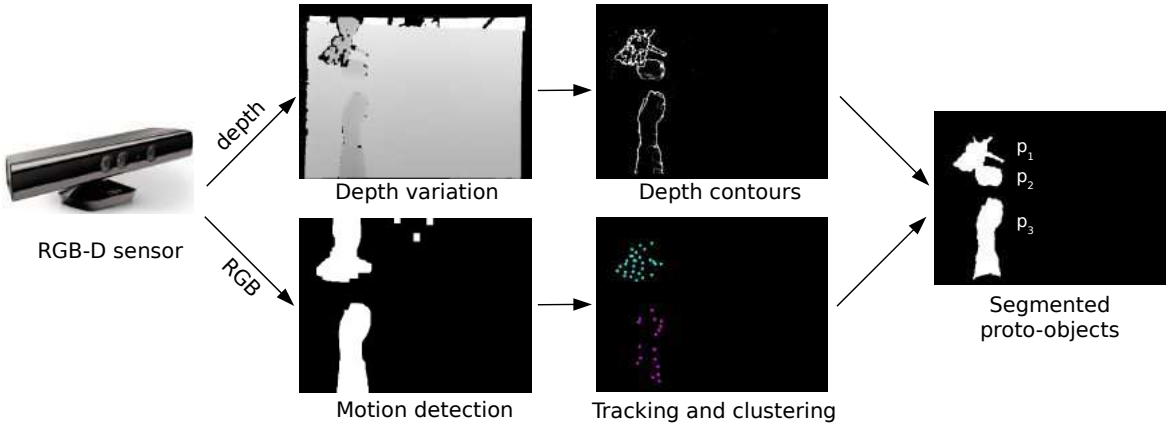


FIGURE 4.3 – The main stages of segmentation of the visual space into the proto-objects (p_0, p_1, p_2) and the corresponding image processing results

4.1.1 Motion processing

The proto-object detection begins with visual attention. Taking inspiration from human perception described in Section 2.2, we use an attention mechanism based on saliency in the visual space. In human perception, among various factors of saliency, motion carries a significant part of information about events happening in the environment and their actors [Goldstein, 2010]. In the case of our scenario, regions moving in the visual space often represent parts of a human partner, parts of the robot's body, or manipulated objects, that exactly correspond to entities we need to detect in the visual space. Therefore, motion is chosen as the main source of attention. Moreover, a human partner can attract the robot's attention by simply interacting with an object that produces observed motion encouraging the robot to focus on the object.

Among all possible approaches of motion detection in image processing, we use the Running average [2] based on image differencing. The Running average of an image sequence is computed as a weighted sum of the current image and the accumulator :

$$acc(u,v) \leftarrow (1-\alpha) \cdot acc(u,v) + \alpha \cdot image(u,v) \tag{4.1}$$

where $acc(u,v)$ is a pixel of the accumulator at the position (u,v) ; $image(u,v)$ is a pixel of

2. implemented in the function AccumulateWeighted of the OpenCV library http ://opencv.org

the input image at the position (u, v), and α is the speed of updating previous images with a new image, $\alpha = 0.2$ is used in our algorithm.

The computed running average is subtracted from the current input image, and the obtained image is thresholded into a binary image whose pixels are either white (foreground) or black (background). The choice of the threshold value is grounded on depicting moving areas by white pixels while filtering out noisy pixels. The obtained binary masks corresponding to moving areas of the visual field are shown in Fig.4.4b.

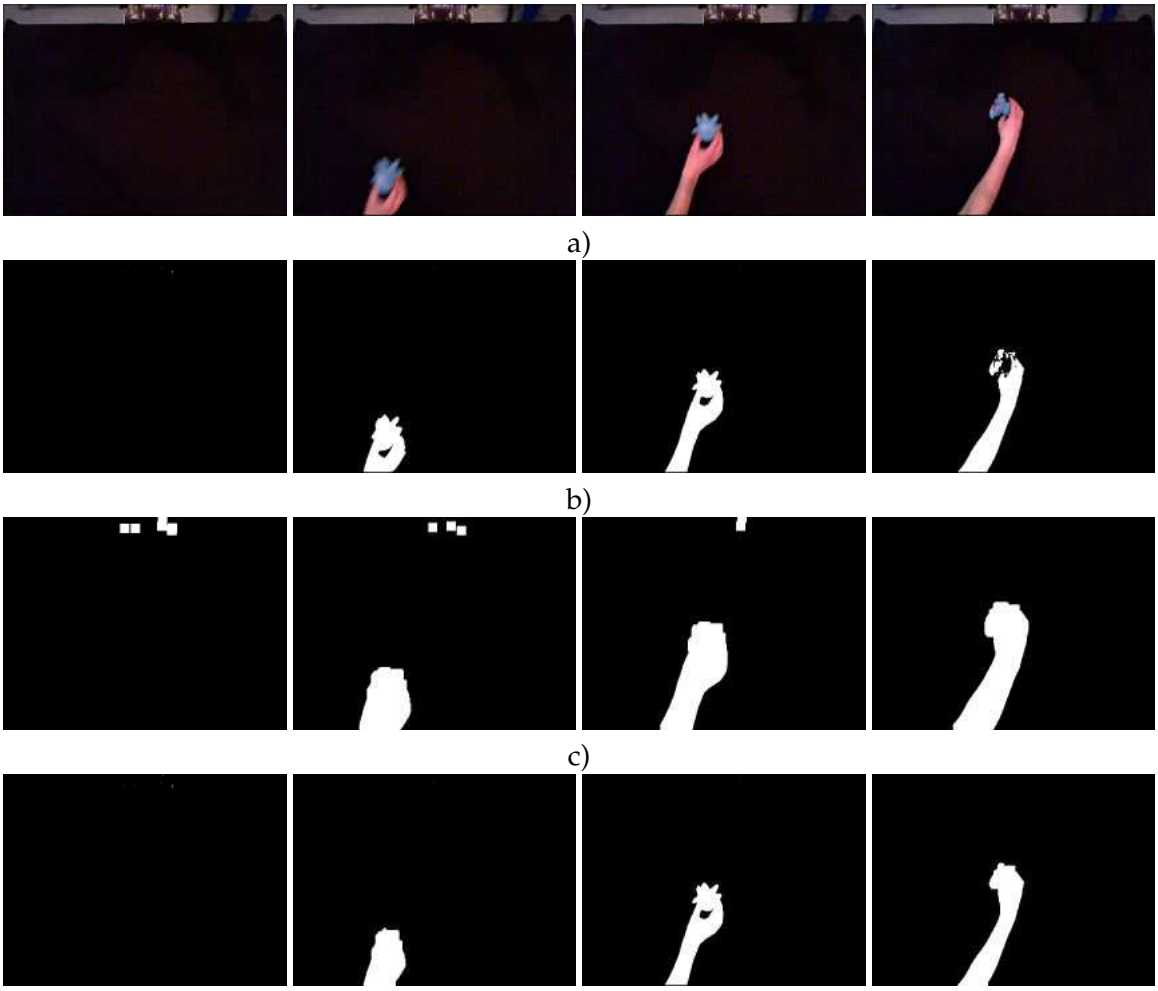


FIGURE 4.4 – Motion detection in the sequence of four images : a)input images, b)moving regions detected by the Running average method, c)the effect of the dilation operation on the moving regions (closing holes), d)the effect of the erosion operation on the moving regions (shrinking regions and erasing noise)

Noisy pixels among the detected moving regions are removed by erosion and dilation operators defined in mathematical morphology [Shih, 2009]. The dilation operation is used to expand image regions (regions of white pixels in case of a binary image), and it consists

of convolution of the current image with a specified kernel (or a structuring element), which determines the shape of the analyzing pixel neighborhood. A structuring element usually has a shape of a square or a cross with a varying size, like shown in Fig.4.5.

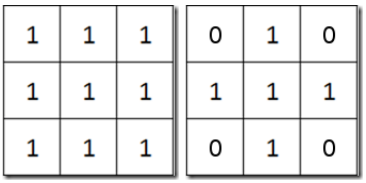


FIGURE 4.5 – Structuring elements : a)3x3 squared structuring element (considers 8-connectedness), b)3x3 cross-shaped structuring element (considers 4-connectedness) ; both structuring elements have the origin (or the anchor) at the element's center ; foreground regions are encoded as one, and background regions are encoded as zero

The dilation operation consists of replacing each image pixel by the maximum value of its neighborhood defined by the structuring element :

$$image(u,v) = \max_{(u',v'):\, element(u',v') \neq 0} image(u+u', v+v'), \tag{4.2}$$

where $image(u,v)$ is a computed pixel of the output image at the position (u,v) ; $element(u',v')$ is a pixel of the structuring element at the position (u',v'), and $image(u+u', v+v')$ is a pixel of the input image at the position $(u+u', v+v')$ that lies in the neighborhood of the computed pixel (u,v).

In case of 3x3 squared structuring element (considering 8-connectedness), the dilation operation on a binary image replaces each black pixel by a white pixel, if it has at least one adjacent white pixel. The dilation operation closes holes in foreground regions (white pixels, in our case), but also it enlarges foreground regions, thus, we further apply an erosion operation in order to shrink foreground regions.

The erosion operation shrinks foreground regions and filters out noisy regions with a size smaller than the size of the structuring element. The erosion operation consists of replacing each image pixel by the minimum value of its neighborhood defined by the structuring element :

$$image(u,v) = \min_{(u',v'):\, element(u',v') \neq 0} image(u+u', v+v'). \tag{4.3}$$

Both dilation and erosion operations can be applied several times in order to augment their effects. In our algorithm, we perform ten iterations of each dilation and erosion operations with a 3x3 squared structuring element. The dilation operation is used to close holes in the moving regions detected by the Running average method, then the erosion operation is used to filter out noise allowing to improve the coherency of the moving regions (as shown in Fig.4.4).

The motion processing is extremely important in our approach, since it generates a large

amount of information about the surrounding environment. In our setup, the RGB-D sensor is fixed and does not move during the robot's movement. Thus, the moving areas of the scene include only real motion from external sources (like the robot or its human partner). If the visual input is taken from the robot's cameras, the robot's movement influences the visual scene, and the moving areas include both real motion and camera motion. In this case, the camera motion can be subtracted by analyzing the optical flow and filtering out all background pixels that move with the same speed.

Furthermore, all moving regions are filtered based on the constraints of the robot's working area, and the visual regions that are unreachable for the robot are ignored. The working area depends on the length of the robot's arm, the pose of the torso, and the height of the table. For the iCub robot, the reachable working space is considered to be an area of 45 cm in radius centered on the robot's base (as shown in Fig.4.6), since the robot was able to reach objects localized in this area, during our experiments.

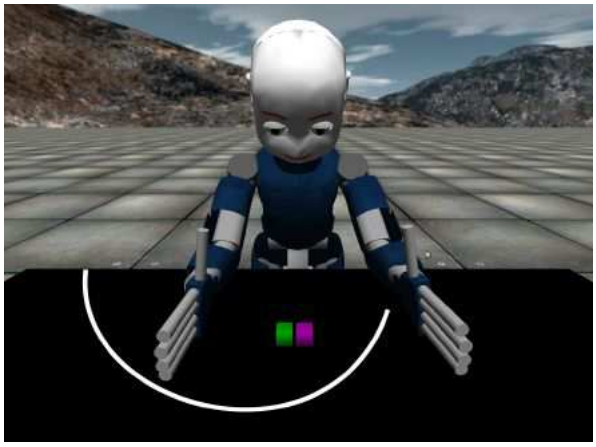


FIGURE 4.6 – The approximate reachable area for the right hand is shown by white curve

4.1.2 Isolation and tracking of proto-objects

The final moving areas of the visual field are analyzed as probable locations of proto-objects. Inside each moving region, we extract Good Features to Track (GFT) [Shi and Tomasi, 1994], since these features are especially developed for the tracking purpose. The feature detector estimates the corner quality measure at each image pixel by computing the local intensity variation matrix (auto-correlation matrix given in equation (3.3)) averaging derivatives in the 3x3 pixel neighborhood, as described in Section 3.2.1.3. Among the image patches with a high variation of intensity in both horizontal and vertical directions, the GFT is detected in the case of significant eigenvalues of the auto-correlation matrix. In our algorithm, we extract 80 GFT points in each image while preserving at least 15 pixels between the points in order to impose their distribution in space.

GFT points are tracked between consecutive images using the Lucas-Kanade method [Bouguet, 2001], that is chosen thanks to its small processing cost, accuracy, and robustness. The Lucas-Kanade tracking method is based on computation of an sparse optical flow for a set of chosen features. The robustness of the tracker, like its sensitivity to motion, depends on the size of the search window and the number of analyzed image scales. In order to achieve the balance between the tracking accuracy and its processing cost, we set the size of the search window to the minimum distance between GFT points, and we choose a suitable number of iterations. Examples of extracted and tracked points are shown in Fig.4.7.

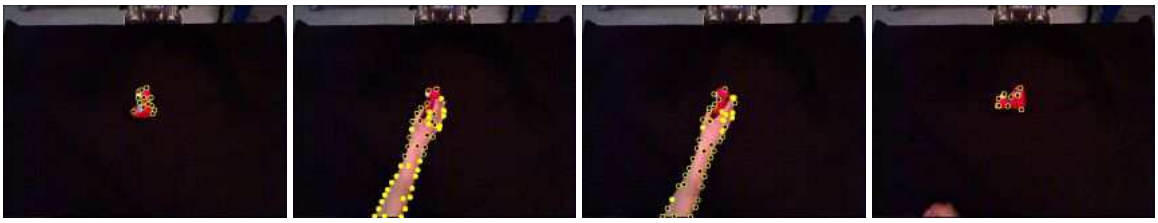


FIGURE 4.7 – Examples of extracted and tracked GFT points in the sequence of four images : all extracted points are shown by big yellow circles, and tracked points are marked by small black circles inside yellow circles

The motion behavior of tracked points is analyzed, and the presence of uniform motion allows to isolate proto-objects inside moving areas of the visual space. Tracked points are grouped into clusters using the agglomerative clustering approach based on position and velocity as we described below. Initially, each tracked point composes its own cluster ; then, each iteration, we merge two clusters with a smallest distance measure given in equation (4.4), and we recompute the distance measures. The clustering process is repeated (as shown in Fig.4.8) until reaching a specified threshold on the minimal distance measure. The examples of obtained clusters are illustrated in Fig.4.9.

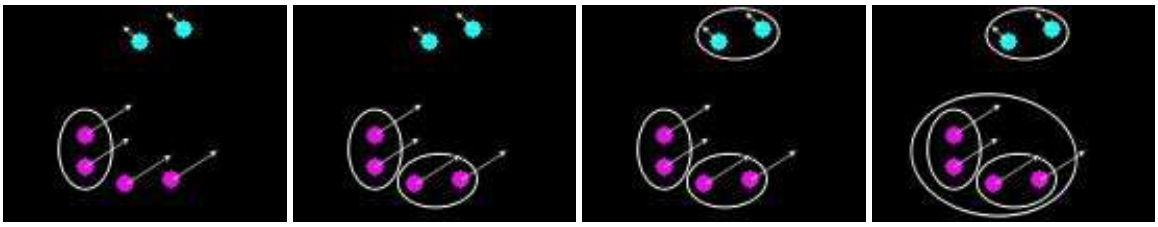


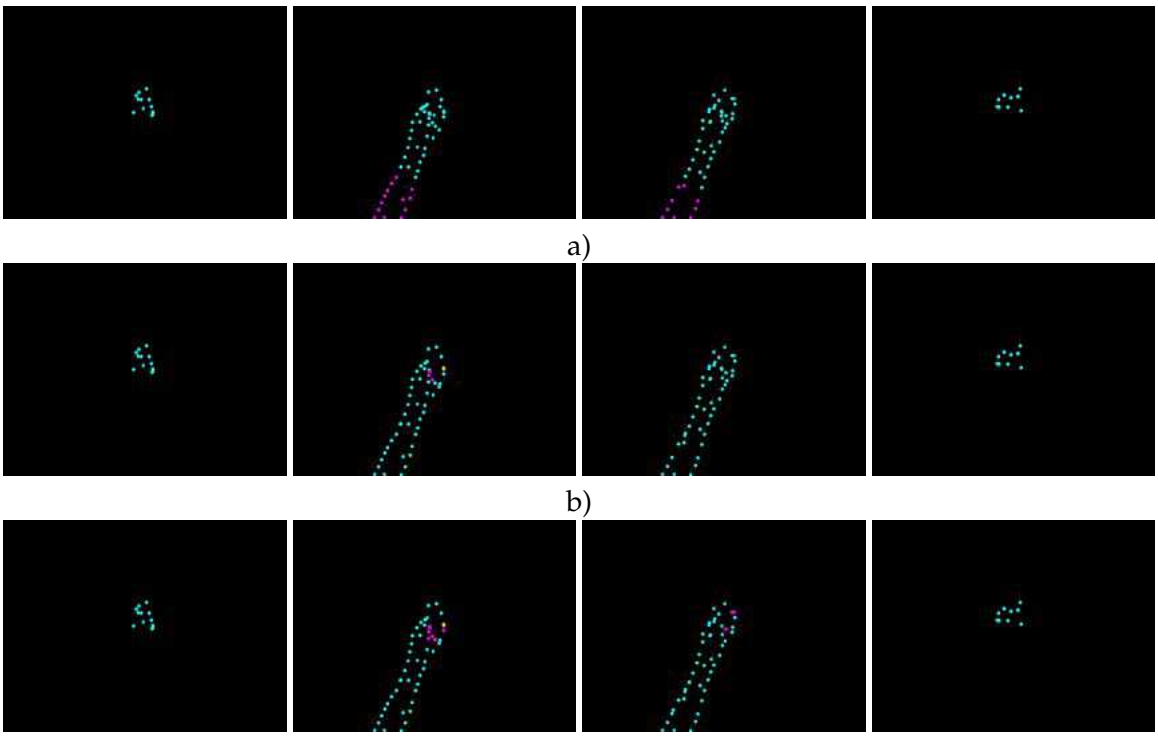
FIGURE 4.8 – Agglomerative clustering : GFT points are shown by colored circles with their direction of motion with respect to the previous image, the clusters obtained after each iteration are shown by big white ovals, a point's color indicates its final cluster

Each cluster of GFT points could be characterized by an average position of its points, and in this case, clusters can be compared based on their relative position computed as the Euclidean distance. However, this measure based on relative position of clusters is not perfect, since it does not consider the direction of motion. In order to incorporate the direction of

motion, we compare clusters based on their relative velocity. However, this measure based on relative velocity results in a single cluster, if all points are static. Thus, the final distance measure between each pair of clusters is based on their relative position and velocity :

$$d(c_i, c_j) = ratio * \Delta velocity(c_i, c_j) + (1 - ratio) * \Delta position(c_i, c_j); \tag{4.4}$$

where $d(c_i, c_j)$ is the distance measure between two clusters c_i and c_j, $\Delta position(c_i, c_j)$ is the Euclidean distance between the clusters' positions, $\Delta velocity(c_i, c_j)$ is the difference in the clusters' velocities, and the *ratio* is the priority to one of characteristics, that is velocity in our case.



a)

b)

c)

FIGURE 4.9 – Examples of clustering based on different distance measures (GFT points of the same cluster are shown by the same color) : a)clustering based on relative position of points, b)clustering based on velocity of points, c)clustering based on position and velocity of points

Once the GFT points are clustered, each cluster identifies a possible proto-object. A proto-object is considered as tracked from the previous image, if more than a half of its points is tracked, otherwise, it is considered as not tracked. In our algorithm, GFT points are extracted only in moving image regions that are not always aligned with real objects' boundaries, and thus, not always can be tracked. Since the proto-object segmentation can vary between images, the probability of tracking points localized closer to the proto-object's center should be higher compared to points localized on borders. Therefore, we choose several GFT points

closest to a proto-object's center as reference points, and our tracking decision is based on these reference points.

4.1.3 Extraction of proto-objects contours

Each group of coherent GFT points is analyzed as a proto-object. Using vision only, each proto-object can be segmented from the background based on a convex hull of its tracked GFT points. However, a convex hull does not always correspond to the real object's boundary. If a convex hull is based on few GFT points, it often cuts the proto-object or captures the background and surrounding items, as shown in Fig.4.10. In order to improve the proto-object segmentation, the results of tracking performed on RGB images are consolidated with processing of the depth data, and the depth variation in the visual field is analyzed to obtain more precise proto-object's boundaries. If convex hulls of GFT points could group together several static objects localized near to each other, then the depth processing allows to isolate the corresponding proto-objects inside a single convex hull.

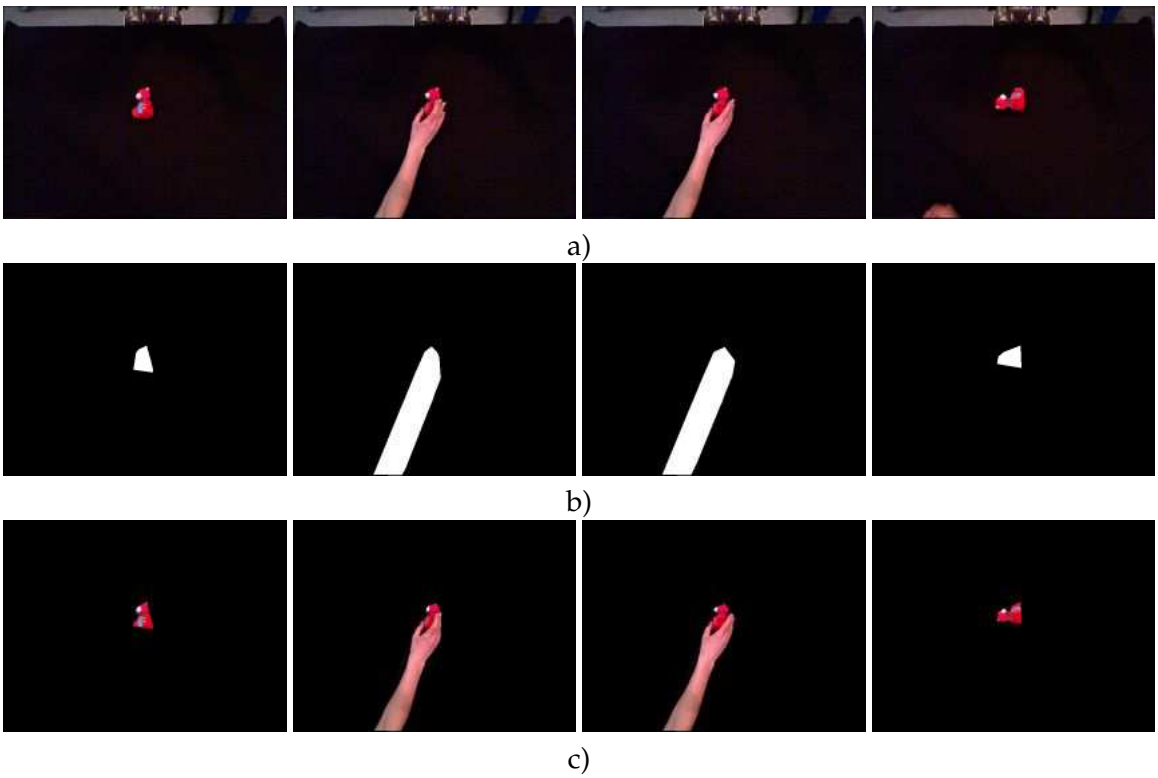


FIGURE 4.10 – The proto-object segmentation based on convex hulls of the GFT points a)input images, b)convex hulls of proto-objects' GFT points, c)resulted proto-object segmentation based on convex hulls of the GFT points. In all images, some parts of proto-objects are cut ; in images with a human hand, the proto-object's region captures partly the table near the hand

At first, the Median blur filter is applied to smooth pixels values in order to reduce the

noise in the input depth data. Then, the Sobel operator based on the first derivative is used to detect horizontal and vertical edges allowing to reveal the depth variation in the visual field. Noisy and non-significant edges are filtered out by thresholding and normalizing the obtained results as shown in Fig.4.11.

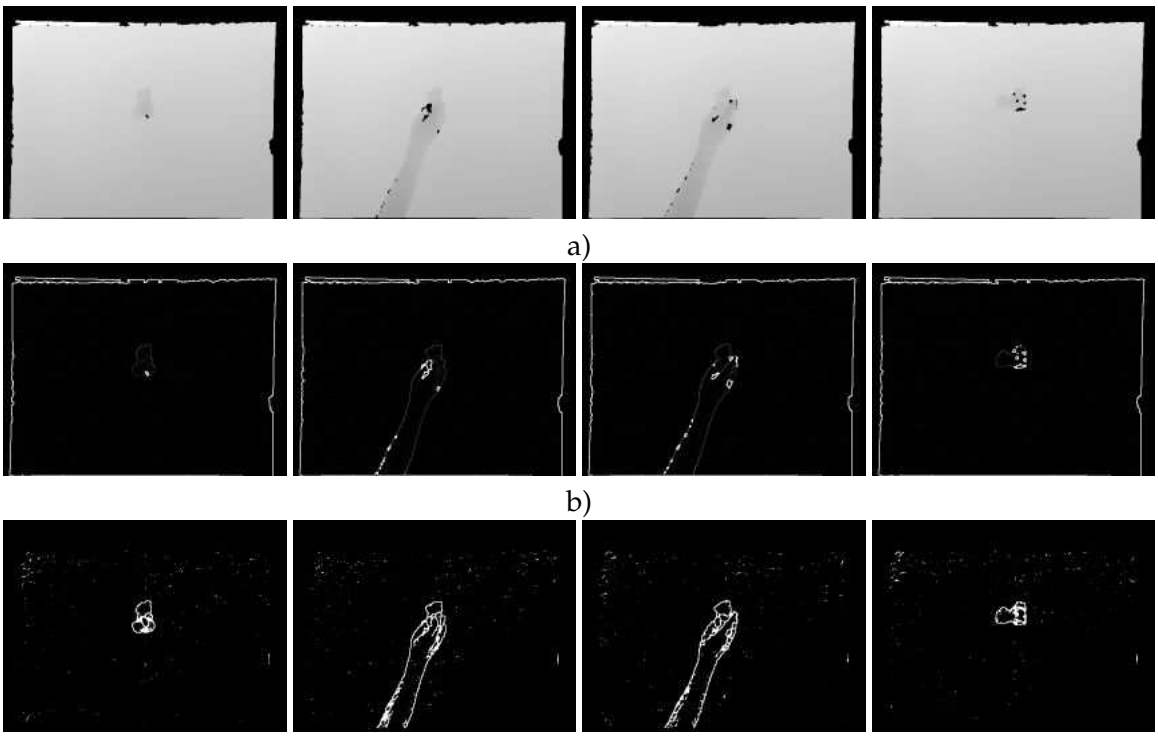


FIGURE 4.11 – Edge detection based on the Sobel operator : a)input depth data visualized in shadows of gray, b)detected horizontal and vertical edges, c)thresholded edges

The obtained edges are not always continuous, thus, the dilation and erosion operations with the 3x3 squared structuring element are used to close broken contours. The obtained continuous contours are transformed into binary masks, as shown in Fig.4.12a. Since the contour detection method is based on the depth data, the contours passing through regions with small depth variations often stay open. It happens in case of long contours and contours crossing image borders, and these contours usually correspond either to parts of the robot or its human partner, or the table's boundaries. Therefore, we filter out possible boundaries of the table, and we perform closing of other contours. If a contour crosses an image border, we connect each pair of its closest points lying on the image border, and we repeat this procedure until the contour is closed. Among other open contours, we close only the longest one considering it as the most meaningful. We traverse the contour, and we close all gaps through interpolation of the contour's nearest points. The final closed continuous contours define proto-objects boundaries, and these contours are used as binary masks in order to

segment proto-objects, like shown in Fig.4.12.

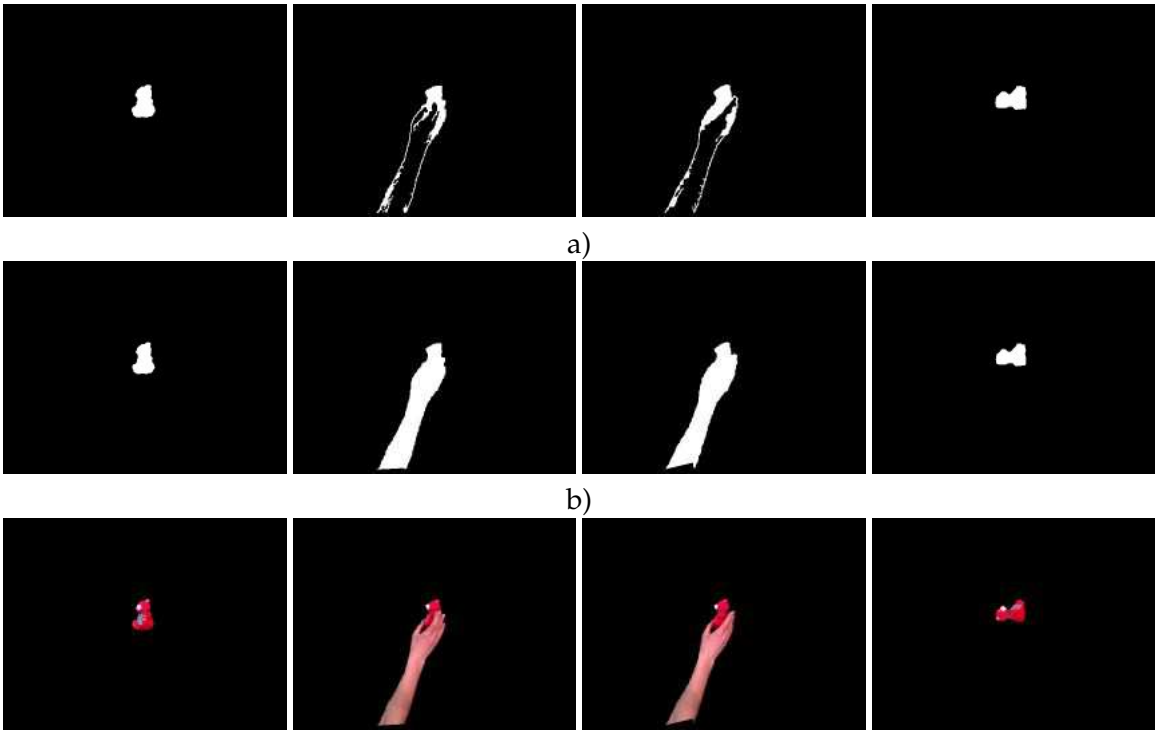


FIGURE 4.12 – The segmentation of proto-objects based on depth contours : a)detected contours transformed into binary masks, b)binary masks resulted after closing the longest contours, c)proto-object segmentation based on depth contours used as binary masks

The motion artifacts, like blur, changing pixels intensities and colors, make it difficult to extract continuous contours and to match GFT points needed for tracking. Therefore, we process only images that contain enough of meaningful information, i.e. images with a specified minimum of GFT points needed as a reference for one proto-object (as discussed in Section 4.1.2), and with at least one closed contour extracted inside a moving area of the visual field.

4.2 Entity representation

The appearance of each segmented proto-object is characterized by low-level *features*, as described in Section 4.2.1. The low-level features are integrated into more complex features that encode a *view* characterizing the *entity's* appearance from one of its perspectives, as described in Section 4.2.2. The overall entity appearance is represented as a *multi-view representation model* described in Section 4.2.3.

