Learning of Multi-Dimensional, Multi-Modal Features for Robotic Grasping

A dissertation presented by

Renaud Detry

Directed by

Prof. Justus Piater

Submitted to the University of Liège in partial fulfillment of the requirements for the degree of Doctor of Philosophy (Computer Engineering).

Jury
Prof. J. Piater, Université de Liège
Prof. N. Krüger, Syddansk Universitet
Prof. J. Verly, Université de Liège
Prof. M. Vincze, Technische Universität Wien
Prof. L. Wehenkel, Université de Liège
Prof. J. Wyatt, University of Birmingham

Département d'Électricité, Électronique et Informatique
Faculté des Sciences Appliquées
Université de Liège

Abstract

While robots are extensively used in factories, our industry hasn't yet been able to prepare them for working in human environments – for instance in houses or in human-operated factories. The main obstacle to these applications lies in the amplitude of the uncertainty inherent to the environments humans are used to work in, and in the difficulty in programming robots to cope with it. For instance, in robot-oriented environments, robots can expect to find specific tools and objects in specific places. In a human environment, obstacles may force one to find a new way of holding a tool, and new objects appear continuously and need to be dealt with. As it proves difficult to build into robots the knowledge necessary for coping with uncertain environments, the robotics community is turning to the development of agents that *acquire* this knowledge progressively and that adapt to unexpected events.

This thesis studies the problem of vision-based robotic grasping in uncertain environments. We aim to create an autonomous agent that develops grasping skills from experience, by interacting with objects and with other agents. To this end, we present a 3D object model for autonomous, visuomotor interaction. The model represents grasping strategies along with visual features that predict their applicability. It provides a robot with the ability to compute grasp parameters from visual observations. The agent acquires models interactively by manipulating objects, possibly imitating a teacher. With time, it becomes increasingly efficient at inferring grasps from visual evidence. This behavior relies on (1) a grasp model representing relative object-gripper configurations and their feasibility, and (2) a model of visual object structure, which aligns the grasp model to arbitrary object poses (3D positions and orientations).

The visual model represents object edges or object faces in 3D by probabilistically encoding the spatial distribution of small segments of object edges or the distribution of small patches of object surface. A model is learned from a few segmented 3D scans or stereo images of an object. Monte Carlo simulation provides robust estimates of the object's 3D position and orientation in cluttered scenes.

The grasp model represents the likelihood of success of relative object-gripper configurations. Initial models are acquired from visual cues or by observing a teacher. Models are then refined autonomously by "playing" with objects and observing the effects of exploratory grasps. After the robot has learned a few object models, learning becomes a combination of cross-object generalization and interactive experience: grasping strategies are generalized across objects that share similar visual substructures; they are then adapted to new objects through autonomous exploration.

The applicability of our model is supported by numerous examples of pose estimates in cluttered scenes, and by a robot platform that shows increasing grasping capabilities as it explores its environment.

Acknowledgments

I owe my deepest gratitude to my adviser, Justus Piater, for his ever pertinent guidance and for his unconditionally enthusiastic support. I am indebted to Justus for his clear insights on visuomotor learning, and in particular for suggesting *grasp densities* which have turned out as an exciting and rewarding avenue. However, I have also appreciated very much to have an adviser with whom I could discuss not only on a strategic level, but also on very technical issues. I am very thankful to Justus for allowing me to (and constantly encouraging me to) travel and meet the robotics community.

I thank Norbert Krüger, Jacques Verly, Markus Vincze, Louis Wehenkel, and Jeremy Wyatt, members of the jury, for devoting time to reading and evaluating this document.

There are many people without whom this thesis would not have been at all possible. Amongst these people are Dirk Kraft, Oliver Kroemer, Norbert Krüger, Jan Peters, and Nicolas Pugeault. Credit for many aspects of our results, in both theory and experimentation, goes directly to them. I am very grateful to them for this fruitful collaboration. Also, I feel very lucky to have had the opportunity to work on two robotic platforms without myself having to worry about the challenges of calibration, inverse kinematics, or path planning. I thank Norbert for his determined and vigorous support in all aspects of my research. I thank Jan for his highly pertinent advice on both my present and future work.