4.2.1 Visual feature extraction

The robot should be able to deal with various kinds of objects, ranging from simple homogeneous objects with few features, to complex textured objects, like shown in Fig.4.13. A single descriptor can not provide a good representation of various visual characteristics, as discussed in Section 3.2, since a general descriptor usually analyzes one type of the visual properties, like color or texture ; in contrast, a local descriptor analyzes only an image area around a key-point. By analyzing the complementarity of different features, we choose a combination of features that describe different visual characteristics and maximize the amount of encoded information.

FIGURE 4.13 – Examples of objects

As a local descriptor based on key-points, SURF shows an efficient and accurate characterization of isolated image areas, thus providing a good description of objects with many details. SURF algorithm can be used to detect key-points and encode their neighborhoods with a 64-dimensional vector, as described in the Section 3.2. However, the detected key-points are isolated and sparse, as shown Fig.4.14. Thus, SURF descriptor used alone does not allow to characterize well homogeneous object regions.

In order to deal with both homogeneous and textured objects, visual information should be analyzed not only around isolated key-points but also with some regularity. The regular image characterization can be achieved by describing the visual content around regularly extracted key-points or segmented image regions. In our experiments reported in Section 5.2, we analyzed the dense-SURF descriptor as a SURF descriptor applied on regularly extracted key-points. According to our results, the dense-SURF descriptor do not perform better than the standard SURF in our application. Therefore, we develop an additional descriptor operating on the level of regularly segmented image regions.

The superpixels algorithm [Micusik and Kosecka, 2009] is used to segment images into relatively homogeneous regions by grouping similar adjacent pixels using watershed segmentation on LoG (Laplacian of Gaussian described in Section 3.2) with regularly spaced seeds. Each obtained superpixel is characterized by the average color of its pixels, as shown in Fig.4.15. The color is encoded in the HSV space (hue, saturation and value) described in

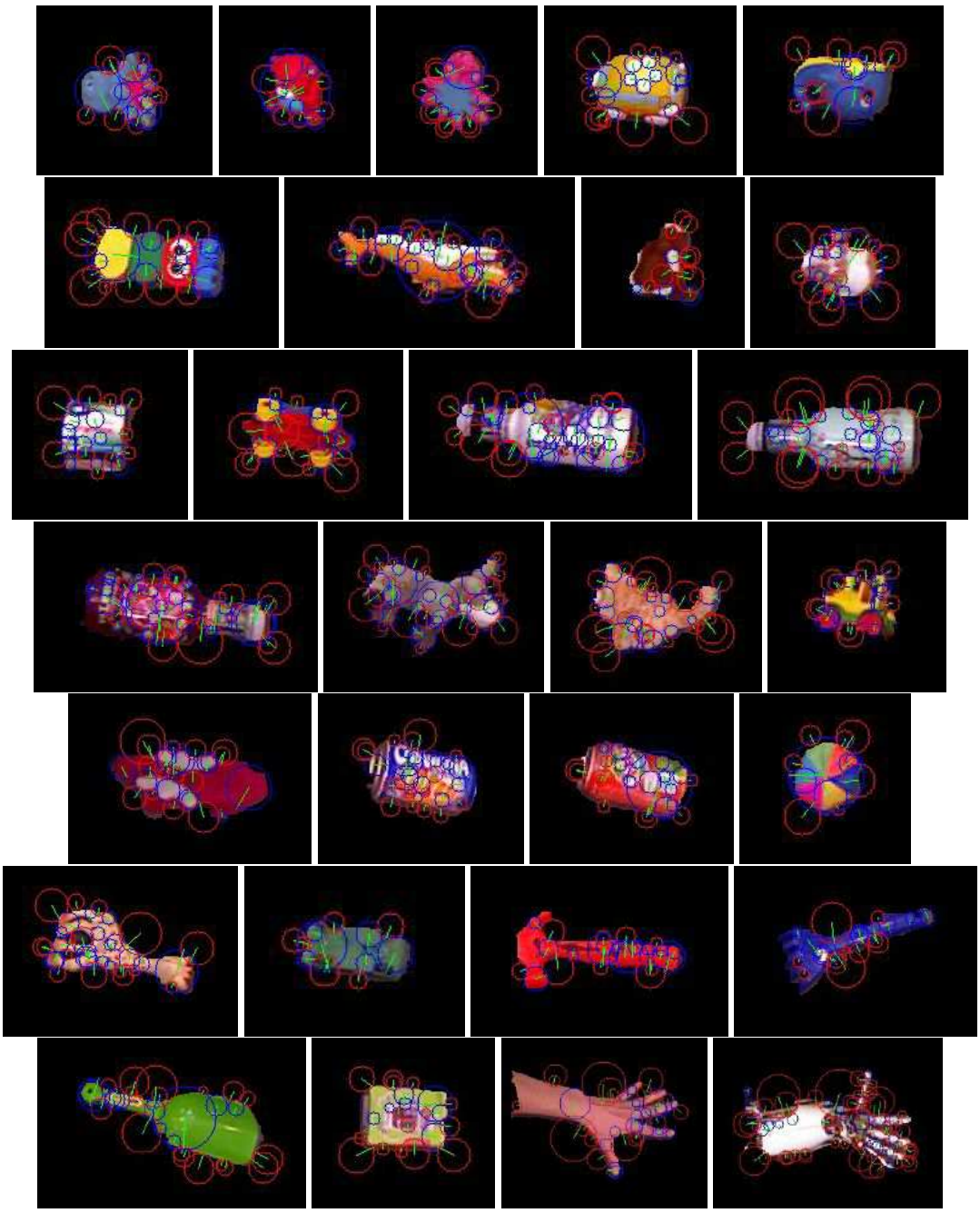


FIGURE 4.14 – Examples of extracted SURF points

Section 3.2. The HSV color space is chosen due to its dimensions that are conceptualized in terms of perceptual attributes in human vision. Moreover, the hue dimension does not change with variation of the light intensity (as happens with RGB values) [Smith, 1978], that makes our superpixel-color feature robust to small changes in illumination.

4.2.2 View representation

The extracted low-level image features are organized into hierarchical representations characterizing the appearances of views of physical entities, as shown in Fig.4.16. Each view characterizes an entity appearance observed from one of perspectives. The view representation is based on the incremental BoW (Bag of Words) approach [Filliat, 2007] extended by an additional feature layer incorporating local visual geometry.

The extracted low-level features are incrementally quantized into dictionaries of visual words based on dissimilarity of features. If the dissimilarity between a feature and each dictionary entry exceeds a specified threshold, a new visual word is added to the dictionary ; otherwise, the feature is assigned to the most similar visual word. The dissimilarity measure between two SURF features is estimated as a histogram difference of their descriptors :

$$\Delta SURF(surf_1, surf_2) = \sum_d (surf_{1_d} - surf_{2_d}), \tag{4.5}$$

where $surf_1$ and $surf_2$ are two compared SURF descriptors, $surf_{1_d}$ and $surf_{2_d}$ are the values from their vectors.

The dissimilarity measure between two superpixel-color features is based on the Euclidean distance between colors in the HSV space :

$$\Delta HSV(hsv_1, hsv_2) = \sqrt{(h_1 - h_2)^2 + (s_1 - s_2)^2 + (v_1 - v_2)^2}, \tag{4.6}$$

where hsv_1 and hsv_2 are two compared superpixel-color features, h_1, s_1, and v_1 are the values of the hsv_1 descriptor, and h_2, s_2, and v_2 are the values of the hsv_2 descriptor.

The quantization procedure provides the SURF and color dictionaries, where each visual word w_f can be represented in the associated feature space as a sphere with a radius equal to the quantization threshold. A fast search procedure [Filliat, 2007] is based on a tree construction using k-means incrementally.

From our experiments which will be reported in Chapter 5, the size of the color dictionary remains relatively stable after processing several objects, since colors repeat among different objects quite often. However, the SURF dictionary grows continuously with the number of objects. In order to avoid rapid growth of the SURF dictionary, we introduce a short- and a long-term memory. The short-term stack contains all extracted SURF features. The features from the short-term stack are filtered according to their co-occurrences over

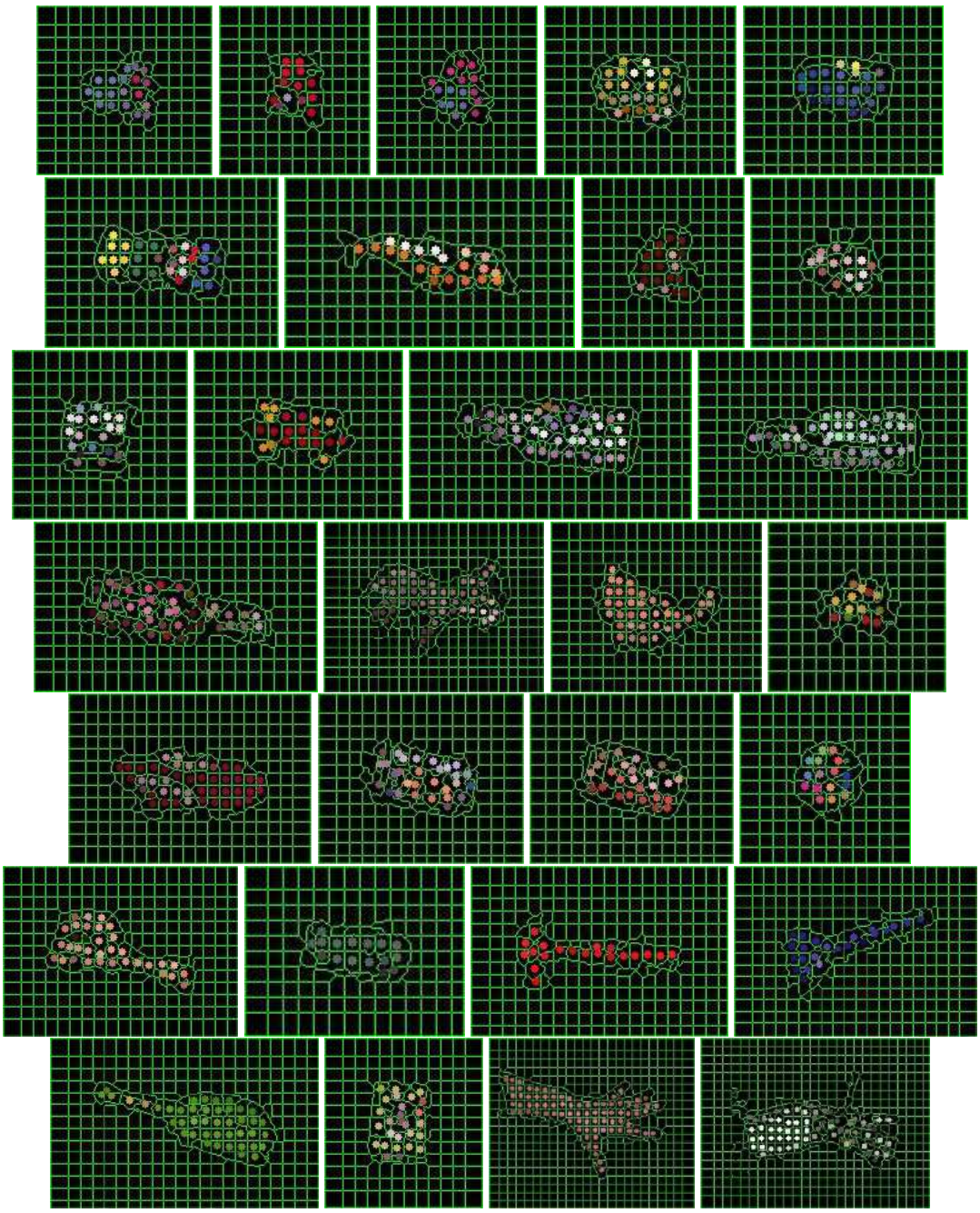


FIGURE 4.15 – Examples of segmented superpixels and their colors

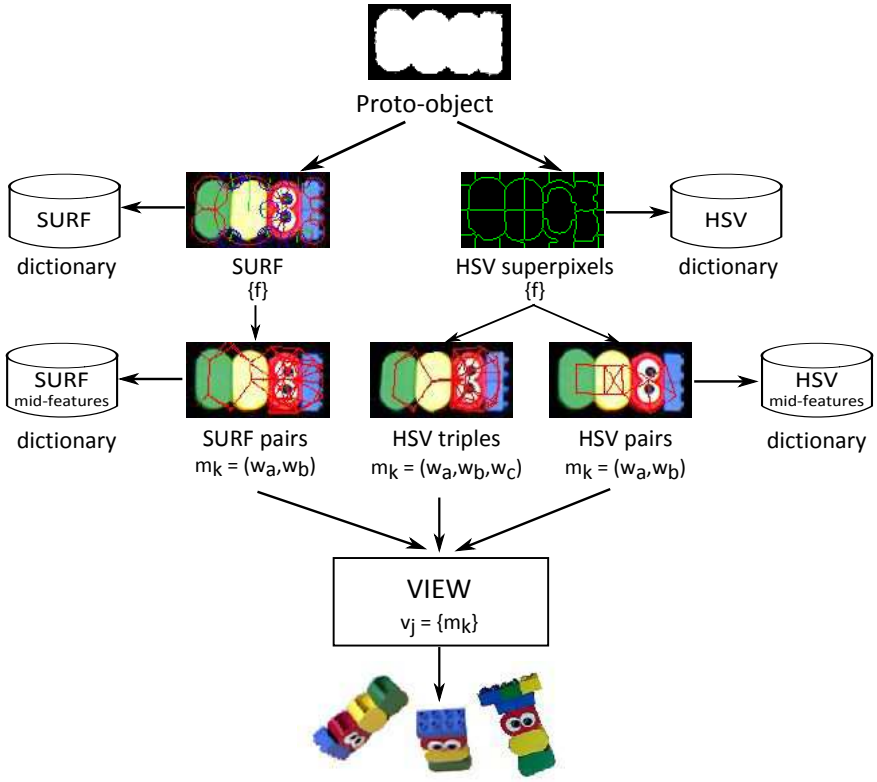


FIGURE 4.16 – View encoding and construction of a hierarchical object model

consecutive frames, and the features seen over several consecutive frames are considered as relevant. The relevant features are stored in the long-term dictionary, that is used in the following processing as a ground level for the entity hierarchical representation.

The low-level features are grouped into a more complex layer of features called mid-features that incorporate local visual geometry. This feature layer allows not only to characterize views by a set of features, like isolated colors or SURF points, but also to synthesize information about features allowing more robust discrimination of objects with same colors or same SURF features. In our algorithm, each low-level feature forms four mid-features with its neighbors that are closest in terms of the Euclidean distance in the image space (see Fig.4.17). Examples of mid-features constructed for different objects are shown in Fig.4.18, 4.19, and 4.20. Thus, each mid-feature m_k combines several low-level features incorporating spatial relation between them. SURF mid-features incorporate relative position of SURF points by grouping closest SURF points into pairs :

$$m_k = (w_a, w_b), \tag{4.7}$$

where m_k is a SURF pair, w_a is one SURF point, and w_b is a neighboring SURF point that is closest to w_a in terms of the Euclidean distance in the image space.

Superpixel-color mid-features incorporate relative position of colors by grouping closest superpixels into pairs and triples :

$$m_k = (w_a, w_b), \tag{4.8}$$

where m_k is a color pair, w_a is one superpixel, w_b is a neighboring superpixel that is closest to w_a in terms of the Euclidean distance in the image space.

$$m_k = (w_a, w_b, w_c), \tag{4.9}$$

where m_k is a color triple, w_a is one superpixel, w_b and w_c are two neighboring superpixels that are closest to w_a in terms of the Euclidean distance in the image space.

SURF pair color pair color triple

FIGURE 4.17 – Examples of mid-features constructed from low-level features (mid-features are shown by the blue color)

Mid-features are incrementally quantized into dictionaries, where each entry corresponds to a pair or a triple of identification numbers (ids) of visual words from the low-level feature dictionaries. The construction of a mid-feature dictionary follows the same concept that was used for quantization of low-level features. If the dissimilarity between a mid-feature and each dictionary entry exceeds a specified threshold, a new mid-feature is added to the dictionary ; otherwise, the mid-feature is assigned to the most similar dictionary entry. The dissimilarity measure between two SURF pairs is estimated as the minimum of two possible pairwise histogram differences of their descriptors :

$$\Delta(m_1, m_2) = min \begin{cases} \Delta SURF(surf_{a_1}, surf_{a_2}) + \Delta SURF(surf_{b_1}, surf_{b_2}), \\ \Delta SURF(surf_{a_1}, surf_{b_2}) + \Delta SURF(surf_{b_1}, surf_{a_2}), \end{cases} \tag{4.10}$$

where m_1 and m_2 are two compared SURF pairs, each pair has two features $surf_a$ and $surf_b$; $\Delta SURF$ is the dissimilarity between two SURF features (in this case, one feature from the first pair and one feature from the second pair) determined in (4.5).

The dissimilarity measure between two superpixel-color pairs is based on the minimal

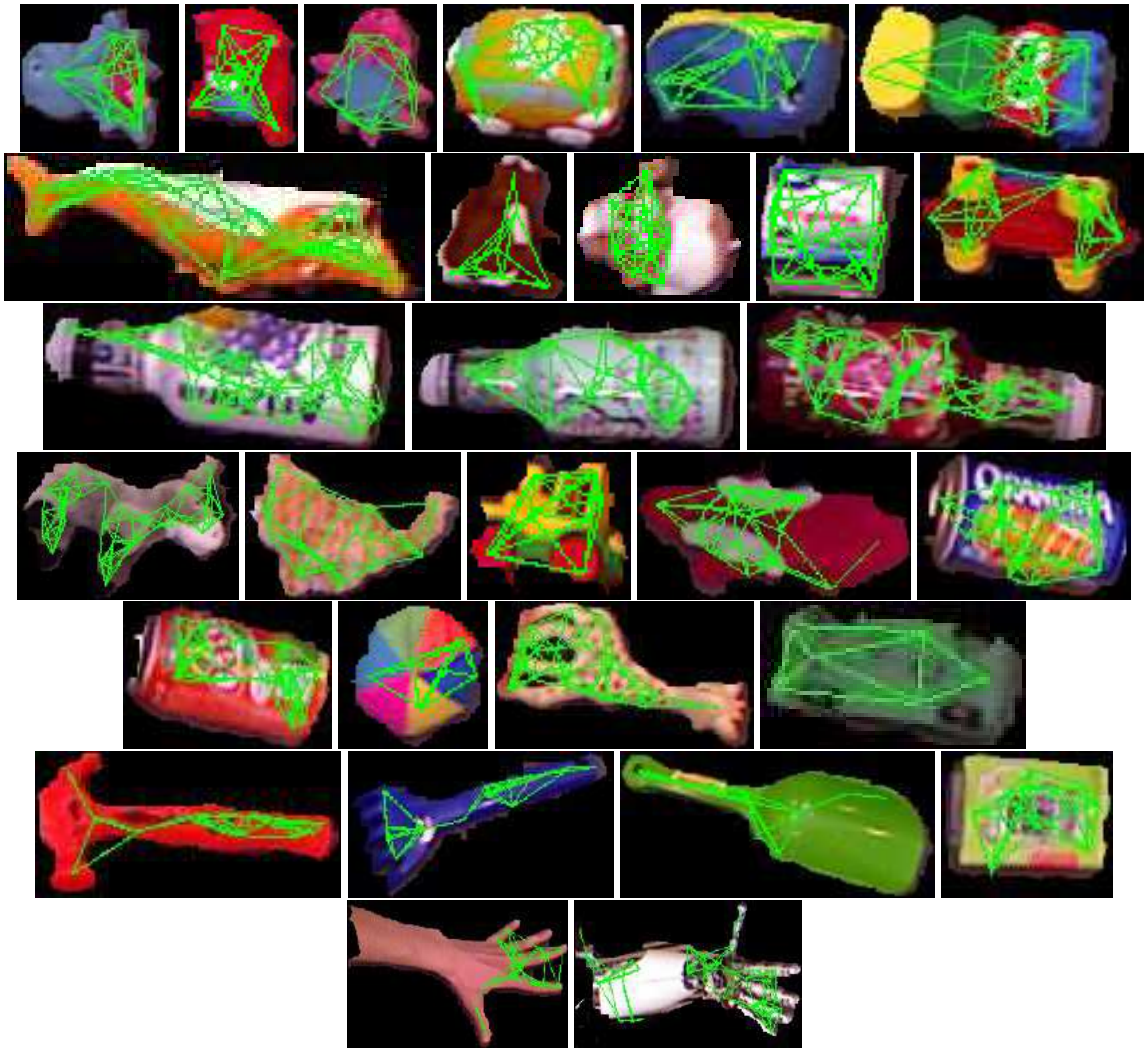


FIGURE 4.18 – Examples of constructed SURF pairs

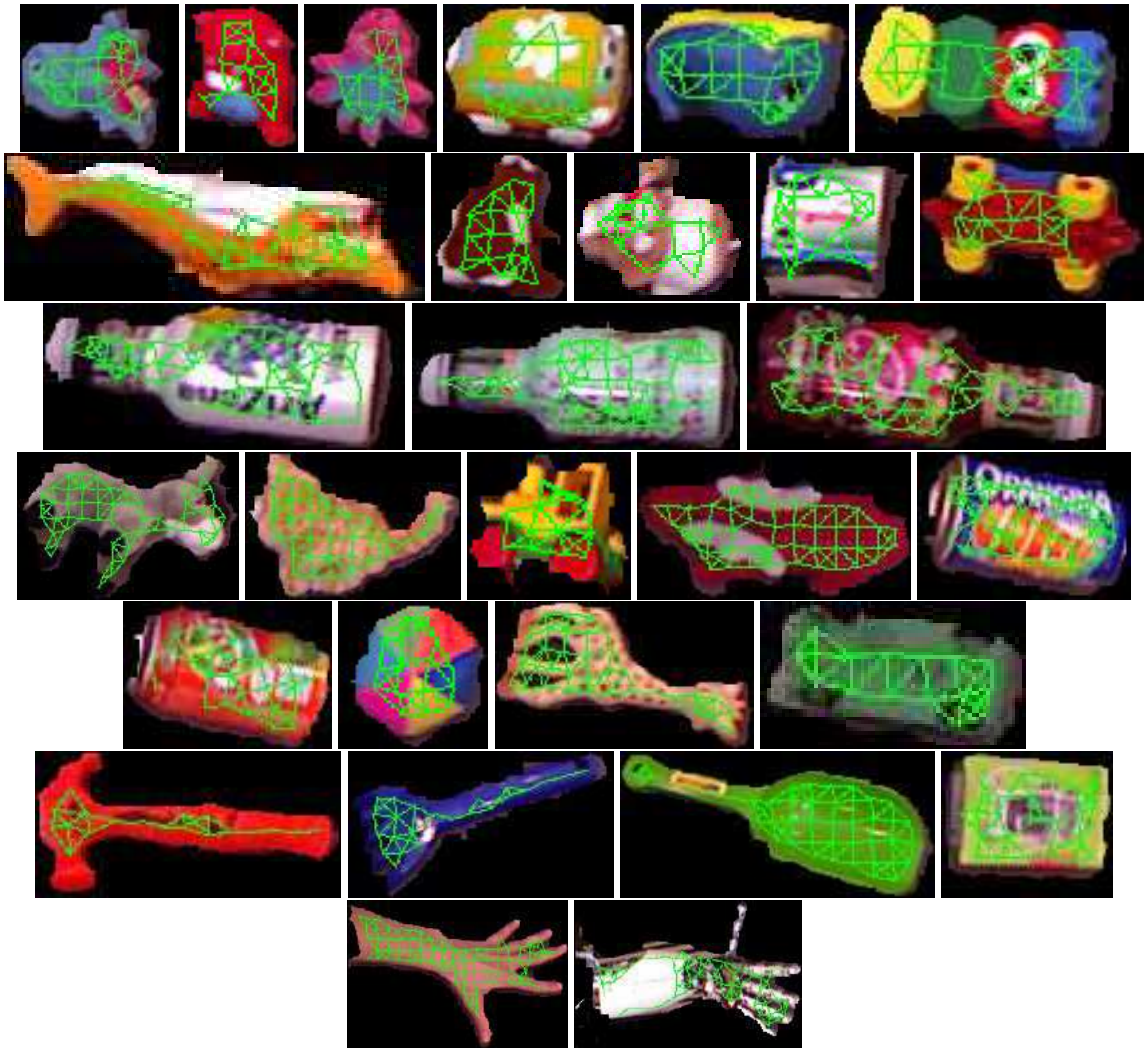


FIGURE 4.19 – Examples of constructed color pairs

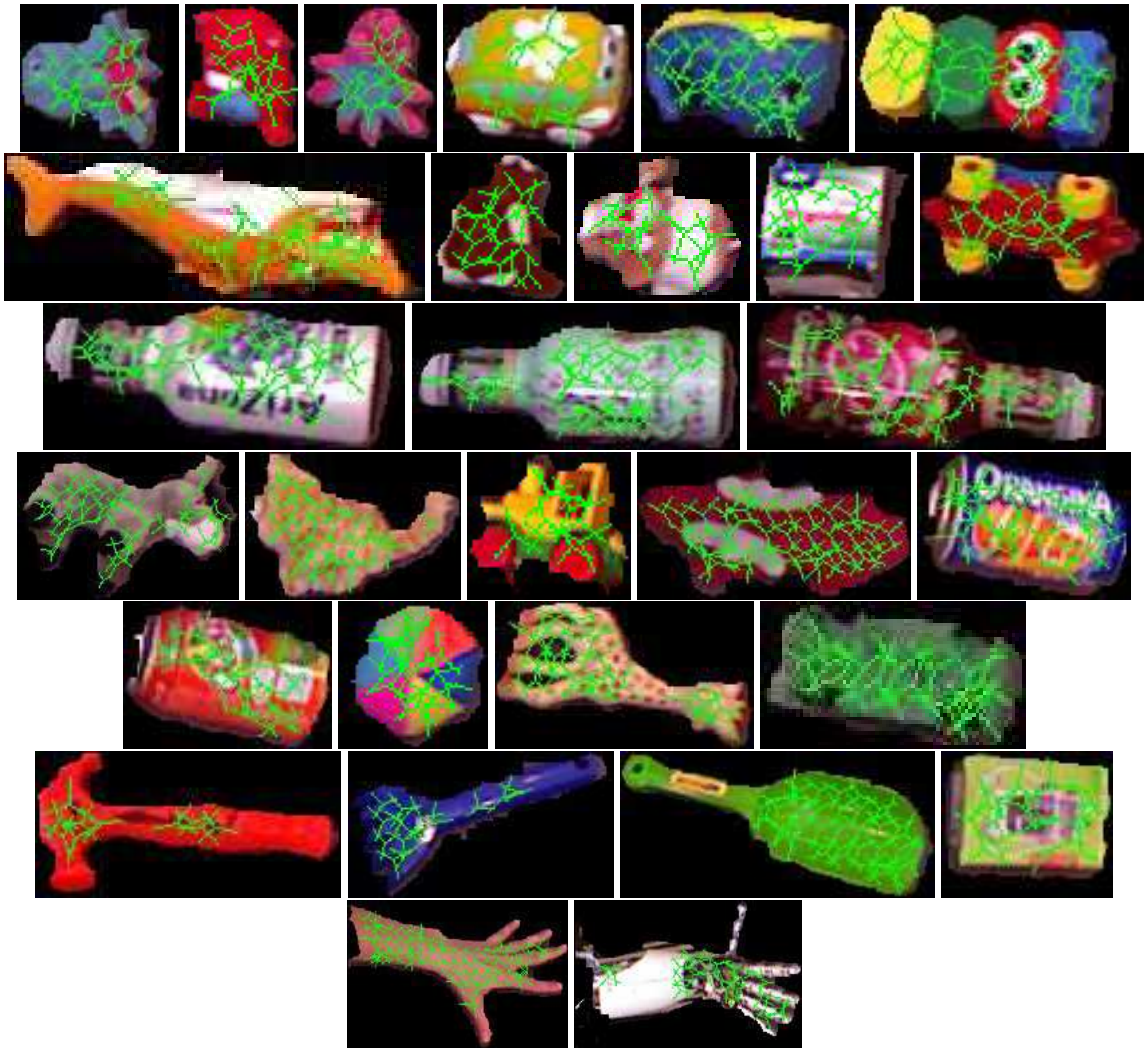


FIGURE 4.20 – Examples of constructed color triples

of two possible pairwise comparison of their descriptors :

$$\Delta(m_1, m_2) = min \begin{cases} \Delta HSV(hsv_{a_1}, hsv_{a_2}) + \Delta HSV(hsv_{b_1}, hsv_{b_2}), \\ \Delta HSV(hsv_{a_1}, hsv_{b_2}) + \Delta HSV(hsv_{b_1}, hsv_{a_2}), \end{cases} \tag{4.11}$$

where m_1 and m_2 are two compared color pairs, each pair has two features hsv_a and hsv_b ; ΔHSV is the dissimilarity between two superpixel-color features (one feature from the first pair and one feature from the second pair) determined in (4.6).

The dissimilarity measure between two color triples is based on the pairwise comparison of their descriptors in a similar way, like in (4.11).

The quantization procedure provides a SURF-pairs, superpixel-color-pairs, and superpixel-color-triples dictionaries. In order to increase the search speed in these dictionaries, the dictionary entries are sorted by the id of the first visual word in each mid-feature.

According to our representation model, all constructed mid-features are used to characterize views, and each view is encoded by the occurrence frequencies of its mid-features :

$$v_j = \{m_k\}, \tag{4.12}$$

where m_k is a mid-feature.

4.2.3 Multi-view representation model

The appearance of a 3D object often varies from different perspectives. In image captured by a visual sensor, a 3D object is perceived as its 2D projection to the scene that was observed from the current position and viewing angle of the sensor, as shown in Fig.4.21. Thus, 2D projections of the same object can be significantly different depending on the object itself (its appearance and shape) and viewing conditions [Hérault, 2010]. There is no direct relation between the degree of the viewing angle and changes in the perceived object's appearance. The change of the viewing angle can result into observation of a completely different appearance (a perspective) or into slightly different appearance, when new details become visible and other details become hidden. In addition, small changes in the perceived object appearance are caused by illumination ; the reflected light sometimes produces shadows and saturations that can make invisible some parts of an object [Goldstein, 2010].

In our approach, the overall appearance of each physical entity is characterized by a multi-view representation model (see Fig.4.22) that covers possible changes in the entity's appearance emerging from different viewing angles and varying illumination. A multi-view representation model is constructed, as described in Section 4.3, and stored in the visual memory of the robot. Each entity is encoded as a collection of views, where each view char-

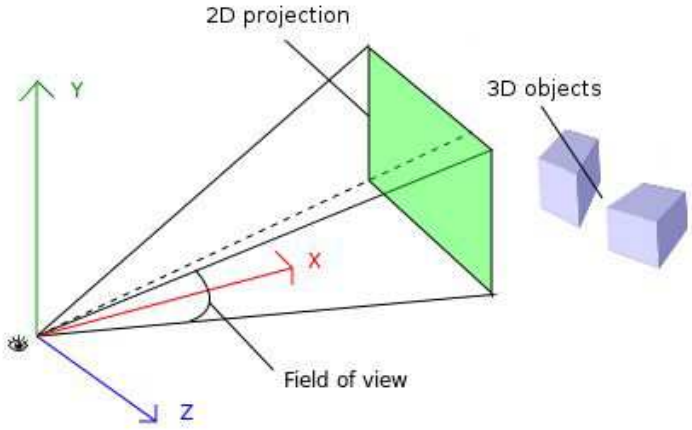


FIGURE 4.21 – The projection of the 3D object into the visible scene

acterizes the appearance of one of entity's representative perspectives :

$$E_i = \{v_j\}, \tag{4.13}$$

where v_j is one of observed views.

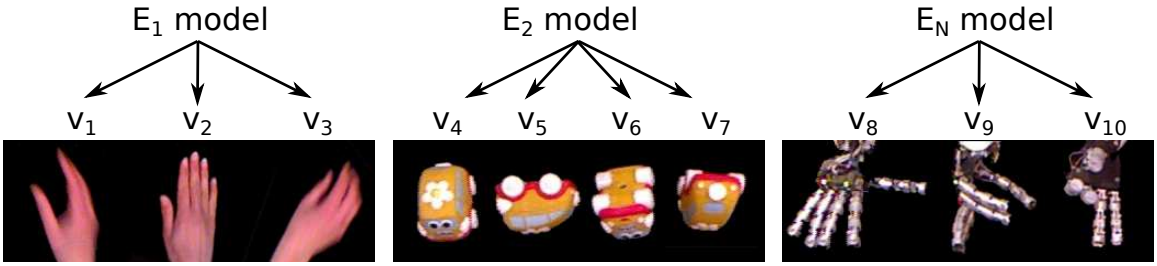


FIGURE 4.22 – Multi-view representation models of three different entities

4.3 Entity learning and recognition

In our study, objects' overall appearances are explored, while the objects are manipulated. In this part of the thesis, the robot learns about objects through observation, while a human partner demonstrates the objects by manipulating them. The robot's perceptual system detect proto-objects in the visual space and characterizes each of them by a set of visual features (as described in Section 4.2), that is learned as a new view or recognized as one of already known views. Then, each identified view is associated with one of physical entities. Since our scenario is based on object manipulation, objects often move together with hands as single proto-objects, thus, we recognize each proto-object either as a single entity or two connected entities.

4.3.1 Learning and recognizing view

Each proto-object detected in the visual space is characterized by extracted low-level features and constructed mid-features. The set of mid-features is associated with one of the views in the robot's visual memory either by recognizing a known view, or by creating a new view. The implemented recognition algorithm is based on a voting method that is used to estimate the likelihood of a mid-feature set (extracted from the segmented proto-object area) being one of the views and on a Bayesian filter that is used to estimates a posteriori probability of each view. The main steps of the recognition and learning procedures are shown in Fig.4.23. The advantage of this approach with respect to supervised algorithms, like SVN or boosting, is the ability to learn new views incrementally, without knowing the number of views in advance and without re-processing all the data while adding a new view.

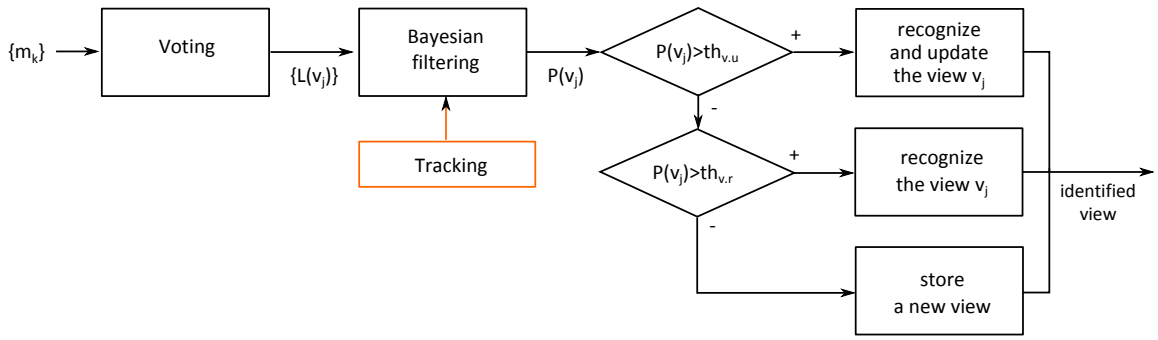


FIGURE 4.23 – The main steps of learning and recognition of views

4.3.1.1 TF-IDF approach

The voting method based on TF-IDF (Term-Frequency - Inverse-Document Frequency) approach was initially used in text retrieval, as described in Section 3.3.1. In image recognition, TF-IDF approach is aimed to evaluate the importance of words with respect to images and to give higher weights to distinctive words. It is often used to represent an image by a set of visual words from a dictionary and the frequencies of these words. In our algorithm, this approach is used to learn and recognize views, where each view is encoded as a vector of mid-features from the dictionary and the frequencies of these mid-features. The statistical measure evaluates the importance of mid-features with respect to known views, and TF-IDF weighting technique gives the priority to distinctive mid-features. The likelihood of a set of mid-features being one of the views is computed as a sum of products of mid-features frequencies and the inverse view frequency :

$$L(v_j) = \sum_{m_k \in v_j} tf(m_k) idf(m_k), \tag{4.14}$$

where $tf(m_k)$ is the frequency of the mid-feature m_k, and $idf(m_k)$ is the inverse view frequency for the mid-feature m_k.

The $tf(m_k)$ accumulates the occurrence of the mid-feature in the view, and it is computed as :

$$tf(m_k) = \frac{n_{m_k v_j}}{n_{v_j}}, \tag{4.15}$$

where $n_{m_k v_j}$ is the number of occurrences of the mid-feature m_k in the view v_j, and n_{v_j} is the total number of mid-features in the view v_j.

The inverse view frequency $idf(m_k)$ is related to the occurrence of a mid-feature among all seen views ; it is used to decrease the weight of mid-features, which are often present in different views, and it is computed as :

$$idf(m_k) = log\frac{N_v}{n_{m_k}}, \tag{4.16}$$

where n_{m_k} is the number of views with the mid-feature m_k, and N_v is the total number of seen views.

4.3.1.2 Voting procedure

During the learning procedure, while the robot observes different objects, we estimate the statistics of mid-features occurrences among views, and the weights of mid-features relevant for each view grow proportionally to the number of occurrences. This estimated statistic is used during the recognition procedure.

For each segmented proto-object, based on the set of its mid-features m_k, we compute the likelihood of recognizing a known view. This likelihood (equation 4.14) of recognizing a known view is computed based on the voting method shown in Fig.4.24. Each mid-feature votes for views where it has been seen before with its $tf - idf$ score. The result of the vote is the likelihood of each view. The voting method is fast, since it uses the inverted index that allows to consider only the views that have at least one common mid-feature with the mid-features of the analyzed proto-object.

4.3.1.3 Bayesian filtering

Views of different real objects can be similar, since one object observed from a certain perspective can resemble another object. The recognition becomes even more difficult, if an object is occluded that often happens during manipulations. In order to deal with these situations, the perceptual system should recognize temporally consistent views, assuming that an object can not change too frequently between two consecutive images. In our approach, the consistency of recognition is achieved by applying a Bayesian filter that improves temporal consistency of view recognition between consecutive images and reduces the potential

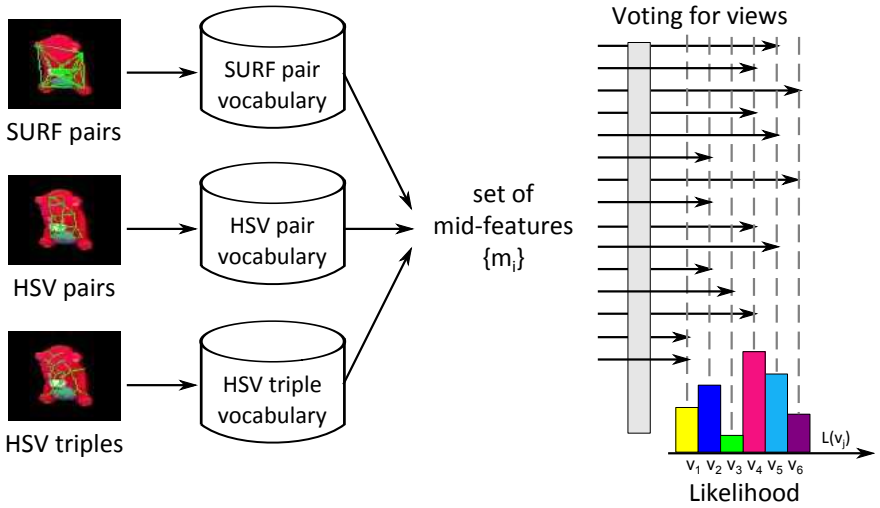


FIGURE 4.24 – The voting method : each mid-feature extracted from the segmented proto-object votes for views where it has been seen before

confusion between entities in a short time scale. Based on tracking, we estimate the probability of recognizing the view from the previous image. The final a posteriori probability of recognizing a view is estimated recursively using its likelihood, its a priori probability computed in the previous image, and the probability of being tracked from the previous image :

$$p_t(v_j) = \eta L(v_j) \sum_l p(v_j|v_l) p_{t-1}(v_l), \tag{4.17}$$

where $L(v_j)$ is the likelihood of recognizing the view v_j, $p_{t-1}(v_l)$ is a priori probability of the view v_l computed in the previous image, $p(v_j|v_l)$ is the probability of tracking the view v_j from the view v_l of the previous image, and η is the normalization term.

The probability $p(v_j|v_l)$ is estimated based on the tracking statistics. All views tracked from the previous image have equal probabilities $p(v_j|v_l)$ among them, and this probability is higher than the probability $p(v_j|v_l)$ of a non-tracked view :

$$p(v_j|v_l) = \begin{cases} \beta, & \text{if the view } v_j \text{ is tracked from the view } v_l \text{ seen in the previous image;} \\ 1-\beta, & \text{otherwise.} \end{cases} \tag{4.18}$$

where β is a coefficient, which we set as 0.8, in order to have also a small probability of recognizing a non-tracked view.

The highest a posteriori probability obtained among all views is compared with several thresholds that are chosen in a way that allows to perform only stable updates among all recognized views and to create new views only in assured cases. Thus, the view with the highest a posteriori probability is

– recognized and updated with a current set of mid-features, if its probability is higher than a threshold $th_{v.u.}$,
– recognized, but not updated, if its probability is higher than a threshold $th_{v.r.}$ with $th_{v.r.} < th_{v.u.}$,
– not recognized, otherwise ; thus, a new view with a current set of mid-features is stored in the visual memory of the robot.

4.3.2 Learning and recognizing entities

The overall appearance of each physical entity is characterized by a multi-view representation model, where views characterize the appearance of representative perspectives. This multi-view model is constructed by tracking the entity between images and accumulating all identified views.

As described in Section 4.1.2, all proto-objects detected in the visual space are tracked between images, and the tracking record stores the associated to them physical entities labels from the visual memory of the robot. If the observed entity is tracked from the previous image, the entity is considered to be the same, and it takes the label of the tracked entity.

If the entity is not tracked from the previous image, its label is identified (as illustrated in Fig.4.25) based on a maximum likelihood approach computed using a voting method similar to the one that is used for recognizing views. In this case, each entity is encoded as a vector of views and the frequencies of their occurrence, and the statistics of the occurrences of views among entities is estimated. The statistical measure evaluates the importance of views with respect to known entities, and TF-IDF weighting technique gives the priority to distinctive views. The likelihood of the current view being one of already known entities is computed as :

$$L(E_i) = \sum_{v_j \in E_i} tf(v_j) idf(v_j), \tag{4.19}$$

where $tf(v_j)$ is the frequency of the view v_j in the entity model, and $idf(v_j)$ is the inverse entity frequency for the view v_j.

The $tf(v_j)$ accumulates the occurrence of the view in the entity model ; it is computed as :

$$tf(v_j) = \frac{n_{v_j E_i}}{n_{E_i}}, \tag{4.20}$$

where $n_{v_j E_i}$ is the number of occurrences of the view v_j in the entity model E_i, and n_{E_i} is the number of views in the entity model E_i.

The inverse entity frequency is related to the occurrence of the view among all seen entities ; it is used to decrease the weight of views, which are present often in models of different entities :

$$idf(v_j) = log \frac{N_E}{n_{v_j}}, \tag{4.21}$$

where n_{v_j} is the number of entities with the view v_j, and N_E is the total number of seen entities.

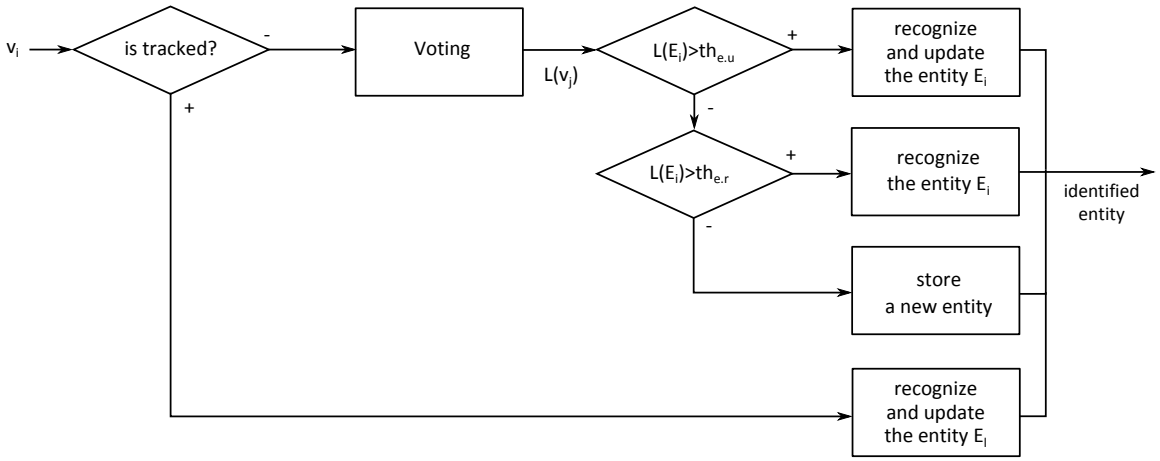


FIGURE 4.25 – The main steps of learning and recognition of entities

The recognition decision is based on the maximal likelihood among all entities ; this likelihood is compared with several thresholds (similar to recognition of views) that are chosen in a way that allows perform only stable updates among all recognized entities and to create new entities only in assured cases. Thus, the entity with the maximal likelihood is

- recognized and updated with a current view, if its likelihood is higher than a threshold $th_{e.u.}$,
- recognized, but not updated, if its likelihood is higher than a threshold $th_{e.r.}$ with $th_{e.r.} < th_{e.u.}$,
- not recognized, otherwise ; thus, a new entity with a current view is stored in the visual memory of the robot.

By identifying physical entities and tracking them in the visual space, their multi-view representation models (like shown in Fig.4.22) are constructed and updated with the views observed while the entity is tracked between images, like shown in Fig.4.26.

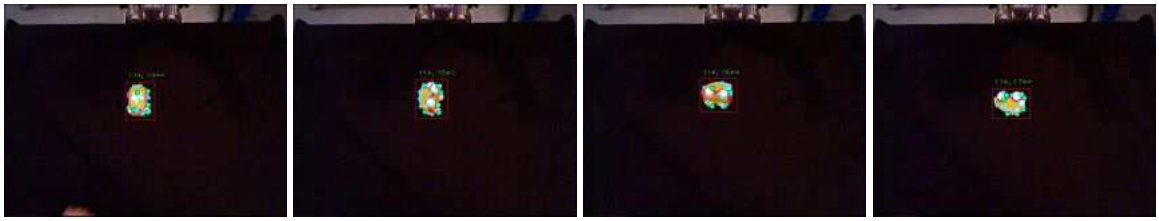


FIGURE 4.26 – The construction of the multi-view representation model : each image shows the tracked entity and its observed view added to the the entity's representation model

4.3.3 Connected entities recognition

In our scenario, objects are explored through manipulation. In this part of the thesis, manipulations are performed only by a human partner, in the following part the robot also manipulates objects. As we have observed during our experiments, any kind of manipulation of objects introduces additional difficulties in processing of the visual data. During manipulations with objects, the human partner's hand or the robot's hand moves simultaneously with the grasped object, and both the object and the hand holding it are detected as a single moving proto-object, like shown in Fig.4.27. Moreover, a hand holding the object produces multiple occlusions and sometimes divides the grasped object into parts. This problem requires an object segregation, as it is called in psychology and described in Section 2.1. Following this idea, our approach segregates connected entities based on the prior experience in terms of the knowledge about already seen entities. Once the robot has seen a human hand or a robot hand moving alone in the visual space, then, the hand can be recognized as one of the connected entities moving simultaneously.

In order to identify connected entities inside a single proto-objects, we use a double-check recognition procedure (summarized in Fig.4.28). The intermediate results obtained during this procedure are demonstrated in Fig.4.29. In the first stage, the most probable view is recognized based on the highest a posteriori probability among already known views, exactly as described in Section 4.3.1. During the second stage, the mid-features that do not belong to the most probable view are used for recognition of another possible view. The connected view is recognized in case of a high recognition probability ($P(v_{j2}) > th_{v.c.}$) and the sufficient number of recognized mid-features. Thus, each segmented proto-object is recognized either as a single view or as two connected views. Each view is associated with one of physical entities base on the recognition algorithm described in Section 4.3.2. Finally, if both the human hand and the manipulated object have been seen already, and the corresponding entities exist in the visual memory of the robot, they can be recognized as connected entities, when the object is grasped.

The ability to recognize connected entities is really important during manipulation of objects. It helps to prevent erroneous updates of views and entities models, while an object is grasped. Since both an object and a hand can be identified as connected entities, the object views are not updated with mid-features of the hand views. Moreover, the information about connected entities is used during the entity categorization presented in Section 7.2 and during the interactive object learning presented in Section 7.3.

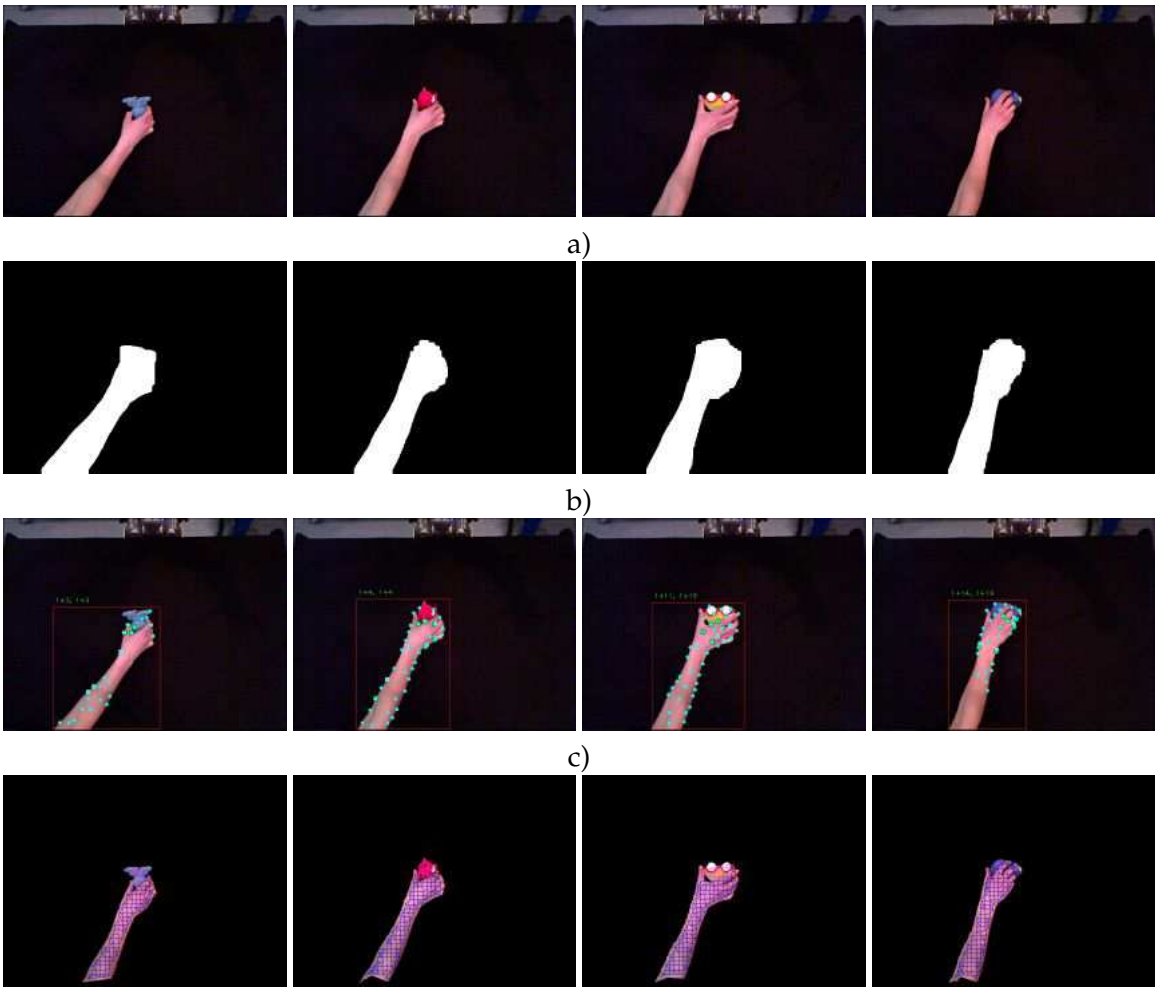


FIGURE 4.27 – Examples of connected entities : a)input images with objects occluded by a human hand on 10%, 25%, 50%, and 75% (from left to right) ; b)detected proto-objects with a human hand and a grasped object moving simultaneously ; c)proto-objects recognized as connected entities ; d)mid-features (in this case, color pairs) of connected entities (the mid-features of the first recognized entity are shown by the magenta color, and mid-features of the connected entity are shown by the blue color

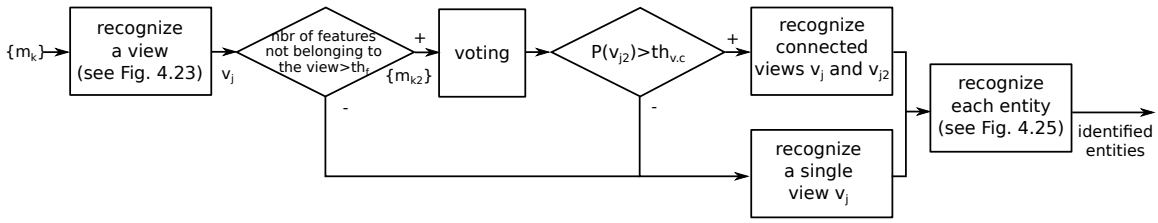
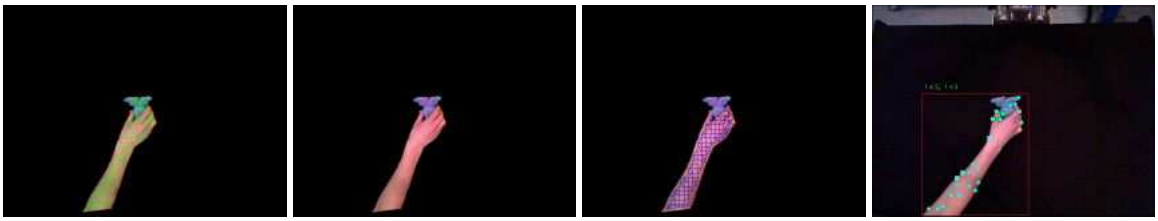


FIGURE 4.28 – The main steps of connected entities recognition



a) b) c) d)

FIGURE 4.29 – Recognition of connected views : a)all extracted mid-features (in this case, color pairs) ; b)the mid-features of the first recognized view, c)the mid-features of the second recognized view, d)the proto-object recognized as connected views ($v_1 + v_3$) and associated entities ($E_1 + E_3$)

4.4 Conclusion

The implemented perceptual system presented in this chapter allows the robot to segment its visual space into regions of interest and to learn or recognized physical entities that can correspond to objects, human hands, or parts of the robot's own body. Without the use of image databases, nor specialized face/skin detectors, the perceptual system acquires all information about the visual scene by continuously extracting low-level features and synthesizing them into hierarchical representation of entities.

The perceptual system uses the concept of proto-objects as units of visual attention signifying about possible physical entities. The appearances of proto-objects are analyzed using complementary features that are chosen to consider maximal information about different visual properties and to characterize well different real-world objects, from simple homogeneous objects to more complex textured objects. The chosen complementary features (SURF and colored superpixels) allow the system to be robust to object texture level, orientation, scale variations, and illumination. The feature vocabularies are constructed incrementally by adding new information when it is available. Another advantage of our approach with respect to simpler Bag of Words approaches is the integration of local visual geometry. Proto-objects are characterized not only as a collection of low-level features, but also by spatial relations between low-level features integrated into mid-level features and further used to encode views. Each view characterizes an entity appearance from one perspective, and an overall entity appearance is characterized by a multi-view representation model incorporating possible appearance changes emerging from different viewing angles, scales, and varying lightning conditions.

The implemented system is based on incremental learning, where new entities and their views are acquired over time and easily added to the visual memory of the robot as soon as they are available. The advantage of the chosen learning method with respect to supervised algorithms (like SVN or boosting) is the ability to learn new entities, without knowing the number of entities in advance and without re-processing all data while adding new data.

Another distinctive property of our system is the ability to distinguish between simultaneously moving connected entities, that is important in the case of manipulation of objects. Moreover the recognition of connected entities is used during the interactive object learning, that will be explored in the second part of this thesis.

CHAPITRE 5

Experimental evaluation of the perceptual approach based on observation of the environment

The implemented perceptual approach is evaluated on the iCub humanoid robot. In this part of the thesis, the robot learns about its close environment through observation, while a human partner interacts with the robot and demonstrates different objects. The experimental setup, the scenario, and the overview of the robotics platform, including its hardware and software architecture, are described in Section 5.1.

Preliminary evaluation of the perceptual system's design choices is presented in Section 5.2, where we compare several object detection methods and several object representation models based on different visual features.

The evaluation of the robot's learning performance is presented in Section 5.3 including the evaluation of the robot's ability to detect physical entities in the visual space and to learn their visual appearances.

5.1 Experimental setup

In our setup, the iCub robot is placed in front of a table, and the visual input is taken from a RGB-D sensor mounted at a distance of 75 cm above the robot's base, like shown in Fig.5.1a. The RGB-D sensor is used instead of the stereo-vision from the robot cameras, since it provides a fastest way to acquire reasonably accurate depth data. However, from a functional point of view, all the experiments could be performed with the embedded stereo-vision.

The chosen position of the sensor allows to see the interactive area in front of the robot. The visual field captures objects localized on the table, some parts of the robot body, and some parts of human partners interacting with the robot. The position of the sensor should

not be closer to the robot, since the sensor's minimum viewing distance (needed for acquisition of depth data) is limited to about 0.45 m, and some robot's actions result in approaching the robot hands closer to the sensor.

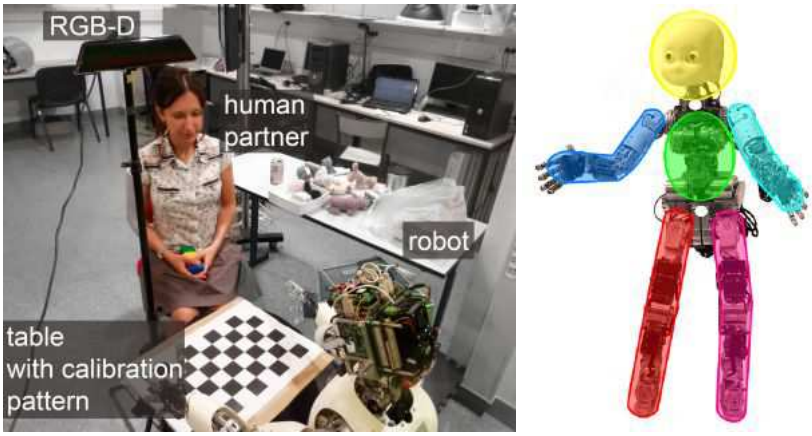


FIGURE 5.1 – The experimental setup and the robot

5.1.1 Description of the robot

The iCub [1] is a humanoid robot developed during the RobotCub [2] European project. The robot is about one meter high with dimensions similar to a child at the age of 3.5 years. The robot is used mainly for research in the domain of developmental robotics, cognition, and artificial intelligence.

5.1.1.1 Hardware

The robot has stereo-vision sensors, sound sensors, and an on-board controller which communicates with sensors and actuators. 53 actuated degrees of freedom allow to move the robot's head, arms, torso, and legs. The robot's motor joints can be controlled through an interface or through communicating commands sent through robot ports described in the following section. The robot's motor joints are organized into the following control groups (see Fig.5.1b) :

- head with 6 main joints,
- two arms with 7 arm joints and 9 finger joints,
- torso with 3 joints,
- two legs with 6 joints per leg.

1. http ://www.icub.org
2. http ://www.robotcub.org

5.1.1.2 YARP middle-ware

The iCub robot is controlled through the YARP [3] platform, which provides the possibility to communicate with the robot and to manage the robotic hardware. YARP allows to build a robot control system as a set of modules communicating between each other and with hardware devices [Metta et al., 2006].

The communication with the robot is achieved by sending commands though robot ports. The acquisition of the robot state is accomplished by reading data from robot ports. Each group of robot motors is controlled through the following ports :

- $/robotName/part/state : o$ used to acquire the information about motors states,
- $/robotName/part/rpc : i$ used for commands that require replies,
- $/robotName/part/command : i$ used for streaming commands.

In our experiments, the robot is controlled through a multi-module architecture developed for curiosity-driven exploration of the environment and described in Section 8.1.3. The implemented perceptual system forms one module of this architecture called *vision*. However in this part of the thesis, the perceptual system can work separately from the robot, since it processes only the visual data that can be acquired from any external sensor or from the robot cameras using the following ports :

- $/robotName/cam/left$ connected to our port $/vision/input\_eyeLeft : i$,
- $/robotName/cam/right$ connected to our port $/vision/input\_eyeRight : i$.

5.1.2 Scenario

In our general scenario, a human partner interacts with the robot in a way similar to an adult interacting with a child. A human partner demonstrates various objects to the robot, like shown in Fig.1.2. Each object is demonstrated through manipulation that allows to observe its different perspectives. In average, each demonstration lasts about one minute and contains 500 images per object. During object demonstration, neither vocal commands, nor other kind of supervision are given to the robot, so the perceptual system decides wherever it observes a new or a known object and associates each real object with as many physical entities and views as needed. The whole set of objects used in our experiments is demonstrated in Fig.5.2.

We design several interactive scenarios, where objects can be demonstrated one by one or several objects at the same time ; objects can be demonstrated by placing them on the table or by holding them in a hand, like shown in Fig.5.3.

3. http ://eris.liralab.it/yarp

toys bottles tools

FIGURE 5.2 – The objects used in our experiments

a) b) c)

FIGURE 5.3 – Examples of scenarios : a) demonstration of a single object, b) demonstration of several objects, c) demonstration of an object by holding it in a hand

5.1.3 Evaluation methodology

Our research investigates the exploration of the robot's close environment including its entities that can be objects, human parts, or parts of the robot's body. If in this part of the thesis, the robot learns through observation, in the next part, the robot learns through interaction, that makes difficult to evaluate the learning performance using existing image databases. Indeed, as learning is incremental and iterative, it is difficult to have a precise evaluation of the robot's performance at a given time. Thus, the robot's performance is evaluated at several stages of the incremental learning process, for example, after each experiment or each image sequence (as shown in Fig.5.4).

Evaluation of unsupervised object learning is a difficult problem. We evaluate our system at two levels :

– detection and tracking rate are estimated based on labeled images,

– recognition rate is estimated by processing a separate image database.