I thank the Cognitive Vision Group in Odense, in particular Leon Bodenhagen, Emre Başeski, Lars Baunegaard With Jensen, Sinan Kalkan, Anders Kjær-Nielsen, Dirk Kraft, Norbert Krüger, Florian Pilz, Mila Popović, Nicolas Pugeault, and Shi Yan, for their help and support, and for always making me feel welcome in Odense.

I thank all my colleagues and friends from the PACO-PLUS project for sharing their ideas with me. I feel very lucky to have had the opportunity to work on a large project with such a motivated and passionate group.

I thank Gentiane Haesbroeck for her help with statistics, Fabien Scalzo for coaching me during the first six months of my work, and Denis Defrère for reading parts of this document and for making insightful suggestions.

I thank the entire staff at the Montefiore institute for making my four years as a Ph.D. student a wonderful experience, and for contributing to this work in many different ways. I am in particular grateful to the Vision Group – Arnaud, Damien, Thomas, and Wei. I also warmly thank my office neighbors and friends Bertrand, Florence, Raphaël, and Thibaut, for their constant support, and for making every day at the institute a fun day!

The *GraspIt!* simulator is written by the Robotics Lab at Columbia University, NY.

Its C++ interface *graspif* is developed at the Computer Vision and Active Perception Lab, KTH. I thank them both for generously providing me with these great tools.

I thank the Belgian National Fund for Scientific Research (FNRS) and the EU Cognitive Systems project PACO-PLUS [IST-FP6-IP-027657] for supporting this research financially.

I am indebted to my family and friends for showing motivating interest in my work, but also for helping me disconnect from it when needed. In particular, I wish to express my infinite gratitude to Audrey, to my brother, and to my father; thank you for your care and support!

Contents

Chapter 1

Introduction

Our work belongs to the fields of computer vision and vision-based grasping. Our objective is to develop a robotic agent that can learn and execute grasps on real-life objects, within a human environment. To this end, we present a three-dimensional accurate object model for autonomous visuomotor interaction, along with means of learning the model autonomously from experience. This visuomotor object model represents object grasping strategies along with visual object features that predict their applicability. Such a model allows an agent to grasp objects lying in arbitrary poses, as grasp parameters emerge from visual observations. The model is learned by letting the robot autonomously experience the correlation between successful grasps and visual structure. Concretely, this means that the robot learns how to grasp objects by using them, or simply by playing with them. Our approach thus allows a robot to operate in environments designed for humans, such as houses, offices, and human-operated factories, in which object positions are unpredictable, and where new objects appear and need to be understood and dealt with.

1.1 Robot Learning

Over the past few decades, robots have proved increasingly efficient at solving systematic tasks. Today's industry is able to build and program robots that can potentially execute tasks with an accuracy and speed largely superior to humans'. Yet, to date, robots have almost exclusively been able to work in highly controlled environments designed for them. The reason for this is that most industry robots work on programs which make strict assumptions on the structure and dynamics of their environment, and which are specific to the task at hand; when small environmental or task-related variations occur, the robot has to be re-programmed. As an example, let us consider a robot working in a car factory. This robot may be programmed to pick up a wheel from a feeder located to its left, and bolt it onto a car that sits to its right. If this robot is moved to another factory where the feeder is on its right side, or a factory where the robot is expected to remove wheels instead of attaching them, human intervention will be required before it is able to work again. Today's robots are still far from humans' versatility.

A compelling challenge of modern robotics is to conceive robots that can work in environments designed for humans, such as houses, offices, and human-operated factories [Kemp et al., 2007a]. From a robotic viewpoint, these environments are inherently unpredictable. Designing a robot that can readily work in an arbitrary house or factory is infeasible. Hence, the community has moved beyond preprogrammed designs, and it is now designing robots that can *learn and adapt* to new tasks and environments. By observing the environmental effects of their actions and the actions of others, these robots can progressively *acquire* the knowledge necessary to execute their work. Consequently, the "program" that governs the robot's actions evolves over time.