The evaluation database contains 50 images for each object, and each object is shown from different perspectives (example of images are shown in Fig.5.5). While processing images from the database, the outcome from the perceptual system is analyzed in order to estimate the object recognition rate, the number of physical entities and views associated

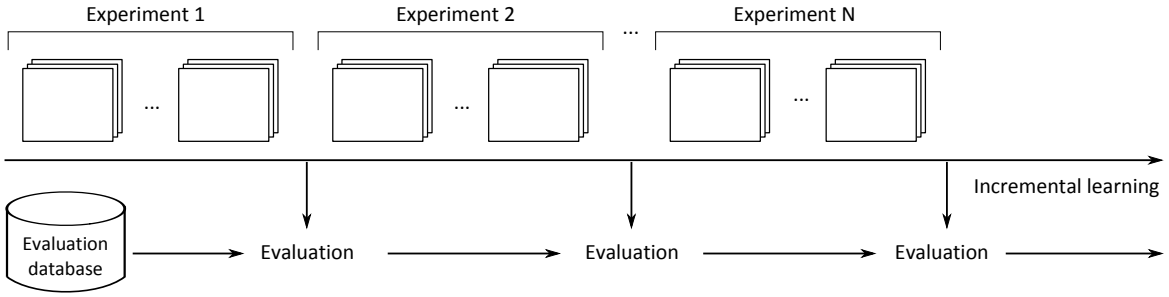


FIGURE 5.4 – The evaluation of the incremental learning process : the system's ability to recognize objects is estimated at several stages of the learning process or after each experiment that can include several blocks, like demonstrations of several objects

with each object.

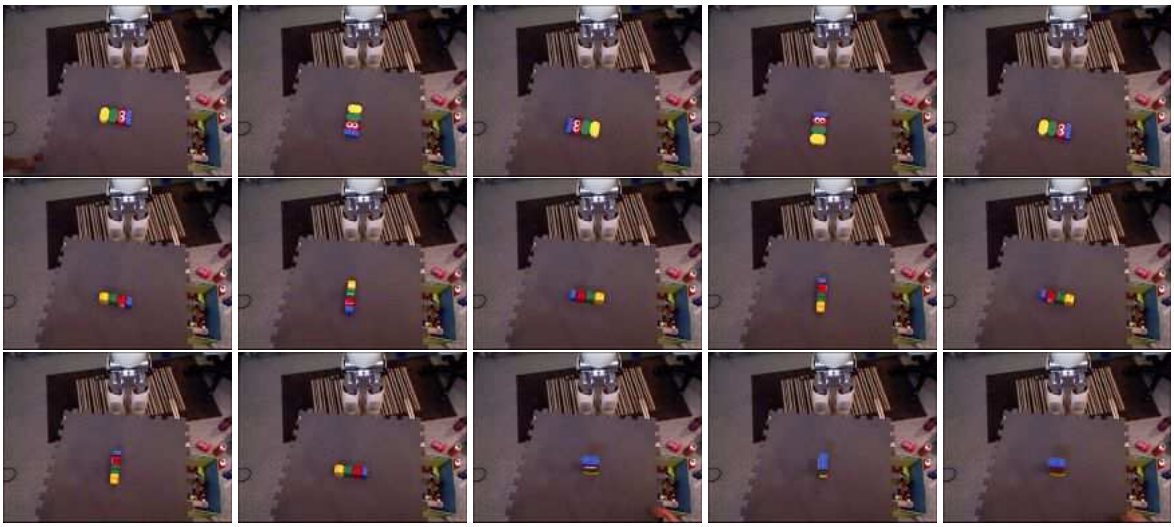


FIGURE 5.5 – Examples of images from the evaluation database

An object is considered to be detected by the perceptual system, when it is segmented as a proto-object at a given position. The object *detection rate* is computed as a percentage of images with properly segmented proto-objects, with respect to the total number of images with the object. The *tracking rate* is estimated as a percentage of tracked proto-objects with respect to the total number of proto-objects.

In order to evaluate the learning performance of the perceptual system, we analyze the accuracy of recognition of previously seen objects. While processing images from the database, we compute the amount of physical entities and views associated with each real object. The recognition of each real object is analyzed based on the following parameters :

- a major label as the id of the physical entity most frequently associated with the object,
- pure labels as the ids of entities associated with this object, but never with other objects,
- noisy labels as the ids of entities associated with several objects.

The object *recognition rate* is computed as a ratio of the number of images with the object recognized as one of its major/pure labels, with respect to the total number of images with the object. The recognition rate is estimate for each object based on two types of labels :

- recognition rate based on a major label, as a percentage of an object's instances assigned to its major label,
- recognition rate based on pure labels, as a percentage of an object's instances assigned to its pure labels.

5.2 Preliminary evaluation of the system's design choices

Different issues on computer vision, such as detecting or learning objects, can be resolved by a variety of image processing methods. During the design of our perceptual system, we search for the best suitable method for each processing stage, and we compare the efficiency of alternative methods. In this section, we compare possible ways of object detection and characterization of object appearance.

Object detection and characterization methods are evaluated based on the experiment, where a human partner demonstrates 12 objects (shown in Fig.5.6) by manipulating each object one by one. In total, the experiment lasts about 12 minutes and contains about 6000 images.

FIGURE 5.6 – The set of 12 objects

5.2.1 Evaluation of object detection methods

In our approach, the detection of proto-objects includes several stages, such as motion processing, isolation of proto-objects based on tracking, and extraction of proto-object boundaries based on depth contours, as described in Section 4.1. As an outcome, each stage provides areas of possible proto-objects, and each area can be used as a mask for proto-object segmentation. Each of the stages produces very different segmentation of proto-objects, and it effects the object learning performance. We compare the final object recognition and tracking rates (shown in Table 5.1) obtained with the following methods of proto-object segmentation :

- segmentation based on moving regions,

– segmentation based on convex hulls around tracked points,
– segmentation based on depth contours.

TABLE 5.1 – The recognition and tracking rates obtained with different proto-objects segmentation methods

Proto-object segmentation method	Recognition rate,% based on pure labels	Recognition rate,% based on a major label	Tracking rate,%
Motion-based	86.5	52.2	46.2
Tracked points-based	82.3	51.5	36.8
Depth contours-based	98.2	55.7	75.8

From our experiment, the best results are obtained, when proto-objects are segmented based on depth contours. Proto-object segmentation based on both moving regions and tracked points results in a smaller recognition and tracking rates, and produces much more entities that correspond to parts of the background. Proto-object boundaries based on moving regions are often shifted from real object boundaries, especially, in case of fast object motion. However, boundaries based on moving regions provide better results than boundaries based on tracked points, since the convex hulls around tracked points often cut parts of objects and include parts of the background. The proto-object segmentation based on depth contours outperforms other segmentation methods. The difference in results between methods are not very large, however, in case of a more complex background, depth-based segmentation should further outperform other segmentation methods, because they would include more noisy features from the background.

5.2.2 Evaluation of object representation model

In our approach, objects appearances are characterized by multi-view representation models. As the basis of each representation model, we use low-level features. While choosing a set of complementary low-level features, we search for features that consider different visual characteristics, as described in Section 4.2.1. Eventually, we use SURF points, which are adapted to textured objects, and colored superpixels, which are adapted to homogeneous objects.

In order to incorporate local visual geometry, low-level features are grouped into mid-features, as described in Section 4.2.2. Mid-features can be constructed in different ways. The number of low-level features grouped into one mid-feature effects the informativeness of mid-features. Since mid-features are used to encode view representations, they also effect the object learning performance. We evaluate our system with view representations based on mid-features constructed as pairs and triples of closest low-level features, and the obtained average object recognition rate is reported in Table 5.2.

TABLE 5.2 – The recognition rate obtained with different mid-features

Features used in a view representation	Recognition rate,% based on pure labels	Recognition rate,% based on a major label
SURF and HSV (low-level)	84.3	40.7
SURF pairs and HSV pairs	97.8	48.5
SURF pairs and HSV triples	88.5	46.8
SURF pairs, HSV pairs, and HSV triples	98.2	55.7

From our experiments, the best recognition rate is obtained, when the view representation is based on three types of mid-features, such as SURF pairs, HSV pairs, and HSV triples. Thus, we keep this set of mid-features as a basis for representation of entities views.

According to our algorithm, each low-level feature forms n mid-features with its neighbors. We evaluate our system with several n values, since this parameter effects the size of a feature set characterizing each view and thus, it effects the informativeness of the view representation and finally, the object learning performance. The average object recognition rate obtained with n equal to 2, 3, and 4 mid-features, is shown in Table 5.3.

TABLE 5.3 – The recognition rate obtained with different view representations

Features used in a view representation	Recognition rate,% based on pure labels	Recognition rate,% based on a major label
4 SURF pairs and 2 HSV pairs	96.7	48.8
4 SURF pairs and 3 HSV pairs	97.5	53.2
4 SURF pairs and 4 HSV pairs	97.8	55.3
4 SURF pairs and 2 HSV triples	84.8	45.7
4 SURF pairs and 3 HSV triples	89.8	48.0
4 SURF pairs and 4 HSV triples	87.7	50.2
4 SURF pairs, 4 HSV pairs, 4 HSV triples	98.2	55.7

From our experiments, the best recognition rate is obtained, when each color feature forms 4 mid-features, both pairs and triples. Thus, we keep this configuration for all features and construct 4 mid-features for each low-level feature. However, as the results are quite close, depending on the set of objects and the quality of images, this choice could be reconsidered in future work.

5.2.3 Recognition of connected objects

In all our experiments, the robot learns objects, while a human partner demonstrates objects by manipulating them. In this scenario, each object is often grasped by a human

partner and compose a single moving proto-object with a hand ; this proto-object is identified by the perceptual system as connected entities, as described in Section 4.3.3.

Various manipulations with objects results in a different amount of an object occlusion by a human hand. We evaluate the system's ability to recognize objects occluded by a hand on 10, 25, 50, and 75%. The evaluation is based on the pre-recorded sequence of images, where a human partner manipulates 10 objects (shown in Fig.5.7). In this experiment, we use a smaller set of objects, and we estimate the recognition and the detection rates based on labeled images. The obtained detection and recognition rate is reported in Table 5.4.

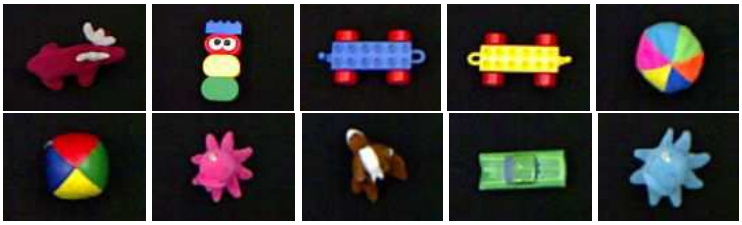


FIGURE 5.7 – The set of 10 objects

TABLE 5.4 – The impact of occlusion on the object detection and recognition rates

Amount of occlusion, %	Detection rate, % connected[+single] entities	Recognition rate, % connected[+single] entities	Total detection/recognition rate,%
10	54 [+25]	38 [+2]	79/40
25	63 [+7]	32	70/32
50	47 [+6]	18	53/18
75	38	0	38/0

The system is able to recognize objects occluded by a human hand up to 75%. The increasing amount of occlusion decreases the recognition rate, but it is not problematic for us, since our goal is not a recognition of objects in each single image, but rather continuous learning of coherent object models using the available information about connected entities.

5.3 The performance of object learning

After having evaluated several parameters with a restricted set of objects, we now turn to the evaluation of the complete system with the chosen parameters on a larger set of objects. In this experiment, the robot learns about its close environment through observation, while a human partner interacts with the robot and demonstrates up to 20 objects shown in Fig.5.8. In average, the experiment lasts about 20 minutes and contains about 10000 images.

The implemented perceptual system is evaluated over the following characteristics : the

FIGURE 5.8 – The set of 20 objects

ability to detect objects in the visual space and the ability to learn their appearances in order to recognize them later.

5.3.1 Evaluation of objects detection

Detection of objects is evaluated on a pre-recorded sequence of images, where a human partner demonstrates 20 objects (shown in Fig. 5.8) by manipulating each object one by one. The detection and tracking rates have been estimated based on labeled images. The perceptual system shows an average detection rate of 98% and a tracking rate of 77%.

The detection rate is influenced by each of the stages of proto-object detection, such as motion processing, isolation of proto-objects based on tracked points, and extraction of depth contours, described in Section 4.1. Here, we analyze the percentage of proto-objects isolated based on tracking points with respect to the percentage of proto-objects isolated by processing depth contours. The isolation of proto-objects based on tracking points, described in Section 4.1.2, has allowed to detect about 97.8% of proto-objects. The other 0.2% of proto-objects have not been isolated based on tracking that can occur, when several proto-objects are static and localized near to each other ; however, these proto-objects have been isolated properly based on depth contours. Thus, the processing of depth contours allows not only to precise boundaries of moving proto-objects but also to isolate static proto-objects localized near to each other.

5.3.2 Evaluation of objects learning

The robot's learning performance is evaluated on the pre-recorded sequence of images with a human partner demonstrating 20 objects, as in the previous section. Using the database (as described in Section 5.1.3, but with 20 objects), we estimate the robot's ability to recognize already seen objects. The recognition rate is computed for each object based on its major and pure labels and reported in the Table 5.5.

The average recognition rate based on pure labels is about 80%-90%, and it differs be-

TABLE 5.5 – The results obtained by learning through observation : the recognition rate, and the number of pure entities and views associated with each object

Object	Recognition rate based on pure labels,%	Recognition rate based on a major label,%	Number of associated pure entities	Number of views in a major label	Number of associated pure views
O_1	96	33	6	2	9
O_2	90	78	3	3	6
O_3	96	40	6	1	6
O_4	60	44	3	2	4
O_5	41	41	1	2	2
O_6	63	40	7	1	7
O_7	60	52	2	1	2
O_8	100	50	4	1	4
O_9	96	32	8	1	9
O_{10}	80	22	8	1	8
O_{11}	84	23	6	1	6
O_{12}	87	47	4	1	4
O_{13}	100	97	2	2	2
O_{14}	87	38	7	1	7
O_{15}	90	25	5	1	5
O_{16}	100	100	1	1	1
O_{17}	100	80	2	2	2
O_{18}	100	99	2	1	2
O_{19}	100	99	1	2	2
O_{20}	83	76	4	1	4
Mean	85.7	55.8	4.1	1.4	4.6

tween objects. The percentage of objects recognized by major labels with respect to objects recognized by pure labels is shown in Fig.5.9. Intuitively, objects with different shapes and colors have been recognized better than objects that are similar between each other from some of perspectives. From the confusion matrix shown in Fig.5.10, the maximal confusion has occurred for the object *lego-car* O_{11}, that was confused with the objects *lego-toy* O_6 and *red bear* O_2. These three objects have similar colors, and O_{11} has similar SURF points with the object O_6 (these points are localized on lego blocks with identical features). However, two identical objects *octopus* O_1 and O_3 that differ only by color, have been distinguished rather well between each other.

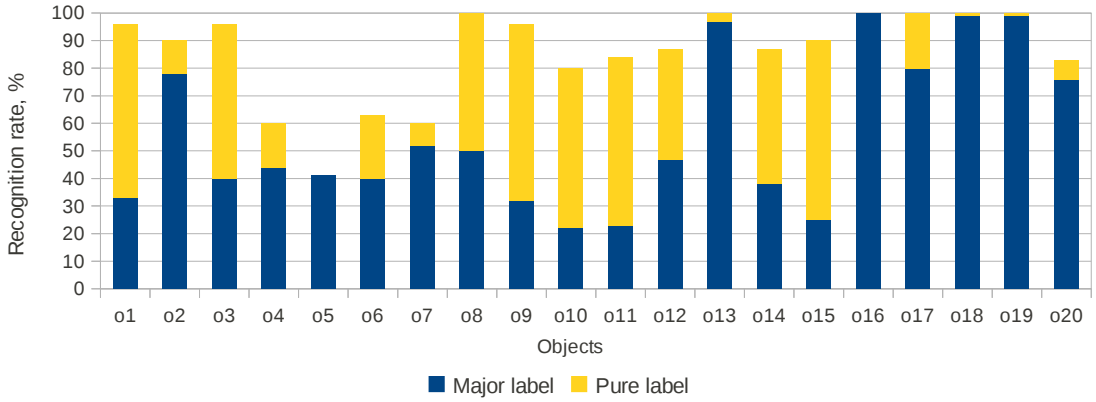


FIGURE 5.9 – The object recognition rate based on major labels (shown by the blue color) with respect to the recognition rate based on pure labels (shown by the yellow color)

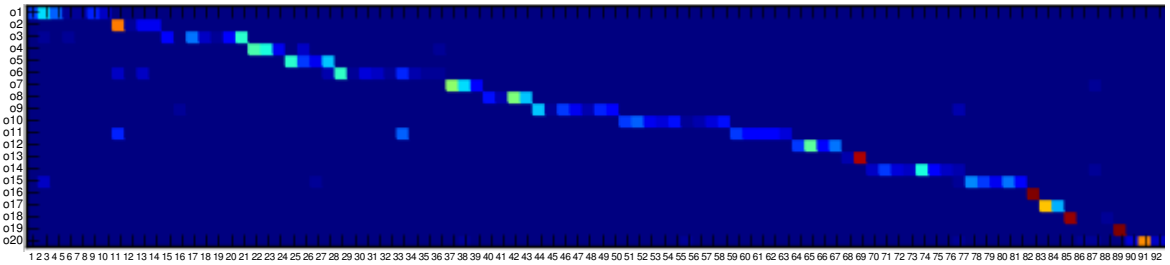


FIGURE 5.10 – The confusion matrix, where objects are shown in lines, and the associated physical entities shown in columns ; the color range (from blue to red) represents the percentage of objects instances associated with each entity, where the blue color corresponds to 0%, and the red color corresponds to 100%

Examples of created representation models that correspond to major objects labels are shown in Fig.5.11. At this stage, when the object learning is based on observation only, the representation models contain just few views ; however, we expect increasing of the number of views, if objects would be manipulated for a longer time.

Some objects have been associated with several physical entities (examples of major and

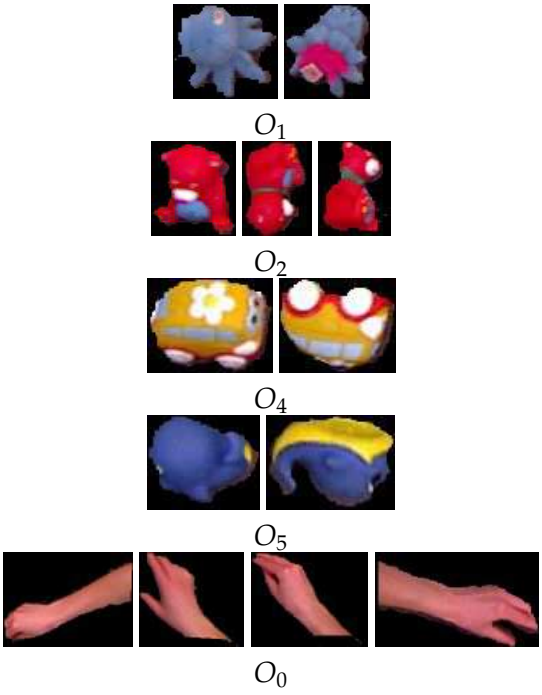


FIGURE 5.11 – Examples of representation models of the major entities that correspond to the objects O_1, O_2, O_4, O_5, and O_0 (each model with its views is illustrated in one line)

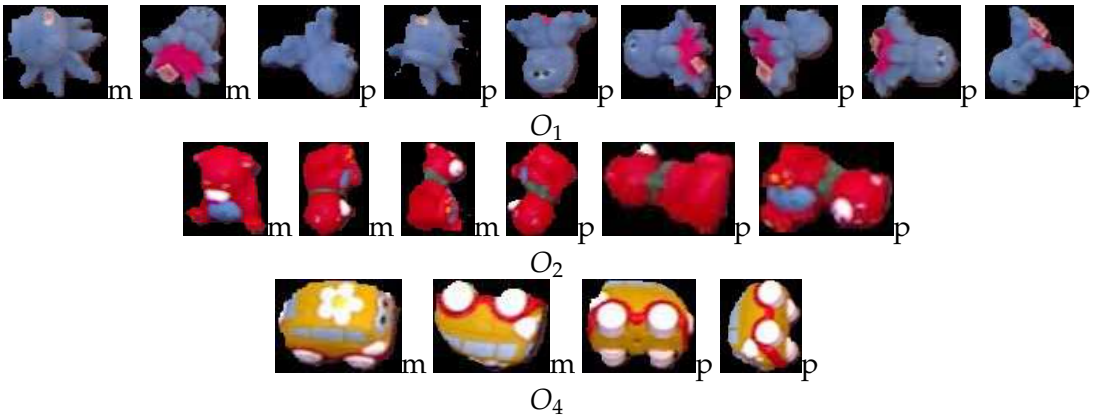


FIGURE 5.12 – Examples of major and pure views of the objects O_1, O_2, and O_4 (major views are indicated by *m* letter, and pure views are indicated by *p* letter)

pure views are shown in Fig.5.12), that occur, when a human partner hides an object out of the view of the visual sensor and demonstrates another object perspective that makes impossible to track the object and therefore, impossible to learn it as a single physical entity. Multiple physical entities created for a single real object are not ideal, but are already an interesting stage, since in this case, the object learning is based only on observation that is just one of aspects of object learning in humans. For example, if an object can not be recognized from a given perspective, humans usually turn the object in order to see one of its representative perspective that allows to recognize the object. Exploration of objects in infants is performed through continuous physical interaction and communication, as discussed in Chapter 2, and as will be implemented in Part II of this thesis.

5.3.2.1 The influence of the number of learning objects

In our scenario, the total number of learning objects is not fixed. Since the learning is incremental, new objects can be added continuously. We analyze the effect of increasing the number of learning objects and its influence to the system's ability to recognize already seen objects. In this experiment, we show 20 objects one by one to the robot, and we evaluate the system's recognition rate after showing each new object. The obtained average recognition rate based on both labels is shown in Fig.5.13.

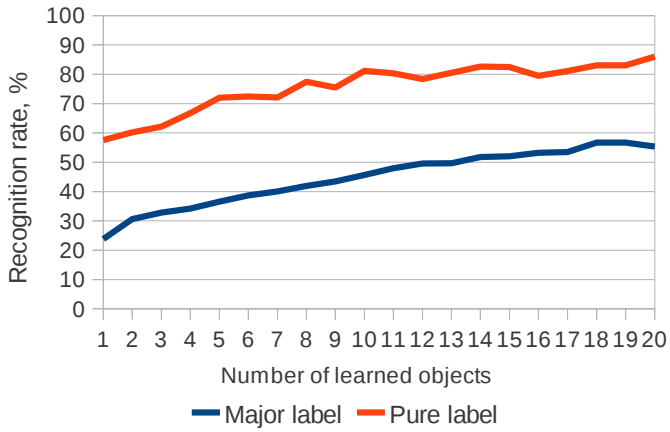


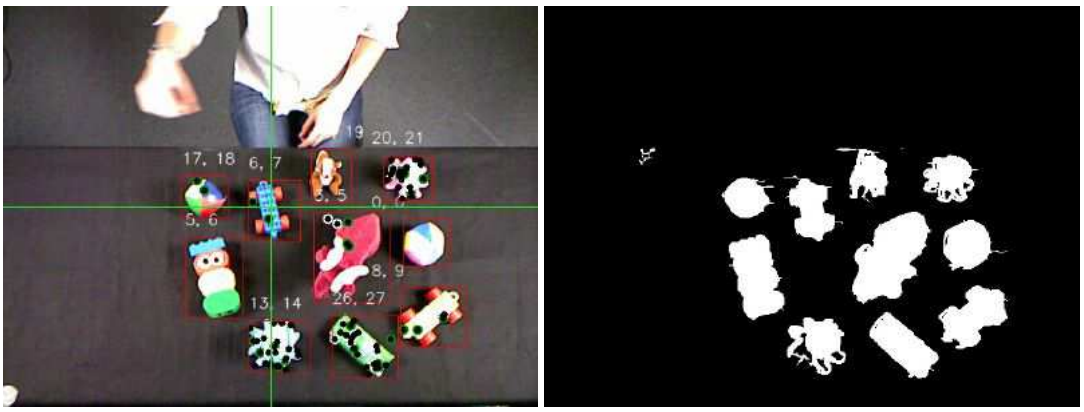
FIGURE 5.13 – The influence of the number of objects to the average recognition rate

Surprisingly, the recognition rate based on both labels grows, when the number of learning objects increases, and after learning approximately 8 objects, the recognition rate based on pure label remains nearly stable. This effect can be explained by the confusion of objects with the human hand. Since in our experiments, all objects are manipulated by the human, each object is often seen together with the human hand, and thus, objects are confused not only between each other but also with the human hand. However, continuous manipula-

tion of different objects increases the dissimilarity of the human hand from the objects, that finally leads to the improvement of the average recognition rate.

5.3.2.2 Simultaneous processing of several objects

We also analyze the system's ability to process multiple objects presented at the same time in the visual field. The system has been tested with up to 10 objects presented at the same time in the visual field, like shown the Fig. 5.14, and all of objects has been detected and recognized.



a) b)

FIGURE 5.14 – Simultaneous processing of several objects : a) 10 objects presented in the visual field, b) the corresponding segmentation of the visual space

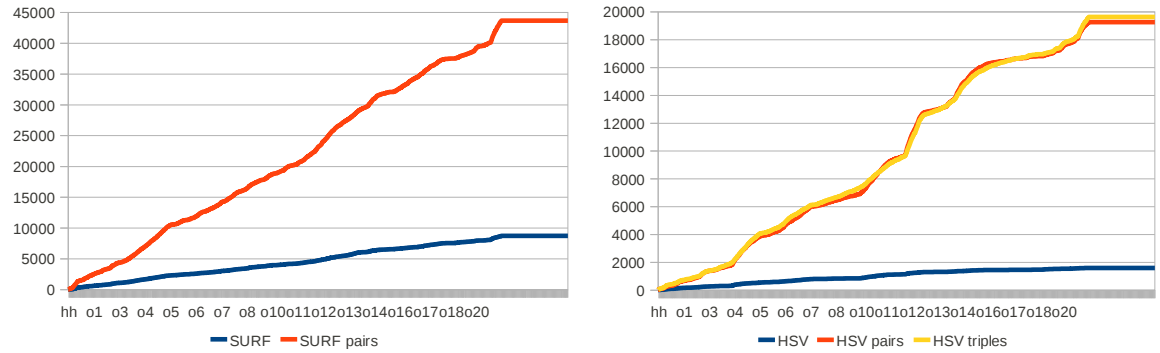
5.3.2.3 Dictionaries growth

While learning objects, all gathered information about their appearances is stored in dictionaries in the visual memory. Thus, the dictionaries grow with the increasing number of observed objects. The growth of dictionaries over time is shown in Fig.5.15. Once the feature dictionaries have reached a certain amount of data, they grow slower, since the visual features can repeat between objects.

In order to enable longer term experiments, dictionaries should be cleaned over time to suppress insignificant data and to keep only meaningful data that are used often. In this work, we clean the dictionaries of views and entities during active learning, as will be described in Section 7.3. In future work, we plan to clean all dictionaries over time with the goal of eventually stabilizing their size and the associated computation time.

5.3.2.4 Processing time

We analyze the processing cost of the proposed perceptual system and its main stages, while the robot learns 20 objects. During incremental learning, the average processing time



a) b)

FIGURE 5.15 – The growth of the dictionaries : a) SURF and SURF pairs, b) HSV colors, HSV pairs, and HSV triples

was about 0.13 sec for images with one detected object. The presence of several objects in the visual field tends to increase computation time. The distribution of the processing cost between different stages of the proposed perceptual system is shown in Fig.5.16. The highest computation cost belongs to the recognition/learning of views, in particularly to the search of features in dictionaries. The evolution of processing time over time is shown in Fig.5.17. In addition to the total processing time, we also show the evolution of its most expensive component with respect to all other processing stages.

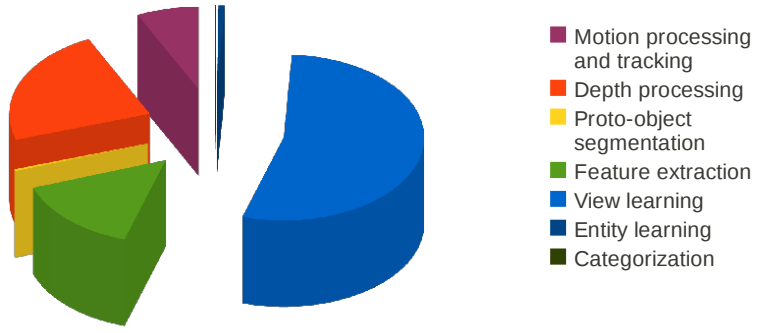


FIGURE 5.16 – The distribution of the processing time between the main stages of the perceptual system

From our experiments, the time required to process an object varies significantly between objects depending on their complexity or the number of their features. Moreover, the cost of view recognition/learning increases with the dictionaries growth. Other processing stages, like proto-object detection, segmentation, tracking, feature extraction, and categorization, take all together about 0.06 sec per image, and their processing cost stays relatively stable over time.

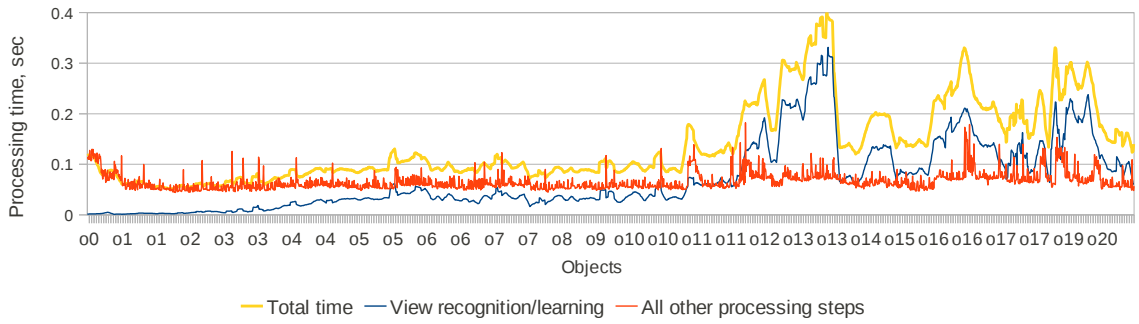


FIGURE 5.17 – The evolution of processing time, where each value corresponds to the time (in seconds) took to process one image with at least one object : the total processing time is shown by the yellow color, the time taken by view recognition/learning is shown by the blue color, and the time taken by other processing stages except view recognition/learning is shown by the orange color

5.4 Conclusion

In this chapter, we have grounded the design of the proposed perceptual system on experimental evaluation of possible ways of object detection and characterization. Further, we have evaluated the system's learning performance. The system has shown an appropriate entity detection and recognition rate, while learning 20 objects from demonstration of a human partner. The system was able to detect and recognize not only objects presented alone, but also objects manipulated by a human partner. Objects occluded up to 75% and moved together with a human hand have been recognized as connected entities. The recognition of connected entities enables the system to be robust to occlusion, and it prevents erroneous updates of objects models with views of connected entities.

The perceptual system was able to learn different types of objects from simple objects with few colors to more complex textured objects. The system has shown to be invariant to object rotation, viewing angle, and small illumination changes. Various manipulations with objects show, that the system is robust to motion artifacts, and the experiments with complex background show that the system performs well in case of a visual clutter.

Deuxième partie

Development of active perception approach

CHAPITRE 6

State of the art : interactive perception

Interactive perception is a integral approach towards autonomous learning about the environment. Interactive perception integrates multiple capabilities, like perception, control, learning, and planning. Interactive exploration is one of the most informative ways of learning about the environment, as we have learned from the infants development in Section 2.2.

In interactive exploration of the environment, the knowledge about own body provides a great capacity. For example, infants start to learn about the world from the development of a sense of own body, and only later perform interactive actions directed to exploration of the environment, as discusses in Section 2.2. Self-identification enables to acquire the spatial understanding of the surrounding environment with respect to the position of own body. This spatial perception is one of high-level cognitive functions, that allows to achieve needed effects of own actions and to perform complex actions requiring planning. Different concepts of self-identification are studied in Section 6.1.

The identification of the robot's own body in the visual space allows to enhance interactive perception and to improve the efficiently of interactive exploration of the environment. The robot's interactive actions are used to learn about objects, their appearance, and other properties, and also to understand the objecthood, like if an object is graspable or not. The overview of research studies on interactive perception is presented in Section 6.2, where we describe approaches aimed on object detection, segmentation, and learning object properties.

The discussion about advantages and limitations of reviewed approaches that led to our choice of the self-identification method and to design of interactive perceptual approach, are provided in Section 6.3.

6.1 Robot self-discovery

Among the variety of robot self-discovery methods, most algorithms are based on a prior knowledge or local approaches. Some strategies exploit a predefined pattern of robot's motion, a predefined appearance of the robot body, or body kinematics, such as joint-link struc-

ture. For example, [Hulse et al., 2009] detects the position of a robot hand holding an object of a known appearance, and the detection of the hand is based on tracking an object but not a robot hand. The identification of a robot hand wearing a colored glove is performed in [Nagi et al., 2011]. Both these techniques simplify the robot self-identification, but impose some limitations. Since these algorithms are not independent of the appearance of the robot and its behavior, they cannot be easily adapted to changing appearance of the robot's body, nor motion patterns. The independence on a priori knowledge would enable to generalize the self-identification over new end-effectors, like a robot part extended by a grasped tool. The ability to perform actions using a tool, for example, in order to enlarge the robot's workspace while reaching a distant object, is already a step towards the development of high-level cognitive functions.

An early prototype of a biologically inspired self-recognition approach is developed by W. Grey Walter in 1949 based on robots called Tortoises. These robots had a lamp attached to their head, and the interpretation of its light in a mirror can be considered as a kind of self-recognition. This example suggests that self-identification can be achieved through perception of changes in the environment resulted from own actions [Holland, 1997].

6.1.1 Sensorimotor studies

The sensorimotor studies in robotics take inspiration from both developmental psychology discussed in Chapter 2 and neuroscientific aspects. From a neuroscientific perspective, studies on own body perception indicate about neurons in a monkey cortex that respond to both visual stimuli with own hand incorporating its extension [Iriki et al., 1996]. From developmental psychology, humans and animals show the ability to acquire own body representation during their development [Rochat and Rochat, 2009].

The early work on detection of a robot hand based on its motion is presented in [Marjanovic et al., 1996]. The limitation of this approach is its assumption of a single moving region in a visual scene. However, in real environments, the visual motion can be produced by different sources of motion that can be a robot, a human, or other actors, or even elements of the environment that are influenced by these actors. Considering the visual motion as a consequence of an action performed by a physical actor, the visual motion should be almost immediate with the performed action. Following this idea, the localization of robot hands in the visual space can be based on the time-correlation between the robot's action and visual motion, like it is done in the research studies [Metta and Fitzpatrick, 2003], [Michel et al., 2004], and [Gold and Scassellati, 2005].

The time delay between the initiation of the robot's movement and the emergence of the robot parts in the scene is learned in order to identify a robot hand in [Michel et al., 2004]. The robot hand is identified as the first moving entity appearing in the visual field within the learned time window after the initiation of the robot's action. This algorithm is able to

detect both the robot's motion and its reflection in a mirror, but it is limited to one active source of motion at a time.

The localization of the robot parts based on the correlation between the velocity of the robot's movements and the optical flow in images, is proposed in [Metta and Fitzpatrick, 2003]. This approach allows to localize the robot arm in images without a prior information about the robot's visual appearance and also in case of several sources of motion.

The integration of multimodal sensorimotor experiences gathered during interaction with the environment and self-observation, is used for acquisition of the robot's body schema in [Grzyb and del Pobil, 2008]. The basic body schema is acquired by analogy with infants development at the stage of the Piaget's primary circular reactions. This developmental stage is reproduced in a robot based on simple movements repeated, according to their model, for a pleasure as a variable that increases, when a certain tension discharges. During these robot's movements, the correlation between motor commands and motion in the visual field is analyzed similar to [Metta and Fitzpatrick, 2003] and used to acquire the robot's body silhouette. The basic body schema including both kinematics (length of body parts and their relative positions) and dynamics (weight, inertia) is constructed by analyzing the robot's motion pattern and end-states through proprioception. The following development of the body schema is accomplished through interaction with the environment by analogy with the Piaget's secondary circular reactions, when the robot learns about its own body through simple actions with objects.

A developmental approach that enables a humanoid robot to define its body based on visuomotor correlation, is proposed in [Saegusa et al., 2012]. The visuomotor correlation is estimated using the proprioceptive and sensory information explored during head-arm movements. During the learning stage, the robot performs movements generated by stochastic motor babbling and senses the visual and proprioceptive feedback in terms of the speed of visual motion and the speed of a group of robot's joints. In case of high correlation, the robot identifies the moving object as its body part and memorizes the visuomotor information by accumulating the body posture and visual features in the visuomotor memory. This method enables the robot to anticipate visual images of its own body and it is adaptable to extended body parts. The robot can detect its arms also in occlusion and can predict the appearance and location of its arms. Continuing this work, the identification of the robot's own body is used for learning actions, such as fixation, reaching and grasping objects [Saegusa et al., 2013].

6.1.2 Identification of self and others

An approach aimed at learning about the robot's actions and actions of other physical actors through contingency, is proposed in [Gold and Scassellati, 2005]. In this method,

self-recognition is achieved by analyzing the time delay between the robot's action and the changes in the environment. The generic method of understanding a dynamic environment is based on actions and perception of responses of these actions :

- responses followed almost immediately after actions are considered as sensing the robot's own effectors ;
- responses delayed from actions or responses continued a bit longer after finished actions are considered as sensing the effects of own actions on the environment ;
- responses further delayed from actions are considered as sensing of actions of other physical actors.

The autonomous discovering of the robot hand during a natural interaction with a human is proposed in [Kemp and Edsinger, 2006]. The system uses a spherical camera in the body's reference frame and analyzes the visual input and proprioceptive sensing. Mutual information is used to identify which salient region of the visual space and which visual features are influenced by the robot hand. Since the visual system seeks to detect human and robot parts, it focuses on regions that are close to the camera or move with a high speed.

6.2 Interactive perception

The observation of an environment provides some information about its objects, but often this information is not sufficient, if we need to perform tasks with objects. In order to obtain a comprehensive information about an object and to build its useful representation, the object should be explored through interaction. Interactive learning allows to learn an overall object appearance and to discover object properties [Kyrki and Kragic, 2008]. The robot's ability to actively explore its environment and objects is also known as active vision [Kootstra et al., 2007].

The active exploration of the environment can be performed through displacement of a robot or by executing robot actions aimed to interact with its environment. In case of learning through robots actions, the identification of parts of the robot body is really useful. The robot self-identification provides a better motor control and more powerful processing of the visual information during and after interaction. Self-identification helps to analyze changes in the visual scene resulted from interactive actions.

The interactive exploration of the robot's environment and objects is investigated in many research studies ; we focus on

- detection and segmentation of objects from the background, performed in [Metta and Fitzpatrick, 2003], [van Hoof et al., 2012], [Kootstra et al., 2007],
- object learning and recognition, performed in [Ude et al., 2008], [Natale et al., 2005], and [Browatzki et al., 2012].

6.2.1 Detection and segmentation of objects

Active object detection and segmentation can be performed by executing robot actions aimed at interacting with its environment, like in [Beale et al., 2011], [Kootstra et al., 2007], [Katz et al., 2010], [van Hoof et al., 2012], and [van Hoof et al., 2012].

The early work on interactive object segmentation [Metta and Fitzpatrick, 2003] proposes to segment objects using simple interactive actions, like poking. The work shows that simple actions, without any complex manipulations, already facilitate segmentation of objects. Once the robot's arm is localized, the boundaries of contacted objects can be identified.

Interaction-based object identification based on push and grasp actions is performed in [Beale et al., 2011]. The visual information available during interaction, is processed with an assumption that pixels within objects move together, that is called "what-moves-together-belong-together" technique [Kootstra et al., 2007]. In the motion-based object segmentation approach [Katz et al., 2010], the robot induces an object motion and segments it based on the assumption that parts of a single rigid body have similar spatial, temporal, and appearance characteristics.

Another motion-based object segmentation method [van Hoof et al., 2012] is focused on selection of maximally informative actions to decompose a scene into objects. The changes in the visual field resulted from robot actions are analyzed not for single image pixels but rather on region-based level, where regions are composed from pixels that are close in the Euclidean and color spaces. Then, image regions are grouped into objects assuming that the regions of the same object should move together, when one of them is pushed. By this way, objects are represented as graphs of connected segments.

6.2.2 Learning object appearance

Active object learning can be accomplished either, when the robot moves around an object, or the robot interacts with the object. In the first case, the displacement of the robot or a camera around an object is used to observe and to learn the object appearance from several viewpoints, like in [Kootstra et al., 2007] and [Paletta and Pinz, 2000]. In the second case, manual object exploration is used to learn an object appearance from several perspectives during manipulations, like in [Ude et al., 2008], [Natale et al., 2005], and [Browatzki et al., 2012].

In some research works on manual object exploration, the robot detects and grasps an object by itself, in other scenarios, a human partner provides an object to the robot. For example, in [Ude et al., 2008], a user places a new object directly into the robot hand, that simplifies the system, since it does not need object detection and grasp planning. In this work, at first, the robot moves an object away from the camera and learns the background model. Then, the robot moves the object closer to the camera, places it in the center of the visual field, and acquires object views by subtracting already learned background. The robot rotates the object

and generates its representation model from snapshots acquired from different viewpoints.

Another object learning approach that starts with an object already localized in the robot hand, is proposed in [Browatzki et al., 2012]. In their algorithm, the object segmentation is accomplished by cropping a central part of the acquired image and by removing an already learned background. The representations of an object or an object class is generated from a collection of object views acquired from various viewpoints. The general idea is similar to [Ude et al., 2008] with the difference, that an object representation is based only on representative views acquired by estimating the object orientation adding maximum new information. This approach also includes interactive object recognition, as described below.

In [Natale et al., 2005], interaction object learning starts, when the object is also placed in the robot hand and detected by tactile sensors of the palm. During manual exploration, the robot approaches the object closer to the camera in four different positions and orientations. Fixation on the object is accomplished by tracking the robot hand. The object model is trained using few images acquired at each object position. Learned objects models are used to modulate the robot's attention in a top-down way, while searching for objects in the visual scene. When an object is recognized in the visual field, the robot estimates the object's orientation and plans a grasp. Reaching and grasping the object are controlled by using a previously acquired robot's body-schema.

Perception and action can be integrated into autonomous learning, when a robot detects and grasps objects by itself and learns objects through interaction without a help of humans. Furthermore, several research studies are aimed at selecting and planning actions that would provide a certain effect, like a successful grasp or turning an object into a representative viewpoint [Katz et al., 2010], [Natale et al., 2005]. The generation of grasping hypotheses that allow to accumulate objects features is performed in [Katz et al., 2010]. In this work, perception and interaction are integrated for autonomous acquisition of kinematic structures of rigid articulated objects. The executability of each generated grasp is verified in order to gather object-specific grasping knowledge.

Active perception is used not only to learn about objects, but also to recognize objects, like in [Kootstra et al., 2007], [Paletta and Pinz, 2000], and [Browatzki et al., 2012]. Object recognition can be based on a robot's interactive actions executed in ambiguous situations, when more evidences are needed for recognition. For example, object recognition based on reinforcing robot actions to turn the object into a discriminative viewpoint, is performed in [Paletta and Pinz, 2000]. The perception-driven recognition approach [Browatzki et al., 2012] distinguishes between similar objects by turning an object into a representative perspective that allows to recognize it ; likewise, objects are recognized through rejection of views of other probable objects.

Other research studies on interactive learning are aimed at selecting an object to explore. Object learning can be based on artificial curiosity [Guerin, 2011], [Oudeyer et al., 2007],

where an object and an action are chosen based on a certain notion of interest, and the robot focuses on less explored objects, while monitoring the learning progress.

6.3 Conclusion

Interactive perception provides a powerful capability to explore the robot's environment and to learn objects autonomously and efficiently. We are going to use interaction with objects in order to improve the knowledge about objects and to learn their overall appearances. Meanwhile, we are interested to identify parts of the robot's body in the visual space. We focus on a generic algorithm that is independent on the robot's appearance and on motion pattern. We are going to analyze the mutual information between the visual and proprioceptive data, similar to [Kemp and Edsinger, 2006], but without constraints on the object speed and its positions from the camera. Our approach has also some similarities with the self-identification concept proposed in [Saegusa et al., 2013]. The apparent distinction of our algorithm consists of analyzing not the speed but rather the localization of motion in the visual field and at the same moment, estimating a robot's arm-torso configuration based on the robot motors states.

The prediction of a robot arm location and action learning are outside of our goal, we rather focus on interactive object learning using advantage of identification of robot parts in the visual space. We are going to improve object models through manipulations. However, during manipulations, an object is grasped by the robot and often overlapped by a robot hand. Therefore, we use the ability to discriminate robot parts from the object in order to correctly update object models. Finally, we take advantage of our approach to measure the quality of objects models that is used to select objects and actions based on curiosity.

CHAPITRE 7

Active perceptual system implementation

In this chapter, we describe the proposed perceptual approach that enables the robot to learn about its close environment through interaction. If learning through observation (presented in Part I of this thesis) allows to detect physical entities in the visual space and to acquire some information about their visual appearance, the major part of information about the environment, its elements, their visual appearances and other properties, can be explored only through interaction. Therefore, we enhance the perceptual system implemented in Part I by integrating the possibility of interactive object learning.

In order to interact with physical entities detected in the environment, at first these entities should be localized with respect to the robot. Therefore, we calibrate the visual sensor relative to the robot, and we estimate the position, orientation and dimensions of each entity in the operational space of the robot, as described in Section 7.1.

During the robot's motor activity, the perceptual system analyzes both sensory information and proprioceptive data, and based on mutual information between these senses, the system identifies the parts of the robot's body among detected physical entities. Among other physical entities, human parts are discriminated from manipulable objects based on their motion behavior, as described in Section 7.2.

Once the robot is able to categorize physical entities, it starts to interact with objects. Both simple interactive actions and manual object exploration are used to improve the knowledge about objects' appearances, as described in Section 7.3, and to enhance their representation models acquired by observation.

The main modules of the proposed active perceptual system are shown in Fig.7.1.

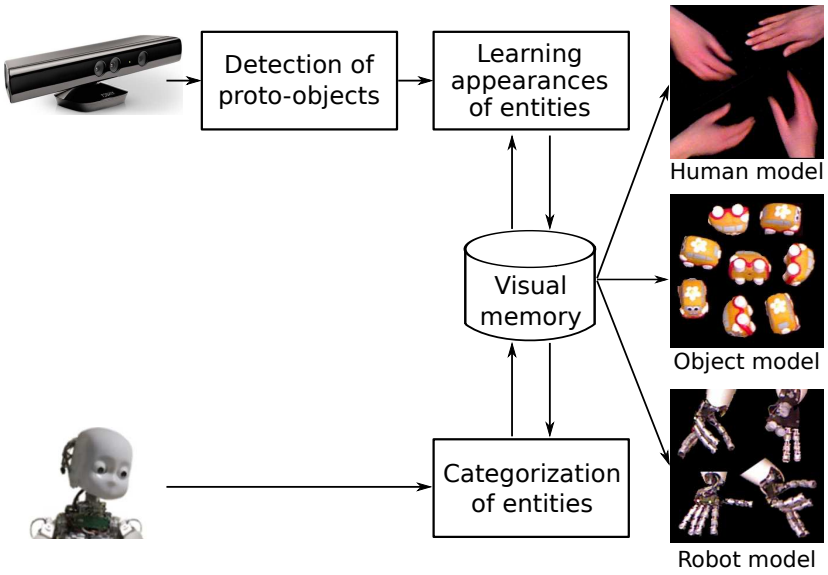


FIGURE 7.1 – The main modules of the proposed active perceptual system : in addition to the modules implemented in Chapter 4 (see Fig.4.1), new categorization module classifies physical entities into parts of the robot's body, parts of a human partner, or manipulable objects, and the learning module is enhanced by the possibility of interactive learning

7.1 Entity localization

Interactions with objects require their localization in the operational space of the robot. In our scenario, the visual input is acquired from the external sensor, thus, the localization of physical entities provides their position in the reference of the sensor. The localization of entities with respect to the robot requires to change the reference between the sensor and the robot. Once the visual sensor is calibrated with respect to the robot, the 3D position of each entity can be estimated in a the robot's space. Further, we estimate the entity's orientation and its real dimensions that will be used later to plan interactive actions.

7.1.1 Localization of entities with respect to the sensor

At first, the 3D position of each entity is estimated with respect to the sensor (as shown in Fig.7.2) by retrieving the depth data from the RGB-D sensor and processing them as a point cloud. The depth data are acquired from the sensor as a depth-map matrix, where each value is a distance (in meters) between a point in space and the sensor. We compute the coordinates of each point of the cloud using the formula :

$$x = x_p \frac{z}{d}, \tag{7.1}$$

where x_p is a coordinate of a pixel in the depth-map (shown in Fig.7.2), z is the distance between the sensor and an object, and d is the distance from the sensor to the image projection

of the object.

In order to estimate d, we use an imaginary point on the focal plane, with coordinates (x_p, y_p, d). Given that $\tan(\alpha) = \frac{x}{z}$, we have $x = \tan(\alpha)z$, so $x_p = \tan(\alpha)d$. Moreover,

$$if x_p = \frac{x_{res}}{2}, \alpha = \frac{fov_H}{2} \tag{7.2}$$

$$\frac{x_res}{2} = d\tan(\frac{fov_H}{2}), \tag{7.3}$$

$$d = \frac{x_{res}}{2\tan(\frac{fov_H}{2})} \tag{7.4}$$

With this value d, we compute the coordinates of the points :

$$x = 2z\tan(\frac{fov_H}{2})\frac{x_p}{x_{res}}, \tag{7.5}$$

$$y = 2z\tan(\frac{fov_V}{2})\frac{y_p}{y_{res}}, \tag{7.6}$$

where x_p and y_p are the coordinates of a pixel in the depth-map, x_{res} and y_{res} are the sizes of the depth-map, fov_H and fov_V are the horizontal and vertical field of view of the RGB-D sensor (in radians).

The obtained point cloud represents the scene observed by the sensor with dimensions of the real scene.

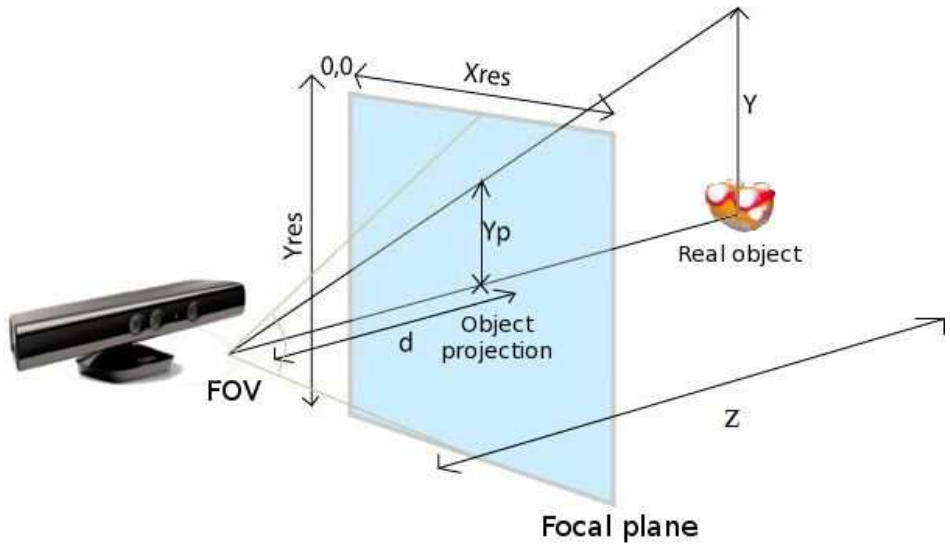


FIGURE 7.2 – The position of the entity with respect to the sensor

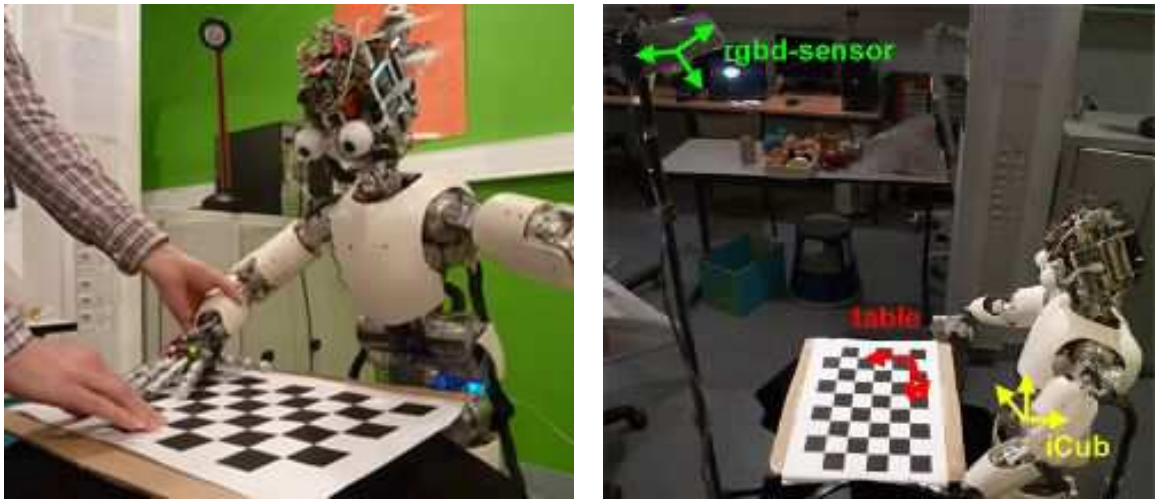
7.1.2 Changing the reference frame between the sensor and the robot

Since the robot performs actions in its operational space, we calibrate the sensor relative to the robot's base. The change of the reference frame between the sensor and the robot requires a transformation matrix that can be obtained using a calibration pattern, like a chessboard, since OpenCV library allows to automatically compute the position of the sensor relative to a chessboard.

In order to compute the transformation matrix, we need both the position of the chessboard and its orientation. The orientation of the chessboard is supposed to be known, since we place the chessboard with a certain orientation. The position of the chessboard with respect to the robot is estimated based on the position of the robot hand placed above the chessboard, since the robot can communicate the position of its hand. Therefore, we move one robot hand to the origin of the chessboard (as shown in Fig.7.3) and acquire its position in the operational space of the robot.

Then, the transformation matrix is computed in the following way :

$$T_{sensor \to robot} = T_{sensor \to chessboard} \times T_{chessboard \to robot}. \tag{7.7}$$



a) b)

FIGURE 7.3 – The calibration of extrinsic parameters of the sensor with respect to the robot : a)the acquisition of the position of the calibration pattern in the operational space of the robot ; b)the reference frames of the sensor, the robot, and the calibration pattern

7.1.3 Localization of entities with respect to the robot

Once the camera is calibrated, the position of entities can be estimated in a new reference frame. The position of an entity is computed as an average position of its points.

The orientation of entity's axes is estimated from eigenvectors and eigenvalues of the covariance matrix of all entities points. The eigenvectors corresponds to three orthogonal vectors oriented in the direction maximizing the variance of entity points along their axis. These eigenvectors are used as a reference frame of the entity.

The quaternion is chosen as a representation of entities' orientations, since this representation is compact, fast, and stable [Gaël and Benoît, 2010]. The reference frame of the robot and the reference frame of the entity are represented using vector triplets. In order to change the entity's orientation from one coordinate system to another, we search a quaternion that allows to align at first the x axis (as shown in Fig.7.4a), and when the x axis is aligned, we align two other axes (as shown in Fig.7.4b). Each quaternion is obtained from a vector product $\vec{Q_{axis}}$ and a scalar product Q_{angle} giving the axis-angle information :

$$\vec{Q_{axis}} = \vec{x_e} \wedge \vec{x_r}, \tag{7.8}$$

where x_e comes from the reference frame of the entity and x_r comes from the reference frame of the robot.

$$Q_{angle} = \arccos(\vec{x_e} \cdot \vec{x_r}), \tag{7.9}$$

$$Q = \begin{bmatrix} \cos(\frac{Q_{angle}}{2}) \\ \vec{Q_{axis}} \times \sin(\frac{Q_{angle}}{2}) \end{bmatrix} \times K, \tag{7.10}$$

where K is such that $|Q| = 1$.

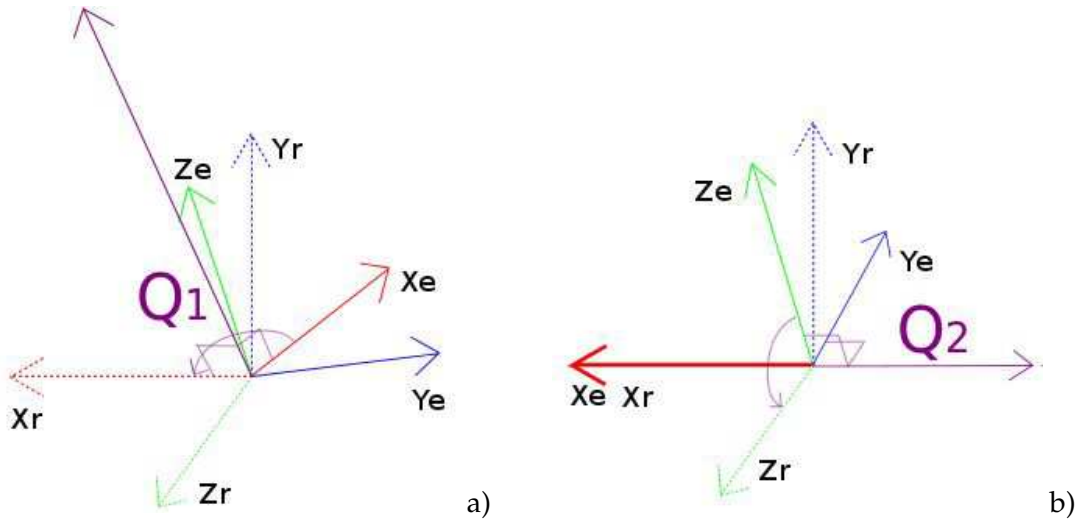


FIGURE 7.4 – Changing the reference frame : a)the first rotation aimed at aligning x axes, b)the second rotation aimed at aligning other axes

The final entity rotation is obtained using the Eigen3 [1] library taking a product of two rotations :

$$Q_{final} = Q_2 \times Q_1, \tag{7.11}$$

where Q_1 is the rotation aimed at aligning the x axis of an entity with the x axis of the robot (shown in Fig.7.4a), Q_2 is the rotation aimed at aligning other axes (shown in Fig.7.4b).

7.2 Entity categorization

The categorization procedure is aimed to identify the nature of physical entities detected in the visual space during natural interaction of the robot with a human partner while exploring the surrounding environment and learning objects. Each physical entity is classified into one of the following categories : a part of the robot's body c_r, a part of a human partner c_h, an object c_o, an object grasped by the robot c_{o+r}, or an object grasped by a human partner c_{o+h}. Before identification of the robot's body, all entities are temporally associated to the unknown category c_u, and their correct categories will be identified in next images.

During the categorization procedure, at first, the parts of the robot's body are discriminated among all physical entities, and then, the rest of single entities are distinguished either as a human part, or a manipulable object category, as shown in Fig.7.5. The connected entities are distinguished either as an object grasped by the robot, or an object grasped by a human partner category.

7.2.1 Implementation of the robot self-identification algorithm

The self-identification algorithm is aimed to identify the hands of the robot among all physical entities detected in the visual space, during interaction of the robot with a human partner.

The implemented algorithm requires minimum prior knowledge, it does not need the predefined appearance of the robot, the robot's joint-link structure, or the predefined pattern of the robot's motion. The independence on the robot's appearance allows to achieve robust recognition of the robot hands in case of changing appearance, in case of occlusion, while holding various objects, and in case of extension of robot's parts by grasped tools. The independence on the robot's behavior enables to perform a variety of interactive actions with objects in order to learn their appearance. The actions used in our work will be detailed in Section 8.1.

Taking inspiration from the child sensorimotor development described in Chapter2, we design a self-identification algorithm that enables the robot to learn about its own body

1. http ://eigen.tuxfamily.org

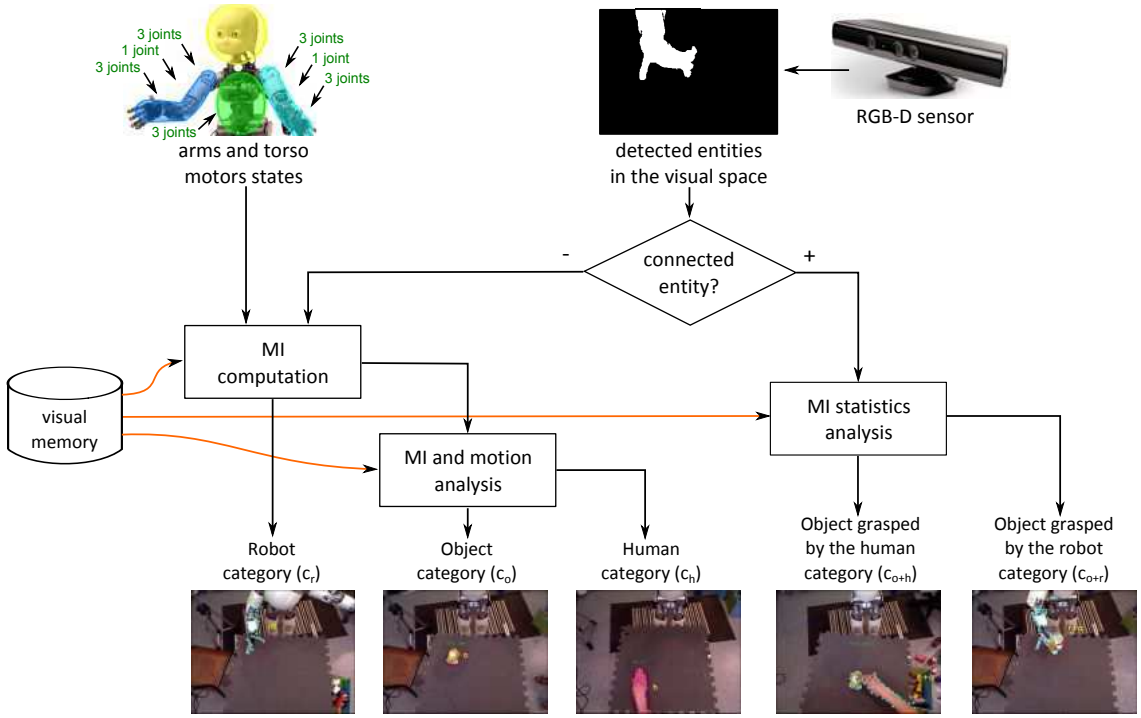


FIGURE 7.5 – The main steps of the categorization algorithm : mutual information (MI) estimated from the visual and proprioceptive data is used to identify parts of the robot's body among all entities, as described in Section 7.2.1 ; computed mutual information is stored in the statistics on categorization in the visual memory ; both the statistics on categorization and the statistics on entities motion are used to discriminate an object category and parts of a human partner, as described in Section 7.2.2 ; as output from the categorization module, each physical entity is assigned to one of following categories : a part of the robot's body c_r, a human part c_h, or an object c_o in case of a single (not connected) entity, and an object grasped by the robot c_{o+r} or an object grasped by a human partner c_{o+h} in case of a connected entity

following a developmental approach. The robot learns to identify its hands, like a child, by freely moving its hands in the visual space.