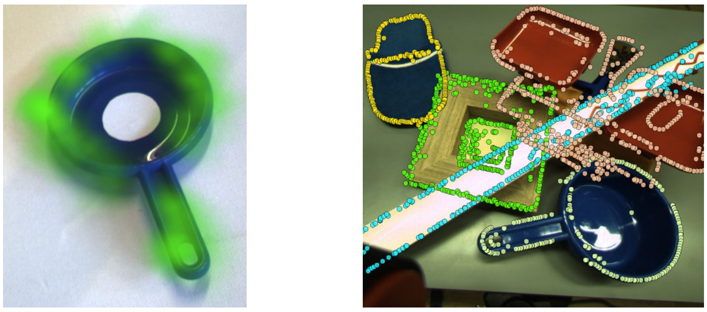
1.1.1 Problem statement: Interactive Learning of Vision-based Grasping

Solving industrial or household tasks requires robotic agents to acquire a large number of skills, such as navigation, walking, tool usage, etc. In this work, we focus on visuomotor grasping. Visuomotor grasping here refers to the computation of grasping parameters from visual observations. Visual observations tell the agent about the spatial configuration of the objects that surround it. Visual observations serve as a basis for computing grasping parameters, i.e., computing the position and orientation to which the robot must bring its hand in order to robustly grasp a target object.

In classical industrial approaches, vision-based grasping is implemented by designing 3D object models by computer-aided design (CAD), and defining a few grasping points onto the models. Such models are thus specific to a single object and a single task – different tasks generally require different grasping points. By contrast, our approach is designed to allow an agent to operate in varying workplaces without requiring re-programming. To this end, the agent is provided with very minimal environmental information. The agent does know about its body, its ability to grasp with its manipulator, and it is aware of the necessity of acquiring grasping skills. However, it has initially no knowledge about the visual or physical properties of objects. We provide the agent with models that allow it to store information about vision- and grasp-related object properties. The environment-specific and task-specific parameters of these models – how an object looks and how to grasp it – are learned interactively, from experience. The agent learns by observing the effects of exploratory actions onto objects. Concretely, the agent learns by playing with objects. For instance, it repeatedly tries to grasp an object, and whenever a grasp succeeds, it drops the object and tries to grasp it again. The agent can also learn by observing a teacher executing grasps. During these experiences, the robot collects information about the appearance of objects, and it learns relations between visual structure and good grasping points. As its experience grows, it becomes increasingly efficient at computing grasps from visual perceptions.

1.1.2 Approach: Probabilistic Sensor Models and Learning Procedures

We are developing a model of vision- and grasp-related object properties. The model can be used to locate, recognize, and grasp objects, and it can be learned autonomously from experience. It consists of a grasp model and a model of visual structure. The grasp model (Chapter 4) is a continuous description of relative object-gripper configurations



(a) Grasp success prediction

(b) Pose estimation

Figure 1.1: Visuomotor model applications. In Figure (a), a precision-pinch grasp model learned from a toy pan allows us to plot grasp success likelihood as a function of (x, y) grasping positions. The likelihood is shown through the opacity of the green mask – green regions are likely to offer good grasping points for pinch grasps. Examples of grasps generated with this model are shown in Figure 1.2. We note that a grasp model describes 3D object-relative gripper positions and orientations. Figure (a) only shows a part of the information contained in the model. Figure (b): Visual object-edge models can be used to retrieve 6D object poses (their 3D position and orientation). The pose of an object is recovered by finding the model alignment that maximizes the likelihood of the visual data. Pose estimation is illustrated by projecting the aligned models onto an image of the scene.