During the robot's motor activity, the visual information is gathered and analyzed together with the proprioceptive data :

- as visual information, the motion of physical entities is analyzed in terms of their position in the visual space,
- as proprioceptive information, the states the robot's arms and torso motors are analyzed in terms of values of the following joints :
  - arm joints, such as shoulder joints (pitch, roll, and yaw), elbow, wrist joints (pronosupination, pitch, and yaw), as shown in Fig.7.6.
  - torso joints, such as pitch, roll, and yaw.

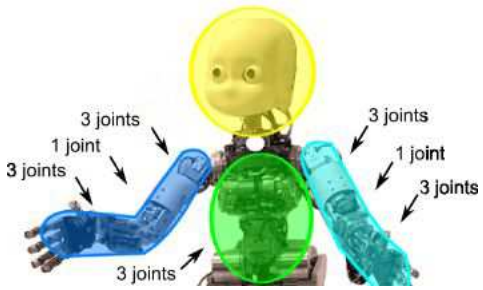


FIGURE 7.6 – The parts of the iCub body, where the head is shown by the yellow color, the torso motor group is shown by the green color, and the arm motor groups are shown by the blue and cyan colors

The states of all mentioned motors are acquired as a set of joint values without considering the functionality of each motor, nor the character of its impact on the displacement of the robot hands. The analyzed joints are chosen due to their influence on the position of the robot hands. The robot's head motor group is not analyzed, since it does not effect on the position of hands. Finger joints are not considered, since their motion do not produce a significant visual displacement of the robot hands.

Each time a new image is acquired from the visual sensor, the joint values of the analyzed motors are acquired through the corresponding arm and torso ports of the robot (as described in Section 8.2). Both visual and proprioceptive data are quantized in order to reduce the dimensionality. The visual space is analyzed at the level of visual clusters obtained by applying a grid (12x10) producing 120 rectangular regions in the image. The position of each physical entity is quantized into the closest visual cluster. The set of joint values is incrementally quantized into a dictionary of arm-torso configurations, where each entry is encoded by a vector of joints angles. In general, we get about 37 arm-torso configurations. The quantization is incremental, and it is based on the distance measure between the current set of joint angles and dictionary entries. If the maximal distance exceeds a specified threshold, a new configuration is added to the vocabulary ; otherwise, the vector of current joint

angles is assigned to the dictionary entry with the minimal distance. The distance measure between two sets of joint angles is computed as a L2 distance :

$$d(a_1, a_2) = \sum_j (a_{1_j} - a_{2_j}), \tag{7.12}$$

where a_{1_j} and a_{2_j} are the joint angles of the compared arm-torso configurations a_1 and a_2.

The correlation between the available visual and proprioceptive data is based on mutual information, similar to[Kemp and Edsinger, 2006]. Although, in our algorithm, the mutual information is used to evaluate the dependency of occurrences of the robot's arm-torso configuration A together with the physical entity E_i localized in the visual cluster L :

$$MI(L_{E_i}; A_{arm_k}) = H(L_{E_i}) - Hc(L_{E_i}|A_{arm_k}), \tag{7.13}$$

where L_{E_i} is the position of the entity E_i quantized into the visual cluster L, A_{arm_k} is the configutaion of the robot's arm arm_k quantized into the cluster A, $H(L_{E_i})$ is the marginal entropy, and $Hc(L_{E_i}|A_{arm_k})$ is the conditional entropy computed in the following way :

$$H(L_{E_i}) = -\sum_l p(l_{E_i}) log(p(l_{E_i})), \tag{7.14}$$

$$Hc(L_{E_i}|A_{arm_k}) = -\sum_{a_{arm_k}} p(a_{arm_k}) \sum_{l_{E_i}} p(l_{E_i}|a_{arm_k}) log(p(l_{E_i}|a_{arm_k})), \tag{7.15}$$

where $p(l_{E_i})$ is the probability of the physical entity localization l_{E_i} ; $p(a_{arm_k})$ is the probability of the arm-torso configuration a_{arm_k}, and $p(l_{E_i}|a_{arm_k})$ is the probability of the physical entity localization l_{E_i}, given the arm-torso configuration a_{arm_k}.

Since we change the appearance of the robot hands during our experiments presented in Chapter 8, different entities can characterize different hands' appearances (for example, hands with and without gloves, like shown in Fig.7.7). Each physical entity accumulates several views characterizing small changes in a hand's appearance resulted for example, by different hand's postures. Thus, $MI(L_{E_i}; A_{arm_k})$ is estimated for each robot's arm arm_k and for each physical entity E_i from the visual memory. Thereby, the robot category c_r can be associated with several entities that corresponds to different appearances of the robot hands.

According to our scenario, the robot's self-identification is accomplished, while the robot moves its hands in the visual space that results in growing mutual information for a corresponding physical entity. The entity is identified as the robot category c_r, when its probability, obtained by normalizing the mutual information to the maximum value for both arms and for all entities, exceeds a specified threshold th_r. On the contrary, the human and object categories have small mutual information due to their independence from the robot's motors. The threshold for identifying the robot category th_r is selected though empirical observation of the distribution of mutual information (shown in Fig.7.8) obtained for the robot's

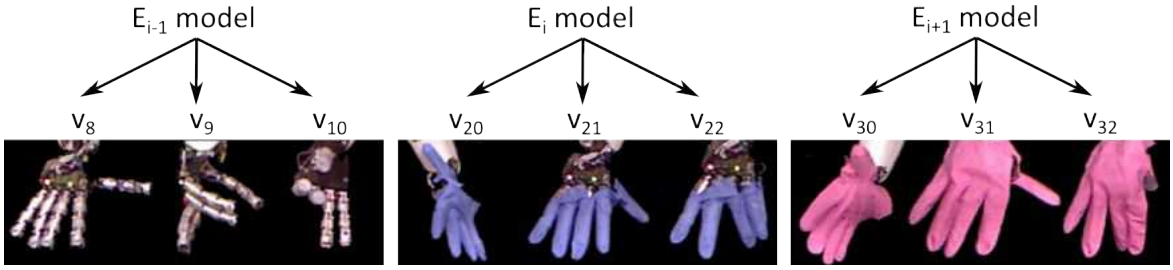


FIGURE 7.7 – The representation models of three different entities that correspond to the robot hands

and non-robot's entities on a small labeled database. Thereby, the physical entity is identified as the robot category c_r, if its probability of being a part of the robot's body is higher than $th_r = 40\%$, and otherwise, the physical entity is considered as one of the non-robot categories described in the next Subsection.

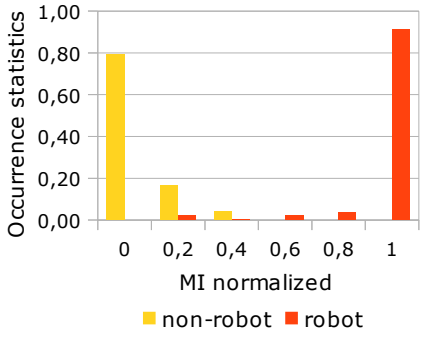


FIGURE 7.8 – The distribution of the normalized mutual information obtained for robot's and non-robot's entities on a small labeled database

7.2.2 Discrimination between the object and human categories

Most objects, like objects used in our experiments, are static most of time, and they are displaced only by external forces provided by the robot or its human partner. Some objects can move under the inertia, when they continue motion after interacting with them (for example, after pushing). In case of our scenario and performed interactive actions, the motion of objects under inertia occurs rarely, and among categories analyzed in our work, only the robot and the human categories can move alone (not connected to other entities). Thus, the object category can be discriminated from other categories based on the statistics on entities' motion (see Fig.7.5) as a static entity that moves only connected to other entities.

The statistics on entities' motion is gathered by the perceptual system while detecting physical entities in the visual space and identifying them either as single entities moving alone, or connected entities moving together. This statistics on entities' motion is analyzed together with output from the self-identification algorithm described in previous Section

(that identifies each entities as the robot or non-robot category), and the following statistics for each entity is accumulated :

- $N_{c_{E_i}}$ as the number of times when the entity E_i moves alone,
- $N_{c_{E_i} \neq c_r}$ as the number of times when the entity E_i moves alone and is identified as a non-robot's entity,
- $N_{c_{E_i}, c_{E_{i2}}}$ as the number of times when the entity E_i moves together with a connected entity E_{i2},
- $N_{c_{E_i}, c_{E_{i2}} = c_r}$ as the number of times when the entity E_i moves together with a connected entity E_{i2} identified as a robot's entity.

The gathered statistics on motion is analyzed separately for single and connected entities based on the following occurrence frequencies :

- $f_s = \frac{N_{c_{E_i} \neq c_r}}{N_{c_{E_i}}}$ as the occurrence frequency of moving alone as a non-robot's entity,
- $f_c = \frac{N_{c_{E_i}, c_{E_{i2}} = c_r}}{N_{c_{E_i}, c_{E_{i2}}}}$ as the occurrence frequency of moving together with a connected entity E_{i2} identified as a robot's entity.

Analyzing the statistics on motion of single entities, the occurrence frequency f_s of moving alone as a non-robot's entity should be low for the object category, since object's entities usually do not move alone, as discussed earlier.

Analyzing the statistics on motion of connected entities, the occurrence frequency f_c of moving together with a connected robot's entity should be high for the object category, since object's entities often move together with robot's entities, for example when the robot manipulates objects, and humans' entities often move together with non-robot's connected entities, for example when the human partner manipulates objects.

The discrimination between the object category c_o and the human category c_h is based on two chosen thresholds $th_o.s.$ and $th_o.c.$ evaluating the motion of single and connected entities. Thereby, each single non-robot physical entity is categorized as :

- the object category c_o, if its occurrence frequencies $f_c > th_o.c.$ and $f_s < th_o.s.$;
- the human category c_h, otherwise.

According to our scenario, the discrimination between the object and human categories is accomplished, while real objects are manipulated that results in gathering the statistics on entities' motion together with the robot and human hands. When a sufficient amount of statistics is accumulated, the robot is able to classify each single entity into one of the following categories : c_o, c_h, or c_r as shown in Fig.7.9.

If case of connected entities detected in the visual space, the category of each entity is retrieved from the statistics on categorization from the visual memory. The connected entities are categorized as :

- the object grasped by a robot category c_{o+r}, if the retrieved category of one connected entity is the robot category and another is the object category,
- the object grasped by a human category c_{o+h}, if the retrieved category of one connected

entity is the human category and another is the object object category,
– the unknown category c_u, otherwise, if none of mentioned conditions is satisfied, for example, the retrieved categories are the robot category and the human category.

a) b) c) d)

FIGURE 7.9 – Examples of categorized entities : a) the human hand is categorized as the human category c_r ; b) the robot hand is categorized as the robot category c_o, and the object is categorized as the object category c_o after interaction ; c) the object grasped by the robot are categorized as c_{o+r} ; d) the object grasped by the human hand are categorized as c_{o+h}

7.3 Interactive object learning

Interactions with environment allows to significantly enhance its exploration. If pure observation of the environment enables to detect objects in the visual space and to acquire some information about their appearance, as described in Chapter 4, interaction with objects allows to improve the knowledge about objects. In our study, interactive actions with objects are aimed to acquire maximum information about overall objects appearances from different viewing angles and at different scales.

7.3.1 Improving object models

Once the perceptual system is able to categorize physical entities detected in the visual space, object entities are explored through interaction with them. The robot performs object-oriented actions presented in Section 8.1. The information about overall objects appearance is gathered mostly through manual object exploration including grasping an object, turning around, and rotating in various directions that allow to observe the grasped object from different perspectives. The appearance at different scales is acquired by approaching an object to the visual sensor. During manipulations, all acquired information is synthesized with the previous knowledge gathered through observation, and used to improve an object representation model by updating it with recognized or newly created views.

Before the robot starts interaction with an object, the perceptual system identifies this object as one of physical entities. In case of a successful grasp, the system remembers the grasped entity as E_g, and the model of this entity will be updated during the manipulation process. This is a kind of self-supervision, where the object is supposed to be the same during

the manipulation. While interacting with the object, the categorization algorithm is able to discriminate the object entity from the robot entity, when they move separately or together, when the object is grasped.

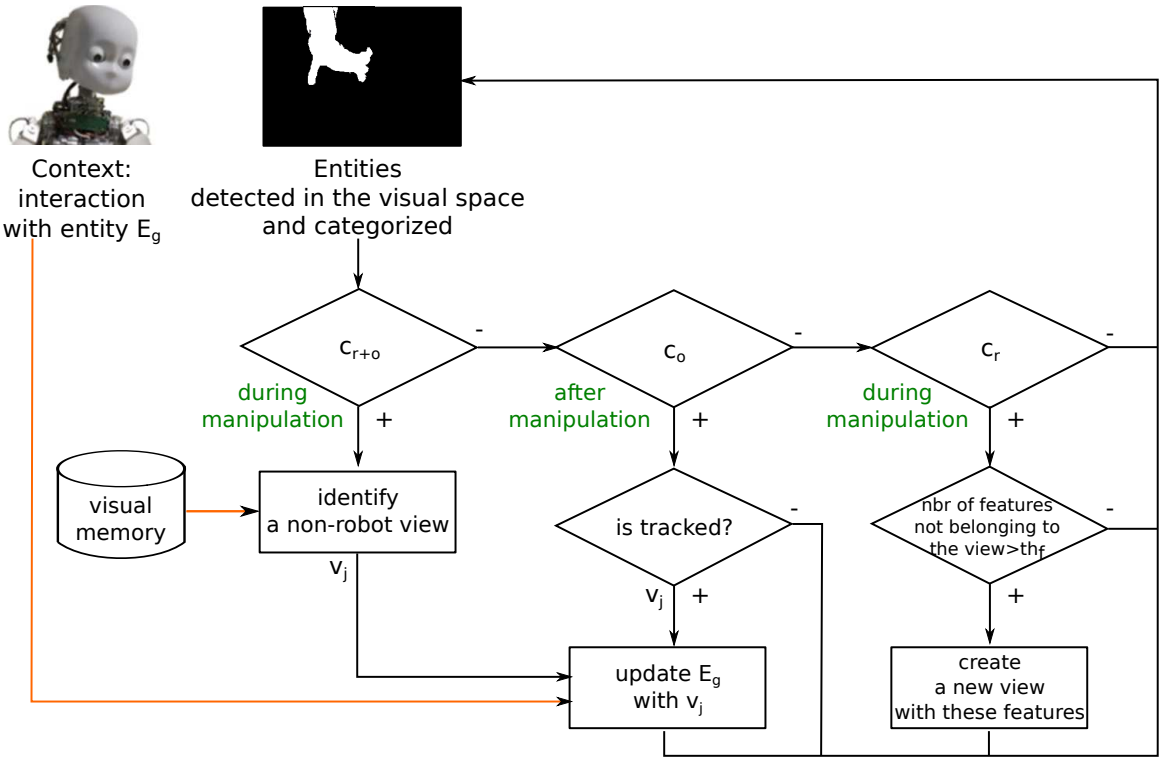


FIGURE 7.10 – Improving the object representation model during manual exploration

According to our algorithm, the perceptual system continuously detects entities in the visual space and categorizes them. In the context of interaction with an object, the learning procedure is summarized in Fig.7.10. If the perceptual system detects connected entities with one entity identified as the robot category, these entities are carefully analyzed. At this point, the object can be identified as an object category or one of non-robot categories, if it was never manipulated before. The categorization of entities has been described in Section 7.2, and here we verify the category of each connected view, in order to prevent erroneous recognition and to perform learning only in assured cases. Each connected view is associated with a set of physical entities $\{E_i\}$ that have this view in their models. The category c_{E_i} of each entity from the analyzed set is retrieved from the statistics stored in the visual memory, and a view is identified as :

- a robot view, if at least one corresponding entity is identified as the robot category $(\exists i, c_{E_i} = c_r)$;
- a non-robot view, if none of corresponding entities is identified as the robot category $(\forall i, c_{E_i} \neq c_r)$.

If connected views are identified as a robot's view and a non-robot view (see Fig.7.11), the

manipulated entity's model is updated with a non-robot view.

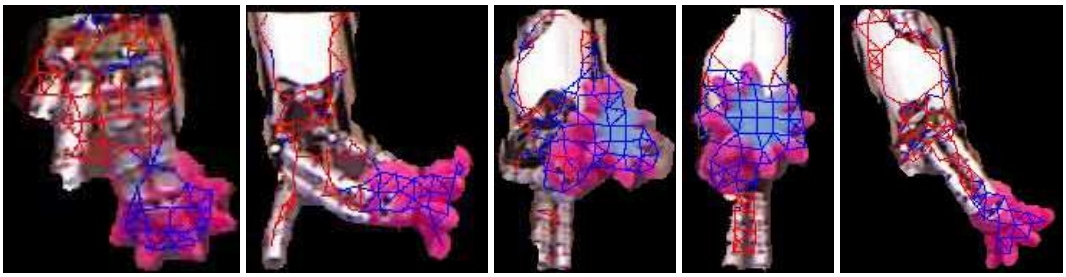


FIGURE 7.11 – Examples of connected views with their mid-features (HSV pairs) : the red mid-features correspond to one connected view (in this case, the robot hand), and the blue mid-features correspond to another connected view (in this case, an object)

If during the manipulation procedure, the perceptual system detects an entity identified as the robot category with a significant part of features that do not correspond to the entity, then a new view with the set of these features is stored in the Visual memory. If a newly created view will be identified in next images, it can be added to the model of the manipulated entity.

At the end of the manipulation procedure, the robot releases its hand, and the grasped object falls on the table and appeared from an unpredicted perspective, that can be still in the visual space of the robot. In this case, if the perceptual system detects single entities identified as the robot and the object category, and the object is tracked from the previous image, we consider, that we observe an effect of the interactive action, and the model of the manipulated entity is updated with the current object view. Thereby, the robot can explore the object appearance by grasping it, throwing, and updating its model with the observed view. As a summery, the active perceptual system is capable to learn objects appearances during manual exploration and in between interactive actions.

7.3.2 Cleaning visual memory

After manipulations, the perceptual system performs a check of the visual memory and cleans the dictionaries of entities and views. The dictionary of entities is filtered by suppressing noisy entities that have no proper views associated only with this entity. The dictionary of views is filtered by suppressing views that have no associated entities ; such views could be created during learning but never adding to entities models. The filtering of both dictionaries makes the robot's knowledge about physical entities more coherent, and it should improve the object recognition.

7.4 Curiosity-driven object exploration

The active perceptual system implemented within the scope of this thesis composes a part of the research performed within the MACSi project [2]. The goal of the MACSi project is to provide a humanoid robot with developmental capabilities that enables the curiosity-driven exploration of the environment.

The curiosity-driven learning takes inspiration from spontaneous attraction of humans toward different activities known as Intrinsic Motivation [Edelman, 1997]. In robotics, the curiosity-driven learning is based on monitoring the evolution of learning progress. In case of object exploration, curiosity-driven mechanisms measure the learning performance while exploring one object, and switch the attention to another object, when a certain learning progress is achieved. The curiosity-driven exploration of objects is aimed at focusing on a less explored object until gathering a certain quality of knowledge about this object.

In this work, the curiosity-driven exploration of objects integrates interactive object learning with intrinsic motivation within a multi-module Cognitive Architecture described in Section 8.1.3. Among the modules of this architecture, the implemented perceptual system constitutes one module, the curiosity mechanism is incorporated to another module. The Curiosity module uses the intrinsic curiosity mechanism based on the Socially Guided Intrinsic Motivation with Active Choice of Teacher and Strategy (SGIM-ACTS) algorithm [Nguyen and Oudeyer, 2013]. The implemented perceptual system detects physical entities in the visual space and communicates the information about them to the Curiosity module that estimates the learning progress. Since the perceptual system characterizes each detected entities by its multi-view representation model, as described in Section 7.3, the quality of this model is used to estimate the learning progress. Therefore, the perceptual system transmits the following information about each detected entity to the curiosity-driven module :

- 3D position,
- orientation,
- the id of the physical entity and the id of its currently observed view,
- the probability of recognizing the physical entity,
- the probability of recognizing the view,
- the total number of views in the representation model of the physical entity.

The learning progress is evaluated using the recognition performance and the quality of representation models estimated based on the number of views in the model, the recognition performance, and the recognition rate of each view. The precise description of the SGIM-ACTS algorithm is outside the scope of this manuscript, but details can be found in [Nguyen and Oudeyer, 2013]. As a general behavior, this algorithm choose actions that may lead to the highest learning progress using the intrinsic motivation mechanism. The

2. http ://macsi.isir.upmc.fr

intrinsic motivation is combined with a social guidance in the following ways :

- social cheering is used to encourage the robot to pursue some of sensorimotor activities and to abandon others,
- stimulus enhancement is used to attract the robot's attention to one of objects, which is then learned by the robot on its own.

The balance between the autonomous curiosity-driven exploration and social guidance allows to achieve continuous and efficient learning.

7.5 Conclusion

In this chapter, we have described an extension of the perceptual system described in Chapter 5 to interactive scenario, where the robot explores the environment though interactive actions. Physical entities detected in the visual space are categorized into parts of the robot's body, human parts, and manipulable objects. The categorization algorithm requires minimum prior knowledge, it does not need predefined appearance of the robot, its joint-link structure, or predefined pattern of motion.

The categorization of entities is used to enhance the interactive object learning aimed to gather the information about an object appearance and to integrate it in its representation model. During interactions with an object, the categorization procedure enables to distinguish the robot hands (its entities and views) from an object, even if an objects is grasped. Thus, a representation model of the manipulated object can be updated with non-robot views preventing erroneous updates and improving the knowledge about the object.

CHAPITRE 8

Experimental evaluation of the active perceptual approach

The performance of the implemented active perceptual system is evaluated on the iCub humanoid robot exploring its environment in an interactive scenario. The experimental setup, the scenario, and the robot's actions performed during our experiments are described in Section 8.1.

During the experiments, the robot is free to move its hands, head, and torso. The robot's actions are aimed at first at categorizing entities by identifying parts of the robot's own body in the visual space and discriminating manipulable objects from other detected entities. The evaluation of the categorization performance is presented in Section 8.2.

Once the robot is able to categorize entities localized in the visual space, it interacts with objects in order to improve the knowledge about their appearances. The evaluation of interactive object learning is presented in Section 8.3. The results obtained during interactive learning are compared with the results obtained during learning through observation presented in Part I of this thesis.

8.1 Experiment setup

The experimental setup is similar to the one already described in Chapter 5 with the main difference, that the robot can interact with its environment through actions directed to entities detected in the visual space. Thus, the visual sensor is calibrated with respect to the robot, and all physical entities detected in the visual space are localized in the operational space of the robot, as described in Section 7.1. The robot learns about the environment and its entities through interactive actions and observation of these actions and resulted changes in the visual space.

8.1.1 Reference frame

All robot's actions performed in our experiments are based on the robot's root reference frame located in the middle of the torso, like shown in Fig.8.1. The axes of this reference frame are :

- z axis pointing upwards parallel to gravity,
- x axis pointing behind the robot,
- y axis pointing laterally as the right hand.

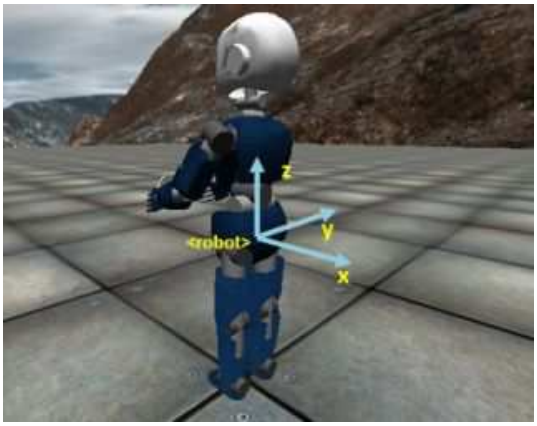


FIGURE 8.1 – The root reference frame of the iCub robot

8.1.2 Description of the robot's actions

The robot's interactive actions are aimed at achieving the following goals :

- identification of parts of the robot's body that can be accomplished through free hand motion or random repetitive actions, like infants do while exploring their own body,
- discrimination of manipulable objects from other physical entities that can be accomplished through actions directed to entities detected in the visual space and observation of effects of performed actions,
- learning objects appearances that can be accomplished through object manipulation and certain object-oriented interactive actions.

The robot's actions require the control of motor joints that can be accomplished by sending commands to the robot. Among the control commands, we use both simple action primitives, like *reach*, *push*, *take*, and complex manipulations designed in [Ivaldi et al., 2012a]. These actions have been implemented by our partners in ISIR in the frame of the MACSi project. The simple action primitives used during experiments lead to execution of commands :

- *reach* action primitive leads to moving a robot hand to the position above an object, while keeping fingers open,

– *push* action primitive consists of reaching an object from a side and pushing it by one robot hand in the direction of the other hand,
– *take* action primitive leads to a three-finger pinch grasp from the top of an object.

The complex manipulations are aimed to explore an object appearance in a most informative way, and they are designed as a sequence of action primitives :

– *TakeLiftFall* manipulation consists of reaching an object, taking it, lifting, and releasing that allows to see a random object perspective, when the object falls on the table,
– *TakeObserve* manipulation consists of reaching an object, taking it, turning, approaching to the camera, and returning to the table, that allows to observe different object perspectives and appearance details at a close scale during manipulation.

After each action the robot returns its hand to the initial position that is outside the field of view of the sensor. In order to ensure compliance during actions, the impedance control is activated at the main joints of the torso and arms. The estimate of joint torques and external forces is based on proximal force/torque sensors located in the middle of arms, and it is performed using an iDyn library for inverse dynamics [Ivaldi et al., 2011].

8.1.3 Software architecture

The robot is controlled through a multi-module *Cognitive Architecture (CA)* combining perception, action, and curiosity-driven behavior [Ivaldi et al., 2012b]. This architecture is especially designed for learning in the context of developmental robotics. The work is performed within the MACSi project [1], and it based on the development of the following robot's skills needed to learn about the surrounding environment :

– *perceptual skills* are developed within the scope of this thesis and include the capabilities of the implemented perceptual system,
– *motor skills* including adaptation and extension of robot control techniques used to perform actions and to learn generic affordances during interaction with objects [Ivaldi et al., 2012a],
– *exploration of sensorimotor spaces in changing body and environment* including adaptation of intrinsic motivation systems [Nguyen et al., 2013].

All skills developed for the iCub robot are integrated into the Cognitive architecture (shown in Fig.8.2), where each module provides certain functionalities implemented by one of the partners : ENSTA ParisTech [2], GOSTAI [3], INRIA [4] and ISIR [5]. The communication between modules and the robot is accomplished through the YARP middle-ware providing an interface to the robotic hardware and devices, as described in Section 5.1.1.

1. http ://macsi.isir.upmc.fr
2. http ://cogrob.ensta-paristech.fr
3. http ://gostai.com
4. http ://flowers.inria.fr
5. http ://isir.upmc.fr

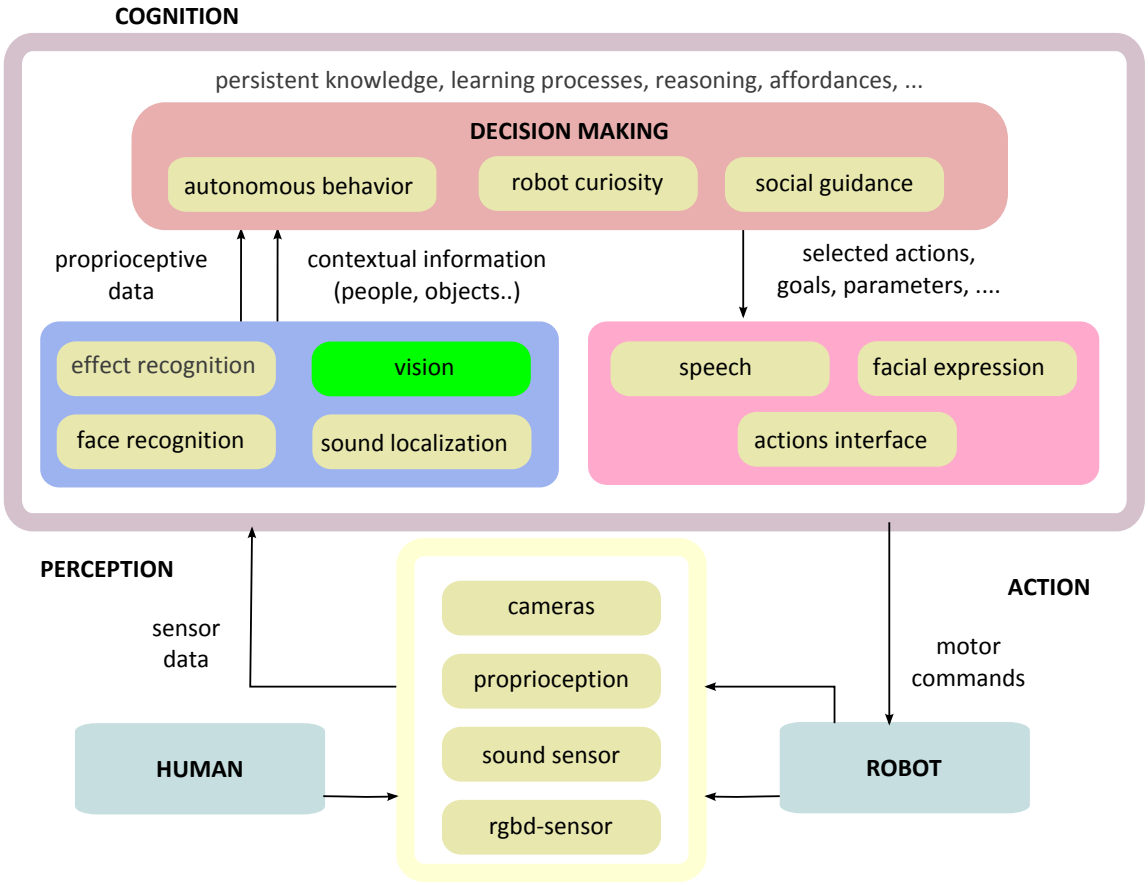


FIGURE 8.2 – The organization of the robot's skills into the multi-module Cognitive Architecture, where the perceptual system implemented within the scope of this thesis is embedded as a Vision module highlighted by the green color

The perceptual system implemented within the scope of this thesis is embedded in the CA as a *Vision* module that communicates with the Decision Making module and acquires some data directly from the robot. As the outcome, the Vision module send the information about detected physical entities to the Decision Making module (as described in Section 7.4) through the output port $/vision/objInfo : o$. As input data, the Vision module receives the information about the currently performed action from the Decision Making module through the input port $/vision/actionInfo : i$. Furthermore, the Vision module required the states of the robot arms and torso needed for the entity categorization algorithm ; these data are acquired from the robot using the following YARP ports :

- $/iCub/left\_arm/state : o$ connected to our port $/vision/part\_armLeft : i$,
- $/iCub/right\_arm/state : o$ connected to our port $/vision/part\_armRight : i$,
- $/iCub/torso/state : o$ connected to our port $/vision/part\_torso : i$.