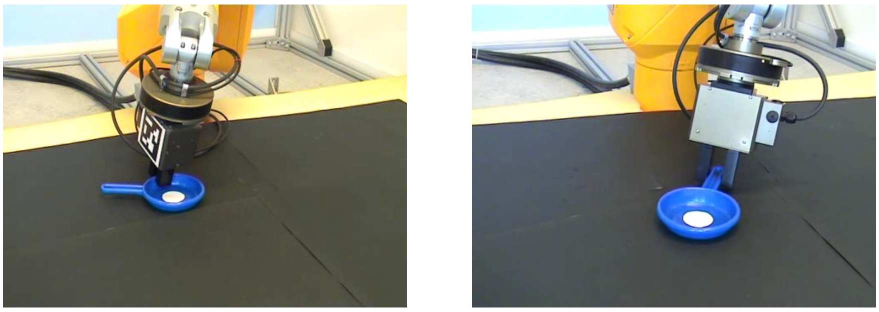


Figure 1.2: Robotic grasp examples. These grasps were computed from the grasp model of Figure 1.1a.

and their likelihood of success. Concretely, the model encodes the different ways to place a gripper near an object so that closing the gripper produces a stable grip. As illustrated in Figure 1.1a, for each relative object-gripper configuration, i.e., for each point around the object in Figure 1.1a, this model allows us to compute the likelihood of success of the corresponding grasp. Initial models are computed from visual cues, or acquired from a teacher. Models acquired from visual cues poorly represent object morphology, while models acquired from a teacher are not specifically adapted to the robot's gripper. These models are thus refined autonomously through exploration to intimately capture the properties of the objects and the gripper: the robot tries a large number of grasps and uses their outcomes to build a refined grasp model (Chapter 4). After the robot has learned a few object models, learning becomes a combination of cross-object generalization and interactive experience: grasping strategies are generalized across objects that share similar visual sub-structures; they are then adapted to new objects through autonomous exploration (Chapter 5).

The visual model represents object structure in three dimensions by encoding the spatial distribution of 3D object edges and object surface points (Chapter 3). The model is learned from 3D edge descriptors generated by sparse stereo methods [Krüger et al., 2004, Pugeault, 2008, Pugeault et al., 2010], or from points generated by a 3D scanner. It can serve to compute the 6D pose (3D position and 3D orientation) of an object in a cluttered scene, as illustrated in Figure 1.1b. In the visuomotor learning context, the role of the visual model is to align the grasp model to an object's pose, in order to allow the robot to grasp objects from arbitrary configurations.

The important common aspect of these two models is that they can be learned autonomously from noisy sensor data: the grasp model is learned from physical interaction with the object while the visual model can be learned autonomously from a few segmented stereo views or a few 3D scans of an object.

Both models rely on probabilistic encoding of low-level sensor and motor data through probability density functions (PDF). As this probabilistic representation is largely shared between the two models, it is discussed below (Chapter 2) independently of a specific modal application. Chapter 2 forms a theoretical basis for the chapters dedicated to vision and grasping.

1.2 Integration in a Cognitive Robotic Architecture

This work was developed in the context of the EU project PACO-PLUS (`http://www.paco-plus.org/`), which aimed at developing cognitive robotic agents capable of (1) developing perceptual, behavioral and cognitive categories, and (2) communicating and sharing these with humans and other artificial agents.

PACO-PLUS' developments are centered around the concept of *object-action complex* (OAC, see Wörgötter et al. [2009], Krüger et al. [2010a,b]). Object-action complexes offer a formalism for relating symbolic planning elements to noisy sensorimotor experience. They are founded on the assumption that objects and actions are intimately coupled. the interplay between objects and actions is the central property around which sensorimotor experience is abstracted into symbolic representations.

PACO-PLUS' cognitive robot platform is organized in a three-level architecture [Kraft et al., 2008]. The bottom level encompasses sensing and actuation. The middle level is concerned with the representation of objects and actions. The top level is responsible for task planning. The mid-level is largely implemented in terms of OACs relating low-level experience to high-level symbols.

The visuomotor model presented here contributes to the mid-level of the PACO-PLUS architecture, by defining objects in terms of visual and grasping features. The model is connected to visual perceptions through image-based sparse-stereo reconstructions of 3D edges [Krüger et al., 2004, Pugeault, 2008, Pugeault et al., 2010], and to grasping experience through a robotic arm and gripper. It connects to higher-level processes by providing planning elements such as "grasp object A from the left side and pick it up". Using the vocabulary of the PACO-PLUS consortium, our visuomotor model can be formalized as an OAC [Krüger et al., 2010a].