8.1.4 Scenario

In our scenario, a human partner interacts with the robot, like it is described in Section 5.1, and the robot interacts with its surrounding environment in order to explore it efficiently. At first, a human partner demonstrates objects to the robot, and each demonstration lasts about one minute and contains in average 500 images per object. Then, the robot explores its environment through interaction. The robot performs simple repetitive actions aimed at exploring the visual space and identifying its own hands ; these actions last about 8 minutes and contain about 3000 images. Further, the robot interacts with its close environment through object-oriented actions. In average, a simple action, like *push*, lasts about 0,5 minute and contains about 250 images ; a complex manipulation, like *TakeObserve*, lasts about 1,5 minutes and contains about 750 images.

8.2 Evaluation of entity categorization

The robot's ability to categorize detected physical entities into parts of own body, human parts, and manipulable objects is evaluated in the interactive scenario while both the robot and its human partner perform actions aimed at exploration of the surrounding environment. The *categorization rate* is computed as a percentage of successfully categorized physical entities with respect to the total number of images with these entities.

8.2.1 Evaluation of self-identification

The robot's self-identification is evaluated on several pre-recorded image sequences and pre-recorded data from robot joints. In one sequence, the robot performs free hand motion and interactive actions described in Section 8.1.2 (see Fig.8.3) ; in total, these actions last about 12 minutes and contain about 4900 images. In the other sequence, both the robot and

its human partner move their hands in the visual space (see Fig.8.4) ; in total, this sequence lasts about 10 minutes and contains about 3800 images.

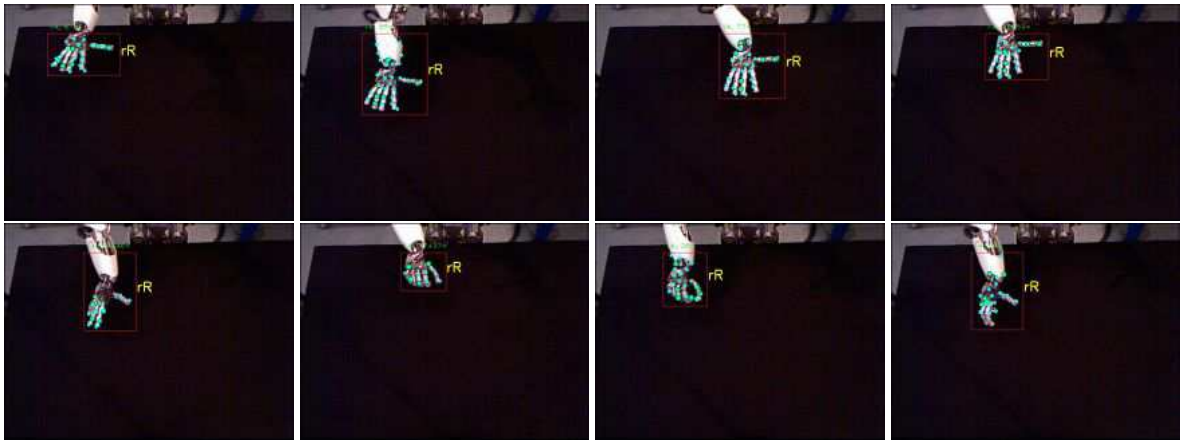


FIGURE 8.3 – Examples of images, where the robot performs free hand motion and simple repetitive actions

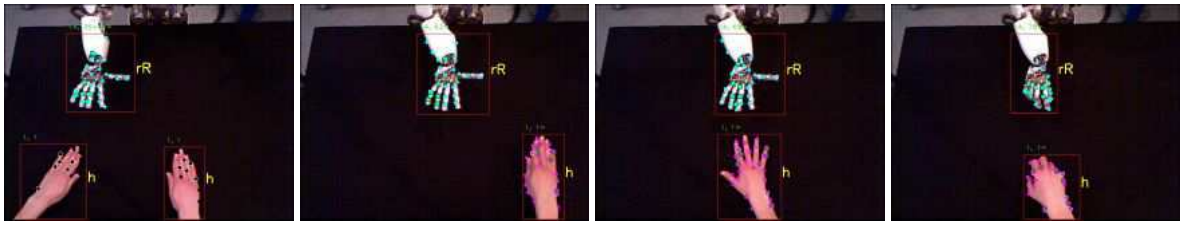


FIGURE 8.4 – Examples of images, where both the robot and its human partner move their hands in the visual space

The perceptual system continuously analyzes the sensory and proprioceptive data in order to discriminate parts of the robot's body from other physical entities detected in the visual space. As a ground truth, we use the expected position of the robot hand estimated though the forward kinematics model and acquired through the YARP ports. Therefore, if the perceptual system detects an entity categorized as a robot part at its expected position, we consider that categorization is correct.

The categorization procedure has shown to identify the robot's hands within the first 8 seconds of their motion in the visual field. In average, the self-identification rate was about 98.2% during the robot's motor activity.

The implemented self-identification algorithm was tested with several appearances of the robot hands. The appearance of the robot hand was changed by wearing colored gloves, as demonstrated in Fig.8.5. All actions performed with the initial appearance of the robot hands were repeated, while wearing each type of gloves. The obtained self-identification rate with each appearance of the robot hand is reported in Table 8.1. From our experiments, the system has shown to be independent on the robot hand appearance :

– in case of wearing the blue gloves, the average self-identification rate was about 98.1%,
– in case of wearing the pink gloves, the average self-identification rate was about 98.0%.

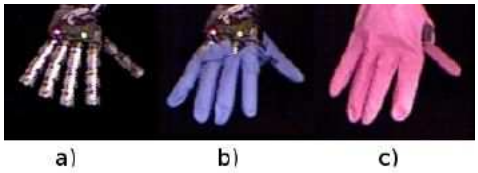


FIGURE 8.5 – Changing the appearances of iCub hands : a) initial appearance, b) wearing the blue glove, c) wearing the pink glove.

TABLE 8.1 – Average self-identification rate

Robot hand appearance	Self-identification rate,%
Initial appearance of robot hands	98.2
Robot hands with the blue glove	98.1
Robot hands with the pink glove	98.0

The obtained self-identification rate slightly varies between different appearances of the robot hand, that can be explained by the size of the worn gloves and the similarity of their appearances with appearances of other non-robot entities. Slightly lower self-identification rate obtained in case of the pink glove can be caused by a large size of the glove that reduces the visibility of a hand motion. Moreover, both types of gloves simplify the appearance of the robot hand with respect to its initial appearance that has a lot of visual features near finger joints. The simplification of the robot hand appearance can result in increasing similarity to other non-robot entities appearances, that decreases the accuracy of entities recognition and thus, decreases the self-identification rate.

8.2.2 Evaluation of categorization of objects and human parts

Once the robot's body is identified, the robot starts to explore its close environment though interactive actions directed to physical entities localized in the visual space at a reachable distance. Among interactive actions, the robot performs simple actions, like *push*, or manipulations, like *TakeLiftFall* and *TakeObserve* described in Section 8.1.1. In this experiment, both the robot and its human partner interact with objects. Interactive actions of the robot are aimed to move or grasp an entity and to verify its displacement in the visual field. Further, in case of a successful grasp, manipulation is used to explore an object appearance, that will be analyzed in the following section. During interaction with entities, the perceptual system continuously gathers the statistics on visual motion of entities and categorize

them. The robot's ability to distinguish between manipulable objects and human parts is evaluated a posteriori by labeling entities with their correct categories.

We have evaluated the robot's ability to categorize human parts and 20 objects on a sequence of about 30000 images recorded during one hour. In this experiment, manipulations of all objects by the human partner last about 20 minutes, and the robot's actions and manipulations of objects last about 40 minutes. Each object has been successfully identified as an object category during the first 5-10 seconds of interaction with it, as shown in Fig.8.6. Human parts have been categorized correctly in 89% of images.

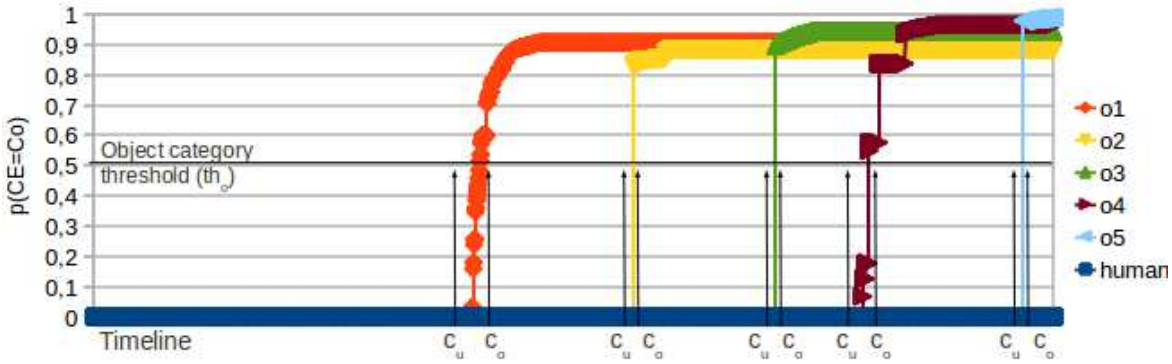


FIGURE 8.6 – Categorization of five objects based on the probability $p(c_{E_i} = c_o)$ of being an object category c_o ; each object appears in the timeline as an unknown category c_u, and once it is categorized, its category is marked in the timeline (in this case, the category c_o)

8.3 Evaluation of interactive object learning

Once the robot is able to categorize physical entities detected in the visual space, the robot starts to explore objects by interacting with them. The ability to categorize entities allows to discriminate the robot hands from objects, when they are seen alone or moving together as a single proto-object. Moreover, entity categorization allows to identify object views inside the robot hands, while the object is grasped and thus, allows to learn the object through manual exploration.

In case of a successful grasp of an object entity, the robot explores it based on one of manipulations, like *TakeLiftFall* or *TakeObserve*, described in Section 8.1.1. Examples of images with the manipulation *TakeObserve* are shown in Fig.8.7. In total, the manual exploration of 20 objects lasts about 30 minutes for each type of manipulation and contains about 15000 images. Manual object exploration is aimed at gathering maximum information about an object's appearance in order to improve its representation model acquired during observation.

After the manual exploration of all objects, the learning performance is evaluated based on the database described in Section 5.1.3. The recognition rate based on major and pure

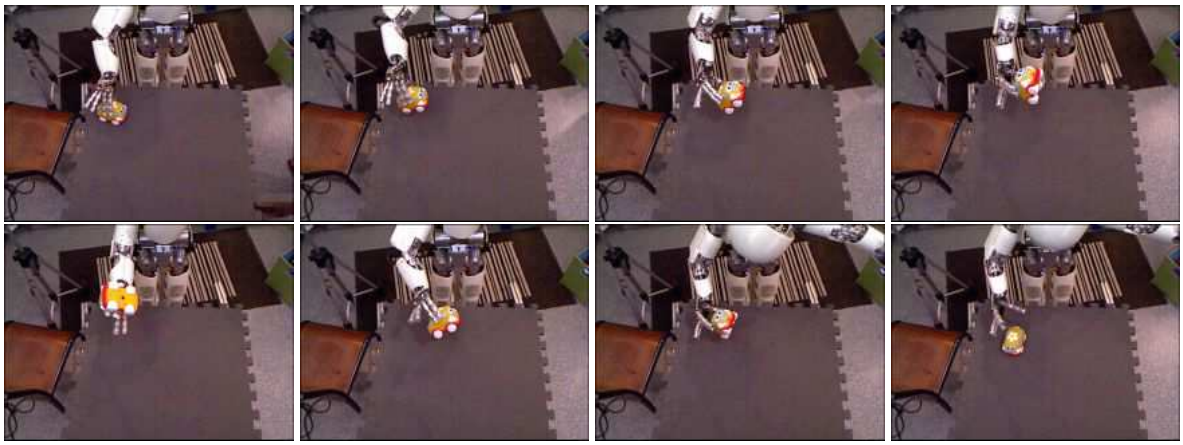


FIGURE 8.7 – Examples of images, where an object is explored based on *TakeObserve* manipulation : the object is grasped, lifted, rotated, approached at a closer distance, turned around to observe its different perspectives, and posed back to the table

labels, the number of pure entities and views associated with each object, are reported in Table 8.2, where all obtained values are presented in pairs comparing the results of learning though interaction (and cleaning dictionaries described in Section 7.3.2) with the results obtained during learning through observation presented in Section 5.3.

For most of objects, the interactive learning results in increasing of the recognition rate based on a major label with respect to the results of learning through observation, and this improvement is shown in Fig.8.8 with respect to the final recognition rate based on pure labels. The recognition rate based on pure labels remains nearly stable, as we have obtained during learning through observation. The changes of recognition rate based on both labels can be explained by the concept of the interactive learning procedure. In our algorithm, interactive learning is aimed at updating the model of a grasped entity that improves the informativeness of the grasped entity model. Thus, interactive learning can improve the major entity label, while leaving other pure labels without significant changes.

Learning through interaction enhances the objects models, since the number of views inside models increases. For objects whose appearances significantly vary between perspectives, the manual exploration is especially useful. While manipulating an object, the perceptual system integrates all recognized views into the representation model of the grasped entity, that enhances the model and makes it more complete. Moreover, the system creates new views that correspond to previously unknown object perspectives. From our experiments, the learning through interaction results in enhancement of the models that correspond to major labels of the following objects O_1, O_2, O_3, O_8, O_9, and O_{11}. The improvements of several models (in particularly, views added to these models) are shown in Fig.8.9.

As we discussed in Section 5.3.2, learning through observation results in association of a single real object with several physical entities. However, interactive learning allows to con-

TABLE 8.2 – The results obtained by learning through interaction : all values are presented in pairs comparing results of learning through interaction / with respect to learning through observation

Object	Recognition rate based on pure labels,%	Recognition rate based on a major label,%	Number of associated pure entities	Number of views in a major label	Number of associated pure views
O_1	96/96	45/33	4/6	3/2	9/9
O_2	100/90	92/78	3/3	4/3	8/6
O_3	98/96	82/40	3/6	3/1	5/6
O_4	58/60	44/44	1/3	2/2	2/4
O_5	91/41	52/41	3/1	2/2	3/2
O_6	63/63	40/40	4/7	1/1	4/7
O_7	60/60	60/52	1/2	1/1	1/2
O_8	100/100	86/50	3/4	2/1	4/4
O_9	89/96	33/32	4/8	2/1	5/9
O_{10}	80/80	23/22	5/8	1/1	5/8
O_{11}	84/84	35/23	5/6	2/1	6/6
O_{12}	87/87	63/47	2/4	1/1	2/4
O_{13}	100/100	100/97	1/2	2/2	2/2
O_{14}	87/87	51/38	4/7	1/1	4/7
O_{15}	94/90	41/25	3/5	1/1	3/5
O_{16}	100/100	100/100	1/1	1/1	1/1
O_{17}	100/100	100/80	1/2	2/2	2/2
O_{18}	100/100	100/99	1/2	1/1	1/2
O_{19}	100/100	100/99	1/1	2/2	2/2
O_{20}	83/83	76/76	2/4	1/1	2/4
Mean	88.5/85.7	66.2/55.8	2.6/4.1	1.8/1.4	3.6/4.6

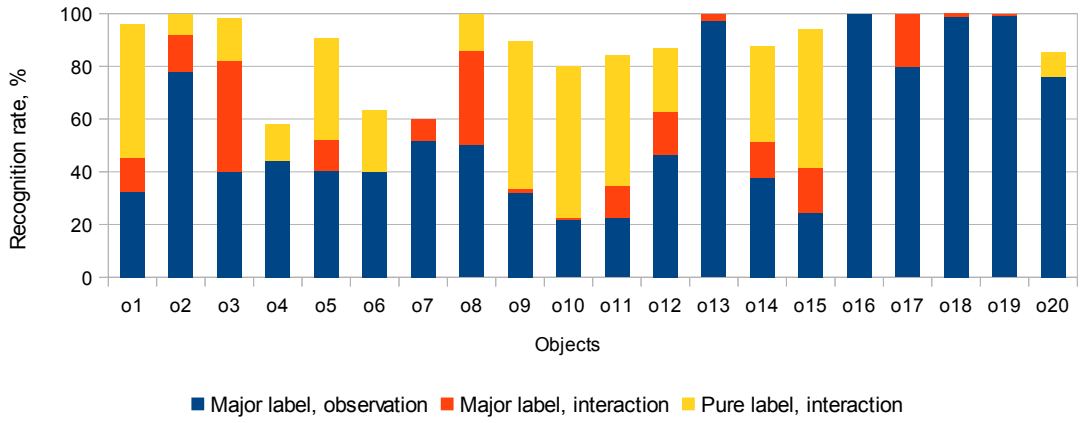


FIGURE 8.8 – Improvement of the object recognition rate : the recognition rate (based on major labels) obtained through observation is shown by the blue color, the improvement of this recognition rate during interactive learning is shown by the orange color, and the final recognition rate (based on pure labels) is shown by yellow color

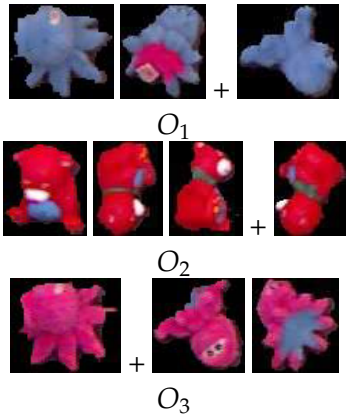


FIGURE 8.9 – The representation models of the major entities that correspond to the objects O_1, O_2, and O_3 (each model with its views is illustrated in one line), where the views added to the models during interactive learning are shown after the + sign

solidate the knowledge about the objects within major entities and to decrease the number of entities associated with objects. The total number of entities and views decreases mostly due to cleaning dictionaries described in Section 7.3, that is performed after manipulations with objects. The number of entities obtained during learning through observation, during learning though interaction, and after cleaning dictionaries is reported in Table 8.3. The number of entities grows a bit during interaction and then, it significantly decreases after cleaning dictionaries.

TABLE 8.3 – The comparison of results obtained during learning though observation and during interactive learning with and without cleaning dictionaries

	Recognition rate based on pure labels,%	Recognition rate based on a major label,%	Total number of pure entities
Learning through observation	85.7	55.8	81
Learning through interaction without cleaning dictionaries	88.4	54.2	85
Learning through interaction with cleaning dictionaries	88.5	66.2	52

We also compare the average recognition rate based on both labels during learning through observation, during learning though interaction, and after cleaning dictionaries. From our results, the interactive learning with cleaning dictionaries not only makes the robot's knowledge more coherent and removes noisy entities but also leads to significant improvement of object recognition rate based on major labels.

8.4 Evaluation of curiosity-driven object exploration

Active object exploration is integrated with the social guidance and the robot's curiosity, as described in the curiosity-driven exploration approach described in Section 7.4. The performance of the curiosity-driven object exploration was evaluated in [Nguyen et al., 2013]. In our experiment, the robot actively explores 5 objects (shown in Fig.8.10) for about one hour. During the learning process, the curiosity mechanism monitors the learning progress and notifies decisions. The decisions are determined by a triple *[object, action, actor]*, where the robot communicates

- the *object* needed to be explored,
- the *action* needed to be performed,
- the *actor* who performs the action.

As an actor, the robot can ask a human partner to perform an action with an object, or the robot performs the action by itself. If an actor is the robot, the repertoire of possible actions

is described in Section 8.1. If the robot asks a human partner to perform an action, the action is either the presentation of a new object, or manipulation of the object that can provide a view unpredictable for the robot.

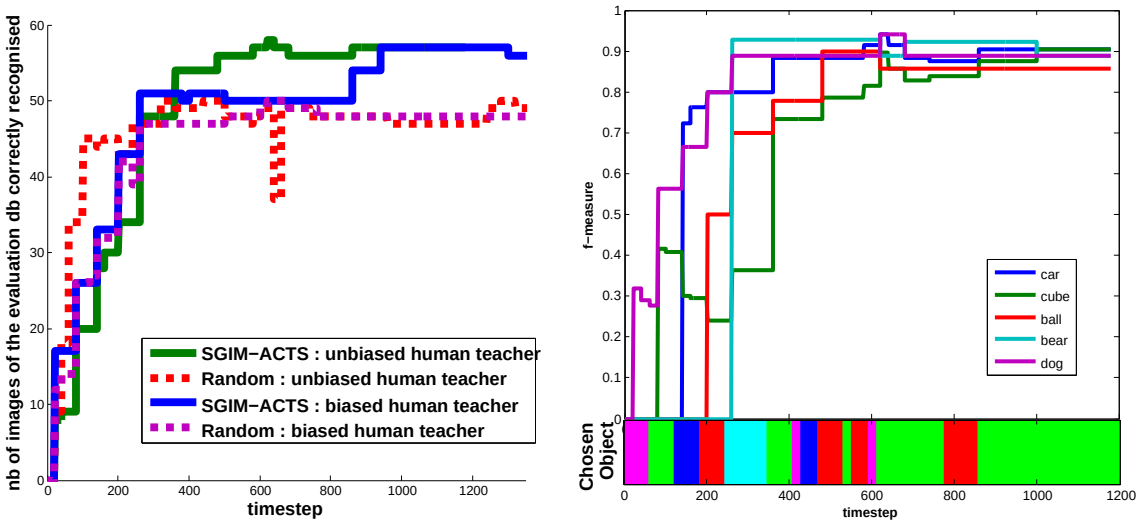
FIGURE 8.10 – Objects used for curiosity-driven exploration

The efficiency of social guidance strongly depends on a human partner. A human partner demonstrating objects properly, showing each time one of previously unseen object perspectives, is considered as "unbiased" teacher. In contrast, a human partner demonstrating objects each time from the same perspective, is considered as "biased" teacher. The curiosity-driven exploration of objects is evaluated with both types of teachers, and the results are compared with a random exploration strategy based on random choice of objects and actions. The average object recognition rate (based on major labels) estimated during the learning process is shown in Fig.8.11a. The curiosity-driven object exploration has shown a higher learning progress compared to the random exploration strategy. Thus, we consider the curiosity-driven behavior as advantageous for efficient and continuous exploration of the robot's environment.

We also analyze how well the robot distinguishes each object based on f-measure estimated as the harmonic mean of precision and recall. The f-measure and the objects manipulated at each timestamp are shown in Fig.8.11b.

From our experiments, the robot manipulates the object *cube* more often than other objects, and especially when the learning progress increases. Among all objects, the *cube* is most complex object, since its views vary significantly (see Fig.8.12). The frontal view of the *cube* consists of four small components, while lateral views consist of two small components with different colors that change depending on the object's side. Moreover, some views of the *cube* are rather similar to views of other objects, like the object *bus*.

Manipulation of complex objects provides more information than manipulation of simple objects, since the appearance of a complex object can change significantly depending on the viewing angle resulted from the performed action. Our experiment has shown that more complex object are requested more often. For example, the robot has spent 54% of its time by exploring the *cube*. Also it shows, that our perceptual system adapts correctly to the objects complexities and represents objects by the reasonable number of views that allows to efficiently recognize each object. Moreover, the system is able to provide a good measure of the quality (described in Section 7.4) of objects representation models.



a) b)

FIGURE 8.11 – Curiosity-driven object learning : a) the average recognition rate obtained at different stages of the learning process for with two different ; the results are computed for both "biased" and "unbiased" teachers using the curiosity-driven exploration strategy and random exploration strategy ; b) f-measure with respect to time. At the bottom of the plot, the manipulated object is shown at each timestamp [Nguyen et al., 2013]

FIGURE 8.12 – Views of the object *cube*

8.5 Conclusion

In this chapter, we have evaluated the performance of the interactive perceptual system. The system has shown a good categorization performance while identifying parts of the robot's own body, human parts, and manipulable objects. The robot's hand was correctly categorized within the first 8 seconds of its motion in the visual field. The self-identification algorithm has shown to be independent on the robot's appearance and on behavior of the robot's motion. Each object was correctly categorized during the first 5-10 seconds of interaction with it.

While learning object through interaction, we have achieved a higher learning performance compared to simple observation of objects evaluated in Chapter 5. Manual exploration was especially useful for objects, whose appearances significantly vary between perspectives. Interactive object learning has shown to improve the knowledge about objects appearances and the informativeness of their representation models previously gathered during learning through observation. From our experiment, models of manipulated objects have been updated with both recognized view and newly created views that correspond to previously unknown object perspectives.

Our system is also able to provide a good assessment of the quality of object models, that has been successfully used to guide the choice of action toward more complex objects for improving knowledge, during the curiosity-driven exploration.

CHAPITRE 9

Conclusion and discussions

In this work we have developed a perceptual approach that enables a humanoid robot to explore its close environment in an interactive scenario, following the context of developmental learning. Taking inspiration from infants development, the robot learns through interaction with a human partner and its own interactive actions. The perceptual system is able to detect physical entities in the visual space, to synthesize the acquired information about their appearances into hierarchical representation models, and to categorize entities into parts of the robot's body, human parts, and manipulated objects. Based on these categories, the robot focuses on manual object exploration aimed at acquiring maximum knowledge about overall appearances of objects and improving their representation models.

9.1 Summary of the approach

The implemented algorithm does not require image databases, predefined objects appearance, or specialized detectors, such as face/skin/skeleton detectors. The proposed approach is based on online incremental and continuous learning. The perceptual system autonomously detects physical entities based on the concept of proto-objects and visual attention. The proto-objects appearances are characterized not as a simple collection of low-level features but rather integrating low-level features and their relative locations into more complex mid-features and further, into multi-view representation models of entities appearances. The multi-view model allows to overcome possible changes of the object's appearance emerging from different viewing angles, scales, and varying lightning conditions. Incremental learning allows to learn new entities and their views over time without knowing the number of entities in advance. As soon as new data are available, they are easily added to the visual memory without re-processing all known data. The entity recognition is based either on tracking, or on its appearance. Entities can be recognized in both cases : when they are isolated and when several entities move together as a single proto-object, that often occurs during an object manipulation.

Physical entities are categorized into parts of the robot's body, human parts, and manipulated objects based on motion behavior and mutual information between the sensory and proprioceptive data. The categorization algorithm does not require predefined appearance of the robot, or predefined pattern of motion. The robot self-identification algorithm is independent on the robot hand's appearance, that was tested by changing the initial appearance of the hand by wearing colored gloves. The system is able to visually identify the robot's hand within few seconds of its motion, and manipulable objects are distinguished within few seconds of interaction with them. The recognition and categorization of entities enable to discriminate the robot's hands from objects, when they present alone or move together as a single proto-object. Moreover, entity categorization allows to identify object's views inside the robot hand, while the object is grasped and thus, allows to learn the object's appearance during manipulations.

Interactive object exploration allows to significantly improve the knowledge about objects appearances. In our algorithm, the information about an object is acquired in between interactive actions and also during manipulations, while the object is grasped. All information acquired about the object's appearance, such as recognized views and newly created views, are integrated into the representation model of the object. According to our experiments, the interactive learning improves informativeness and quality of representation models due to the increasing number of views and suppression of noisy entities. The ability to recognize and categorize connected entities prevents noisy updates of manipulated objects' models. Thus, the average recognition accuracy increases after learning through interaction.

9.2 Discussion and current limitations

Based on experimental evaluation of the proposed perceptual system, we reveal several issues limiting the capacity of the system, and thus, we describe how these issues can be resolved.

9.2.1 Dependence on RGB-D sensor

The perception begins from input data acquisition from a visual sensor. In our system, the visual input is taken from an external RGB-D sensor. The system can also process the input from an embedded robot camera or any web-camera, however, in this case, objects are segmented without using depth data, and their boundaries are less accurate in case of complex scenes and backgrounds. The input from two robot cameras can be processed as stereo vision, though it increases the processing time. The limitation of an external RGB-D sensor is the absence of its control; it does not move together with the robot's body, and it does not allow to control the robot's gaze in order to focus on a particular area of the environment. The possible solution would be to fix the RGB-D sensor above the robot's head

and to adapt the system to changing visual and depth data during the camera motion, that should not take a lot of work, as discussed below.

9.2.2 Dependence on a static platform

In case of adaptation of the proposed perceptual system to a moving platform, we would need to deal with a great optical flow all the time. When the robot moves, the whole scene is moving. From one point, it requires an enhanced processing, but from another point, it gives an additional cue for proto-object detection. Since all points of the background move with a nearly same speed, a proto-object can be detected based on the difference between the speed of its points compared to the speed of background points. Once proto-objects are detected, the system processes only segmented regions, like in our work.

9.2.3 Motion-based object detection

Our proto-object detection algorithm is based on a bottom-up saliency mechanism taking into account visual motion. Thus, our algorithm allows to detect a proto-object, when it moves. Once a proto-object is detected, it can be tracked even, when it is static. However, the system can not make a hypothesis about the existence of a proto-object in initially static areas. In order to resolve this issue, we need a mechanism that re-projects a saliency in a top-down way. For example, already known objects can be found based on their appearances. An existence of unknown objects can be revealed from other saliency aspects in addition to motion, by modeling more precisely selective attention mechanism of human vision. An existence of unknown objects can be also revealed by moving a camera or the robot in order to detect proto-objects based on optical flow, as discussed in the previous paragraph.

9.2.4 Growth of processing time

Our system learns objects continuously, that makes it subjected to continuous growth of dictionaries. We already clean the dictionaries of entities and views, and we see its positive effect and improvement of object recognition. It would be advantageous to clean also the feature dictionaries by filtering out less representative features, while keeping features that repeat often. This procedure will stabilize the dictionaries growth and moreover, objects models based on only representative features could result in improving final object recognition.

The experiments reported in this thesis are based on 20 objects. The increasing number of learning objects or images augments the system's processing time. The processing time grows mostly because of dictionaries growth. The issue can be resolved by stabilizing the growth of dictionaries or using static dictionaries. Smaller dictionaries should improve the

processing time, since the search will be faster. Moreover, in case of less visual features, objects models will be smaller, thus, the learning and recognition procedures should be faster. As reported in Section 5.2.2, we could also remove mid-features based on color triples, that would allow to gain some processing time without significant loss of the object recognition performance. Furthermore, the processing time can be improved by optimizing the code or some algorithms. If images can be processed faster, we can process more images per second that should improve tracking, help to filter frequently repeated information, and finally improve learning performance.

9.2.5 Limitations linked to the sensor resolution

Further, if we need to process much more objects in open-ended scenarios, we think about improving the resolution of the visual sensor. Indeed, in the current experiments, the utility of the SURF features is already limited by the apparent size of the objects in images, which reduces the number of detectable features. For learning and recognizing objects that are farther away, the current resolution would not be sufficient. We could also enhance some processing steps in our system. For example, feature extraction can be replaced by deep learning, that would adapt the features to the sensor and to the environment, and could improve performances.

9.2.6 Limitations of the learning approach

In the performed experiments, our learning approach performed rather well, but if more objects have to be learned, we could add a discriminative approach to improve the learning performance. We could imagine an offline consolidation phase inspired by the role of human dreams that consolidate knowledge into a long-term memory, and the views gathered by our approach could be used as a database to train a discriminative algorithm.

9.2.7 Limitations to 2D appearance

In real world surrounding us, most of physical objects are 3-dimensional, although, we work with 2D objects appearances learning them as views and collecting them into multi-view entities models. In case of using a RGB-D sensor, the acquired depth data can be used for estimation a 3D shape information and integration it into an entity model. This 3D shape information could improve the recognition performance and also give a ground for learning affordances and categories based on an object's utility and behavior during interaction.

9.2.8 Multiple entities associated to the same object

Our perceptual system relies on unsupervised learning that results in associating some objects with several physical entities instead of just one. This is quite common for unsu-

pervised systems, as no ultimate supervision enforces the unity of the model. In order to overcome this issue, we could add a supervised learning layer that would allow to associate several entities to a knowledge. It can be performed during natural interaction with a human partner, similar to children development, when they learn through social interaction. For example, in ambiguous situations, the robot could ask questions to its human partner, or the partner could provide a feedback during the learning process. The human partner could also pronounce the object name, while showing it to the robot, that would help to associate the visual information with the object's identity.

9.2.9 Learning objects instances

Our work is aimed at learning object instances, though a lot of behaviors are linked to categories characterizing objects in terms of their purposes (like bottles, cans, cups, hammers) or in terms of their properties (like red objects, spherical objects, rollable objects). Our work could be extended to learn object categories based on extracted features. Objects with similar appearances could be classified into categories based on similarity of their visual features or representation models. Objects could be also grouped based on their behavior during interaction with them. Another possible solution consists of initial learning of objects classes but not objects instances, when all acquired data are directly integrated into models of objects classes.

9.3 Future work

The implemented system and the achieved results open a lot of possibilities for future work. In this section, we suggest some directions for future research towards open-ended learning for service robots.

9.3.1 Feedback of the knowledge on low-level processing

An interesting extension of this work would be the further integration of the robot's experience gathered through interaction with the environment into its ability of knowledge acquisition. In infants, the development of capabilities to manipulate objects effects on infants perception, especially visual saliency, and attention. It would be advantageous to adapt this influence of interaction on the robot's perception. Once the robot has explored an object manually at a close scale, it has acquired more knowledge about the importance of its visual features, like the importance for interaction, for example a successful grasp, or the importance for correct recognition of the same object from a far distance. This experience gathered through interaction could provide a feedback to perceptual system, for example by changing its attention model, notion of saliency, or extracted visual features.

9.3.2 Integration of audio information

It would be interesting to extend the aspect of social interaction by integrating the audio information in our system. While seeking the multi-modality of learning and taking inspiration from infant-directed interaction, when an adult names an object while showing it to an infant, we could learn about objects not only from visual data but also from audio information. Audio could help grouping together entities that have been associated to the same name, and in the reverse direction, having a recognized object could help to segment its name in the audio stream. This can be viewed as a step towards the development of common language between the robot and its human partner, where the robot is able to learn objects associated with any names that its user would like to use, and it will help to improve object recognition in more complex interactive scenarios.

9.3.3 Adaptation to a mobile platform

Finally, we imagine an extension of our approach to a robot moving around autonomously and discovering its environment on its way and while interacting with people. Our general concept should be robust in case of this application due to multiple filtering of the visual information and synthesizing it throughout different stages. The system should not be overloaded thanks to focusing on meaningful information, like proto-objects that can be continuously detected based on saliency, similar to the way human vision do. The proto-objects appearances could be characterized by more representative set of features, for example using deep learning. Learning about proto-objects should be improved taking advantage from possible sensors, like visual, audio, tactile, if available. Exploring the environment would be a crucial problem in such system, and therefore, it should rely strongly on a curiosity-driven behavior and social guidance in order to learn useful objects.

Bibliographie

[Achanta et al., 2010] Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., and Süsstrunk, S. (2010). Slic superpixels. *École Polytechnique Fédéral de Lausssanne (EPFL), Tech. Rep*, 149300. 35

[Ai et al., 2012] Ai, L., Yu, J., and Guan, T. (2012). Spherical soft assignment : improving image representation in content-based image retrieval. In *Advances in Multimedia Information Processing–PCM 2012*, pages 801–810. Springer. 38

[Aldavert et al., 2010] Aldavert, D., Ramisa, A., López de Mántaras, R., Toledo, R., et al. (2010). Real-time object segmentation using a bag of features approach. *Artificial Intelligence Research and Development*, pages 321–329. 35, 37

[Asada et al., 2009] Asada, M., Hosoda, K., Kuniyoshi, Y., Ishiguro, H., Inui, T., Yoshikawa, Y., Ogino, M., and Yoshida, C. (2009). Cognitive developmental robotics : A survey. *IEEE Trans. Autonomous Mental Development*, 1(1). 18

[Baillargeon, 1999] Baillargeon, R. (1999). Young infants' expectations about hidden objects : A reply to three challenges. *Developmental Science*, 2(2) :115–132. 16

[Bay et al., 2008] Bay, H., Ess, A., Tuytelaars, T., and Van Gool, L. (2008). Speeded-up robust features (surf). *Comput. Vis. Image Underst.*, 110 :346–359. 31, 32

[Beale et al., 2011] Beale, D., Iravani, P., and Hall, P. (2011). Probabilistic models for robot-based object segmentation. *Robotics and Autonomous Systems*, 59 :1080–1089. 28, 101

[Benois-Pineau et al., 2012] Benois-Pineau, J., Precioso, F., and Cord, M. (2012). *Visual indexing and retrieval*. Springer. 31, 33, 34

[Berlyne, 1960] Berlyne, D. E. (1960). *Conflict, arousal, and curiosity*. McGraw-Hill Book Company. 20

[Bertenthal and Fischer, 1978] Bertenthal, B. I. and Fischer, K. W. (1978). Development of self-recognition in the infant. *Developmental Psychology*, 14(1) :44. 18

[Bigün, 1990] Bigün, J. (1990). A structure feature for some image processing applications based on spiral functions. *Computer Vision, Graphics, and Image Processing*, 51(2) :166–194. 33

[Borenstein and Ullman, 2002] Borenstein, E. and Ullman, S. (2002). Class-specific, top-down segmentation. In *Computer Vision-ECCV 2002*, pages 109–122. Springer. 37

[Bornstein et al., 1976] Bornstein, M. H., Kessen, W., and Weiskopf, S. (1976). Color vision and hue categorization in young human infants. *Journal of Experimental Psychology : Human Perception and Performance*, 2(1) :115. 15

[Bouchard and Triggs, 2005] Bouchard, G. and Triggs, B. (2005). Hierarchical part-based visual object categorization. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 710–715. IEEE. 39, xiii

[Bouguet, 2001] Bouguet, J.-Y. (2001). Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm. *Intel Corporation*, 5. 51

[Boureau et al., 2010] Boureau, Y.-L., Bach, F., LeCun, Y., and Ponce, J. (2010). Learning mid-level features for recognition. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2559–2566. IEEE. 36

[Brand et al., 2002] Brand, R. J., Baldwin, D. A., and Ashburn, L. A. (2002). Evidence for 'motionese' : modifications in mothers' infant-directed action. *Developmental Science*, 5(1) :72–83. 19

[Browatzki et al., 2012] Browatzki, B., Tikhanoff, V., Metta, G., Bulthoff, H., and Wallraven, C. (2012). Active object recognition on a humanoid robot. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 2021–2028. 28, 100, 101, 102

[Burger and Burge, 2008] Burger, W. and Burge, M. J. (2008). *Digital image processing*. Springer. 31, 34, 42

[Carbonetto et al., 2008] Carbonetto, P., Dorkó, G., Schmid, C., Kück, H., and De Freitas, N. (2008). Learning to recognize objects with little supervision. *International Journal of Computer Vision*, 77(1-3) :219–237. 35

[Cauwenberghs and Poggio, 2001] Cauwenberghs, G. and Poggio, T. (2001). Incremental and decremental support vector machine learning. *Advances in neural information processing systems*, pages 409–415. 42

[Chandrashekhariah et al., 2013] Chandrashekhariah, P., Spina, G., and Jochen, T. (2013). Let it learn : a curious vision system for autonomous object learning. In *Proceedings of the International Conference on Computer Vision Theory and Applications (VISAPP)*. 21, 28

[Cohen and Cashon, 2003] Cohen, L. B. and Cashon, C. H. (2003). Infant perception and cognition. *Handbook of psychology*. 15

[Comaniciu and Meer, 2002] Comaniciu, D. and Meer, P. (2002). Mean shift : A robust approach toward feature space analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(5) :603–619. 29

[Crandall and Huttenlocher, 2006] Crandall, D. J. and Huttenlocher, D. P. (2006). Weakly supervised learning of part-based spatial models for visual object recognition. In *Computer Vision-ECCV 2006*, pages 16–29. Springer. 38, 39, xiii

[Csurka et al., 2004] Csurka, G., Dance, C., Fan, L., Willamowski, J., and Bray, C. (2004). Visual categorization with bags of keypoints. In *Workshop on statistical learning in computer vision, ECCV*, volume 1, page 22. 31, 37, 40, 42

[Dickscheid et al., 2011] Dickscheid, T., Schindler, F., and Förstner, W. (2011). Coding images with local features. *Int. J. Comput. Vision*, 94 :154–174. 31, 32, 35, xiii

[Edelman, 1997] Edelman, S. (1997). Curiosity and exploration. *Retrieved May*, 11 :2005. 20, 119

[Fei-Fei et al., 2007] Fei-Fei, L., Iyer, A., Koch, C., and Perona, P. (2007). What do we perceive in a glance of a real-world scene ? *Journal of Vision*, 7(1). 12

[Felzenszwalb and Huttenlocher, 2004] Felzenszwalb, P. F. and Huttenlocher, D. P. (2004). Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2) :167–181. 35

[Fergus et al., 2005] Fergus, R., Perona, P., and Zisserman, A. (2005). A sparse object category model for efficient learning and exhaustive recognition. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 380–387. IEEE. 38

[Fergus et al., 2003] Fergus, R., Perona, P., Zisserman, A., and K, O. P. U. (2003). Object class recognition by unsupervised scale-invariant learning. In *Conf. on Computer Vision and Pattern Recognition*, pages 264–271. 38

[Fiala, 2005] Fiala, M. (2005). Artag, a fiducial marker system using digital techniques. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 590–596. 28

[Filliat, 2007] Filliat, D. (2007). A visual bag of words method for interactive qualitative localization and mapping. In *IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 3921–3926. 37, 42, 58

[Fischler and Elschlager, 1973] Fischler, M. A. and Elschlager, R. A. (1973). The representation and matching of pictorial structures. *Computers, IEEE Transactions on*, 100(1) :67–92. 38, 39, xiii

[Fitzpatrick and Metta, 2003] Fitzpatrick, P. and Metta, G. (2003). Grounding vision through experimental manipulation. *Philosophical Transactions of the Royal Society of London. Series A : Mathematical, Physical and Engineering Sciences*, 361(1811) :2165–2185. 21

[Fitzpatrick et al., 2008] Fitzpatrick, P., Needham, A., Natale, L., and Metta, G. (2008). Shared challenges in object perception for robots and infants. *Infant and Child Development*, 17(1) :7–24. 19, 21

[Förstner, 1994] Förstner, W. (1994). A framework for low level feature extraction. In *European Conf. on Computer Vision (ECCV)*, pages 383–394, London, UK. Springer-Verlag. 32, 33

[Förstner et al., 2009] Förstner, W., Dickscheid, T., and Schindler, F. (2009). Detecting interpretable and accurate scale-invariant keypoints. In *IEEE Int. Conf. on Computer Vision*, pages 2256–2263. 32

[Fritsch et al., 2002] Fritsch, J., Lang, S., Kleinehagenbrock, M., Fink, G. A., and Sagerer, G. (2002). Improving adaptive skin color segmentation by incorporating results from face detection. In *IEEE Int. Workshop on Robot and Human Interactive Communication*, pages 337–343. 28

[Gaël and Benoît, 2010] Gaël, G. and Benoît, J. (2010). Eigen v3. http ://eigen.tuxfamily.org. 109

[Gibson, 1988] Gibson, E. J. (1988). Exploratory behavior in the development of perceiving, acting, and the acquiring of knowledge. *Annual review of psychology*, 39(1) :1–42. 19

[Gold and Scassellati, 2005] Gold, K. and Scassellati, B. (2005). Learning about the self and others through contingency. In *AAAI Spring Symposium on Developmental Robotics*. 98, 99

[Goldstein, 2010] Goldstein, E. B. (2010). *Sensation and perception*. Wadsworth Publishing Company. 10, 11, 12, 28, 29, 47, 65

[Grzyb and del Pobil, 2008] Grzyb, B. J. and del Pobil, A. P. (2008). Developing a sense of bodily self. *differentiation*, 1 :2. 99

[Guerin, 2011] Guerin, F. (2011). Learning like a baby : a survey of artificial intelligence approaches. *Knowledge Engineering Review*, 26(2) :209–236. 102

[Haith, 1968] Haith, M. (1968). Visual scanning in infants. *segimal meating of the society for research in Child Development, woncester, mass*. 14

[Han et al., 2011] Han, Z., Ye, Q., and Jiao, J. (2011). Combined feature evaluation for adaptive visual object tracking. *Computer Vision and Image Understanding*, 115(1) :69–80. 33, 34, 35

[Hart and Scassellati, 2010] Hart, J. W. and Scassellati, B. (2010). Robotic self-models inspired by human development. In *Metacognition for Robust Social Systems*. 17

[Hérault, 2010] Hérault, J. (2010). *Vision : Images, Signals and Neural Networks : Models of Neural Processing in Visual Perception*, volume 19. World Scientific. 11, 65

[Holland, 1997] Holland, O. (1997). Grey walter : the pioneer of real artificial life. In *Proceedings of the 5th international workshop on artificial life*, pages 34–44. 98

[Hulse et al., 2009] Hulse, M., McBrid, S., and Lee, M. (2009). Robotic hand-eye coordination without global reference : A biologically inspired learning scheme. In *Development and Learning, 2009. ICDL 2009. IEEE 8th International Conference on*, pages 1–6. IEEE. 98

[Iriki et al., 1996] Iriki, A., Tanaka, M., Iwamura, Y., et al. (1996). Coding of modified body schema during tool use by macaque postcentral neurones. *Neuroreport*, 7(14) :2325. 98

[Itti and Koch, 2001] Itti, L. and Koch, C. (2001). Computational modelling of visual attention. *Nature reviews neuroscience*, 2(3) :194–203. 12, 28, 29, 30, xiii

[Ivaldi et al., 2011] Ivaldi, S., Fumagalli, M., Randazzo, M., Nori, F., Metta, G., and Sandini, G. (2011). Computing robot internal/external wrenches by means of inertial, tactile and f/t sensors : theory and implementation on the icub. In *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on*, pages 521–528. IEEE. 123

[Ivaldi et al., 2012a] Ivaldi, S., Lyubova, N., Gérardeaux-Viret, D., Droniou, A., Anzalone, S. M., Chetouani, M., Filliat, D., and Sigaud, O. (2012a). Perception and human interaction for developmental learning of objects and affordances. In *Humanoids, 2012. Proceedings. 2012 IEEE International Conference on*. IEEE. 122, 123

[Ivaldi et al., 2012b] Ivaldi, S., Nguyen, S. M., Lyubova, N., Alain, D., Vincent, P., David, F., Pierre-Yves, O., and Olivier, S. (2012b). A cognitive architecture for developmental objects learning through active exploration. In *Humanoids, 2012. Proceedings. 2012 IEEE International Conference on*. IEEE. 123

[Jégou et al., 2010] Jégou, H., Douze, M., Schmid, C., and Pérez, P. (2010). Aggregating local descriptors into a compact image representation. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3304–3311. IEEE. 37

[Johnson and Náñez Sr, 1995] Johnson, S. P. and Náñez Sr, J. (1995). Young infant's perception of object unity in two-dimensional displays. *Infant Behavior and Development*, 18(2) :133–143. 15

[Katz et al., 2010] Katz, D., Orthey, A., and Brock, O. (2010). Interactive perception of articulated objects. In *12th International Symposium of Experimental Robotics*, page 1. 28, 101, 102

[Kellman and Arterberry, 2000] Kellman, P. J. and Arterberry, M. E. (2000). *The cradle of knowledge : Development of perception in infancy*. The MIT Press. 15, 16

[Kemp and Edsinger, 2006] Kemp, C. and Edsinger, A. (2006). What can i control ? : The development of visual categories for a robot's body and the world that it influences. In *IEEE Int. Conf. on Development and Learning (ICDL), Special Session on Autonomous Mental Development*. 100, 103, 113

[Kestenbaum et al., 1987] Kestenbaum, R., Termine, N., and Spelke, E. S. (1987). Perception of objects and object boundaries by 3-month-old infants. *British Journal of Developmental Psychology*, 5(4) :367–383. 16

[Kokkinos and Yuille, 2011] Kokkinos, I. and Yuille, A. (2011). Inference and learning with hierarchical shape models. *International journal of computer vision*, 93(2) :201–225. 39

[Kokkinos and Yuille, 2006] Kokkinos, M. and Yuille (2006). Bottom-up and top-down object detection using primal sketch features and graphical models. In *CVPR*. 37

[Kootstra et al., 2007] Kootstra, G., Ypma, J., and de Boer, B. (2007). Exploring objects for recognition in the real word. In *Robotics and Biomimetics, 2007. ROBIO 2007. IEEE International Conference on*, pages 429–434. IEEE. 100, 101, 102

[Kraft et al., 2010] Kraft, D., Detry, R., Pugeault, N., Baseski, E., Guerin, F., Piater, J. H., and Krüger, N. (2010). Development of object and grasping knowledge by robot exploration. *Autonomous Mental Development, IEEE Transactions on*, 2(4) :368–383. 21

[Krüger et al., 2010] Krüger, N., Pugeault, N., Baseski, E., Jensen, L. B. W., Kalkan, S., Kraft, D., Jessen, J. B., Pilz, F., Kjær-Nielsen, A., and Popovic, M. (2010). Early cognitive vision as a front-end for cognitive systems. In *ECCV 2010 Workshop on "Vision for Cognitive Tasks*. 36

[Kyrki and Kragic, 2008] Kyrki, V. and Kragic, D. (2008). Recent trends in computational and robot vision. In *Unifying Perspectives in Computational and Robot Vision*, pages 1–10. Springer. 100

[Lederman and Klatzky, 1987] Lederman, S. J. and Klatzky, R. L. (1987). Hand movements : A window into haptic object recognition. *Cognitive psychology*, 19(3) :342–368. 19

[Lee et al., 2009] Lee, H., Grosse, R., Ranganath, R., and Ng, A. Y. (2009). Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 609–616. ACM. 11

[Lowe, 2004] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60 :91–110. 31, 32, 41

[Mahler, 2000] Mahler, M. S. (2000). *The psychological birth of the human infant : Symbiosis and individuation*. Basic Books. 17

[Marjanovic et al., 1996] Marjanovic, M. J., Scassellati, B., and Williamson, M. M. (1996). *Self-taught visually-guided pointing for a humanoid robot*. From Animals to Animats : Proceedings of. 98

[Matas et al., 2002] Matas, J., Chum, O., Urban, M., and Pajdla, T. (2002). Robust wide baseline stereo from maximally stable extremal regions. In *British machine vision conference*, volume 1, pages 384–393. 29

[Metta and Fitzpatrick, 2003] Metta, G. and Fitzpatrick, P. (2003). Early integration of vision and manipulation. In *Neural Networks, 2003. Proceedings of the International Joint Conference on*, volume 4, pages 2703–vol. IEEE. 98, 99, 100, 101

[Metta et al., 2006] Metta, G., Fitzpatrick, P., and Natale, L. (2006). Yarp : yet another robot platform. *International Journal on Advanced Robotics Systems*, 3(1) :43–48. 79

[Michel et al., 2004] Michel, P., Gold, K., and Scassellati, B. (2004). Motion-based robotic self-recognition. In *Intelligent Robots and Systems (IROS), 2004 IEEE/RSJ International Conference on*, volume 3, pages 2763–2768. IEEE. 98

[Micusik and Kosecka, 2009] Micusik, B. and Kosecka, J. (2009). Semantic segmentation of street scenes by superpixel co-occurrence and 3d geometry. In *IEEE Int. Conf. on Computer Visio*, pages 625–632. 35, 56

[Modayil and Kuipers, 2008] Modayil, J. and Kuipers, B. (2008). The initial development of object knowledge by a learning robot. *Robotics and autonomous systems*, 56(11) :879–890. 21

[Mohan et al., 2013] Mohan, V., Morasso, P., Sandini, G., and Kasderidis, S. (2013). Inference through embodied simulation in cognitive robots. *Cognitive Computation*, pages 1–28. 14

[Nagi et al., 2011] Nagi, J., Ducatelle, F., Di Caro, G. A., Ciresan, D., Meier, U., Giusti, A., Nagi, F., Schmidhuber, J., and Gambardella, L. M. (2011). Max-pooling convolutional neural networks for vision-based hand gesture recognition. In *Signal and Image Processing Applications (ICSIPA), 2011 IEEE International Conference on*, pages 342–347. IEEE. 98

[Natale et al., 2005] Natale, L., Orabona, F., Berton, F., Metta, G., and Sandini, G. (2005). From sensorimotor development to object perception. In *IEEE-RAS Int. Conf. on Humanoid Robots*, pages 226–231. 14, 21, 30, 31, 100, 101, 102

[Needham and Baillargeon, 1998] Needham, A. and Baillargeon, R. (1998). Effects of prior experience on 4.5-month old infants' object segregation. *Infant behavior and development*, 21(1) :1–24. 16

[Nguyen et al., 2013] Nguyen, S. M., Ivaldi, S., Lyubova, N., Droniou, A., Gérardeaux-Viret, D., Filliat, D., Padois, V., Sigaud, O., and Oudeyer, P.-Y. (2013). Learning to recognize objects through curiosity-driven manipulation with the icub humanoid robot. In *Development and Learning, 2013. Proceedings. 2013 International Conference on*. IEEE. 21, 123, 132, 134, xviii

[Nguyen and Oudeyer, 2013] Nguyen, S. M. and Oudeyer, P.-Y. (2013). Active choice of teachers, learning strategies and goals for a socially guided intrinsic motivation learner. *Paladyn*, 3(3) :136–146. 119

[Nister and Stewenius, 2006] Nister, D. and Stewenius, H. (2006). Scalable recognition with a vocabulary tree. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2161–2168. IEEE. 37

[Oakes and Baumgartner, 2012] Oakes, L. M. and Baumgartner, H. A. (2012). Manual object exploration and learning about object features in human infants. In *Development and Learning and Epigenetic Robotics (ICDL), 2012 IEEE International Conference on*, pages 1–6. IEEE. 15, 19

[Oliva and Torralba, 2006] Oliva, A. and Torralba, A. (2006). Building the gist of a scene : The role of global image features in recognition. *Progress in brain research*, 155 :23–36. 12

[Orabona et al., 2005] Orabona, F., Metta, G., and Sandini, G. (2005). Object-based visual attention : a model for a behaving robot. In *Computer Vision and Pattern Recognition-Workshops, 2005. CVPR Workshops. IEEE Computer Society Conference on*, pages 89–89. IEEE. 11, 14, 30

[Oudeyer et al., 2007] Oudeyer, P.-Y., Kaplan, F., and Hafner, V. V. (2007). Intrinsic motivation systems for autonomous mental development. *Evolutionary Computation, IEEE Transactions on*, 11(2) :265–286. 20, 21, 102

[Paletta and Pinz, 2000] Paletta, L. and Pinz, A. (2000). Active object recognition by view integration and reinforcement learning. *Robotics and Autonomous Systems*, 31(1) :71–86. 40, 101, 102

[Paternoster, 2007] Paternoster, A. (2007). Vision science and the problem of perception. In *Cartographies of the Mind*, pages 53–64. Springer. 10

[Piaget, 1999] Piaget, J. (1999). *Play, dreams and imitation in childhood*. Routledge, London. 14, 16, 17

[Prest, 2012] Prest, A. (2012). *Weakly supervised methods for learning actions and objects*. PhD thesis, Eidgenössische Technische Hochschule Zürich (ETHZ). 28

[Pylyshyn, 2001] Pylyshyn, Z. W. (2001). Visual indexes, preconceptual objects, and situated vision. *Cognition*, 80 :127–158. 30

[Rensink, 2000] Rensink, R. A. (2000). Seeing, sensing, and scrutinizing. *Vision research*, 40(10-12) :1469–1487. 12, 13, 14, 30, xiii

[Rochat and Rochat, 2009] Rochat, P. and Rochat, P. (2009). *The infant's world*. Harvard University Press. 17, 18, 98

[Rohlfing et al., 2006] Rohlfing, K. J., Fritsch, J., Wrede, B., and Jungmann, T. (2006). How can multimodal cues from child-directed interaction reduce learning complexity in robots ? *Advanced Robotics*, 20(10) :1183–1199. 18

[Rouanet et al., 2009] Rouanet, R., Oudeyer, P.-Y., and Filliat, D. (2009). An integrated system for teaching new visually grounded words to a robot for non-expert users using a mobile device. In *IEEE-RAS Int. Conf. on Humanoid Robots*, Tsukuba, Japon. 28

[Rudinac et al., 2012] Rudinac, M., Kootstra, G., Kragic, D., and Jonker, P. P. (2012). Learning and recognition of objects inspired by early cognition. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 4177–4184. IEEE. 21, 28, 29, 35

[Russell et al., 2006] Russell, B. C., Freeman, W. T., Efros, A. A., Sivic, J., and Zisserman, A. (2006). Using multiple segmentations to discover objects and their extent in image collections. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 1605–1614. IEEE. 37

[Saegusa et al., 2012] Saegusa, R., Metta, G., and Sandini, G. (2012). Body definition based on visuomotor correlation. *Industrial Electronics, IEEE Transactions on*, 59(8) :3199–3210. 21, 99

[Saegusa et al., 2013] Saegusa, R., Metta, G., Sandini, G., and Natale, L. (2013). Action learning based on developmental body perception. In *IEEE Int. Conf. on Industrial Technology (ICIT)*. 99, 103

[Shi and Tomasi, 1994] Shi, J. and Tomasi, C. (1994). Good features to track. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 593 – 600. 50

[Shih, 2009] Shih, F. Y. (2009). *Image processing and mathematical morphology : Fundamentals and applications*. CRC PressI Llc. 48

[Shotton et al., 2005] Shotton, J., Blake, A., and Cipolla, R. (2005). Contour-based learning for object detection. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 1, pages 503–510. IEEE. 38

[Shotton et al., 2008] Shotton, J., Johnson, M., and Cipolla, R. (2008). Semantic texton forests for image categorization and segmentation. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE. 37

[Siagian and Itti, 2007] Siagian, C. and Itti, L. (2007). Rapid biologically-inspired scene classification using features shared with visual attention. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(2) :300–312. 11, 12, 28, 29

[Sivic et al., 2005] Sivic, J., Russell, B., Efros, A., Zisserman, A., and Freeman, W. (2005). Discovering object categories in image collections. In *Proceedings of the International Conference on Computer Vision*. 41

[Sivic and Zisserman, 2003] Sivic, J. and Zisserman, A. (2003). Video google : Text retrieval approach to object matching in videos. In *Int. Conf. on Computer Vision*, volume 2, pages 1470–1477. 37, 38

[Slater et al., 1991] Slater, A., Mattock, A., Brown, E., and Bremner, J. G. (1991). Form perception at birth : revisited. *Journal of experimental child psychology*, 51(3) :395–406. 15, 16, xiii

[Smith, 1978] Smith, A. R. (1978). Color gamut transform pairs. *SIGGRAPH Comput. Graph.*, 12 :12–19. 58

[Smith and Gasser, 2005] Smith, L. and Gasser, M. (2005). The development of embodied cognition : Six lessons from babies. *Artificial life*, 11(1-2) :13–29. 14, 18, 19, 20

[Southey and Little, 2006] Southey, T. and Little, J. J. (2006). Object discovery through motion, appearance and shape. In *AAAI Workshop on Cognitive Robotics*, page 9. 28

[Tax, 2001] Tax, D. M. (2001). One-class classification. *PhD diss., Delft University of Technology*. 29

[Treisman and Gormican, 1988] Treisman, A. and Gormican, S. (1988). Feature analysis in early vision : Evidence from search asymmetries. *Psychological Review*, 95 :15–48. 12, 13, 36, xiii

[Ude et al., 2008] Ude, A., Omrčen, D., and Cheng, G. (2008). Making object learning and recognition an active process. *International Journal of Humanoid Robotics*, 5(02) :267–286. 42, 100, 101, 102

[Ullman, 1998] Ullman, S. (1998). Three-dimensional object recognition based on the combination of views. *Cognition*, 67(1) :21–44. 40

[Van de Walle et al., 2000] Van de Walle, G. A., Carey, S., and Prevor, M. (2000). Bases for object individuation in infancy : Evidence from manual search. *Journal of Cognition and Development*, 1(3) :249–280. 16

[van Hoof et al., 2012] van Hoof, H., Kroemer, O., Amor, H. B., and Peters, J. (2012). Maximally informative interaction learning for scene exploration. In *Intelligent Robots and Systems (IROS), 2012 IEEE International Conference on*. 100, 101

[Viola and Jones, 2004] Viola, P. and Jones, M. J. (2004). Robust real-time face detection. *Int. J. Comput. Vision*, 57 :137–154. 28

[Volkmann and Dobson, 1976] Volkmann, F. C. and Dobson, M. V. (1976). Infant responses of ocular fixation to moving visual stimuli. *Journal of Experimental Child Psychology*, 22(1) :86–99. 15

[Vyshedskiy, 2009] Vyshedskiy, A. (2009). On the origin of the human mind. *scientific american*. 10, xiii

[Walther and Koch, 2006] Walther, D. and Koch, C. (2006). Modeling attention to salient proto-objects. *Neural Networks*, 19(9) :1395–407. 11, 12, 14, 28, 30, 31, xiii

[Weng et al., 2001] Weng, J., McClelland, J., Pentland, A., Sporns, O., Stockman, I., Sur, M., and Thelen, E. (2001). Autonomous mental development by robots and animals. *Science*, 291(5504) :599–600. 20

[Wersing et al., 2007] Wersing, H., Kirstein, S., Götting, M., Brandl, H., Dunn, M., Mikhailova, I., Goerick, C., Steil, J. J., Ritter, H., and Körner, E. (2007). Online learning of objects in a biologically motivated visual architecture. *Int. J. Neural Systems*, 17(4) :219–230. 28

[Yang et al., 2008] Yang, L., Jin, R., Sukthankar, R., and Jurie, F. (2008). Unifying discriminative visual codebook generation with classifier training for object category recognition. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE. 39, 42

[Zhou et al., 2011] Zhou, K., Richtsfeld, A., Zillich, M., Vincze, M., Vrecko, A., and Skocaj, D. (2011). Visual information abstraction for interactive robot learning. In *Advanced Robotics (ICAR), 2011 15th International Conference on*, pages 328–334. IEEE. 14, 28

[Zhu et al., 2000] Zhu, X., Yang, J., and Waibel, A. (2000). Segmenting hands of arbitrary color. In *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*, pages 446–453. IEEE. 28

Table des figures