The PACO-PLUS project has given us an opportunity to realize a tightly-integrated visuomotor learning scenario. In this scenario, a robot learns object model parameters from scratch through exploration. The robot starts with no object-specific knowledge. In the first phase of the scenario, the robot executes so-called "grasp reflexes" [Popović et al., 2010] onto edges reconstructed with the method of Pugeault et al. [2010]. Many of these lead to nothing, because many edges come, e.g., from floor patterns or from ungraspable object parts. However, eventually, the robot succeeds at binding an object to its gripper – it becomes aware of it because, after lift-up, the gripper cannot be fully closed, hence something must be gripped in it. The robot then rotates the object in front of its camera, and computes a complete 3D reconstruction of object edges [Pugeault et al., 2010]. The process of gripper-object binding through grasp reflexes then acquiring an edge reconstruction is detailed in the work of Kraft et al. [2009].

Once the robot has a 3D edge reconstruction of an object, the reconstruction serves to build (1) a hierarchical visual object model, and (2) an initial grasp model. The construction of the visual model is discussed in Chapter 3. The construction of the initial grasp model is detailed in Chapter 4. Intuitively, the initial grasp model represents grasps loosely arranged around the edges recovered by the 3D edge reconstruction. At this point, the robot begins to explore the object by repeatedly attempting to grasp it at points suggested by the grasp model. Whenever the robot successfully manages to grasp and lift-up the object, it releases it and lets it fall arbitrarily before trying again. As a result, the pose of the object is different at each attempt. Prior to each grasp, the robot makes use of the visual model to visually align the grasp model to the correct object pose. A refined grasp model is eventually learned from the outcomes (success/failure) of the executed grasps. Chapter 4 presents an experiment in which the robot achieves success rates of about 60% when using the refined model, while the initial model only leads to rates of about 10%. A video illustrating this experiment is available at `http://www.montefiore.ulg.ac.be/~detryr/research.php`.

1.3 Related Work

This section's aim is to position our work within the overall robot-learning landscape. Work specifically related to either of the visual and grasp models is not discussed, as it is presented in detail in the next chapters: Chapter 3 includes a discussion of standard and state-of-the-art work on pose estimation and recognition in 2D images and 3D point clouds. Chapter 4 discusses work related to robotic grasping, and Chapter 5 discusses work related to grasp generalization.

1.3.1 Visuomotor Learning and Robotic Manipulation

This section provides a brief overview of problems and solutions related to robot learning in human environments. The discussion is globally focused on visuomotor manipulation, although some arguments and references are related to robot learning in general. Support for this section can be found in an article by Kemp et al. [2007a], in Dagstuhl seminar proceedings edited by Beetz et al. [2010], in the book of Sigaud and Peters [2010a], and in the R:SS Manipulation Workshop series [e.g. Kemp et al., 2007b].

Visuomotor manipulation can be decomposed into a set of subproblems. Robots need to be able to sense the world, and to make use of their bodies to act on it. Robots then need to plan and execute actions that will lead to the accomplishment of the task, which requires means of linking task, perception and action together. Research has shown that preprogrammed systems can integrate multiple perceptual inputs to generate useful actions [Allen et al., 1999, Kragic et al., 2001]. Yet, when working in a human environment, the amount of uncertainty, variation, and novelty the robot has to deal with in task, action and perception domains has lead researchers to move on to *adaptive* behaviors. The text below provides an overview of learning-related methods for motor- and action-related problems (Section 1.3.1) and for sensor- and perception-related problems (Section 1.3.1).

Motor-related Learning

In robotics, motor learning is usually achieved from demonstration or exploration. When learning from demonstration [Billard et al., 2008], an external agent, usually a human, transfers knowledge to the robot by demonstrating a task. Transfer can work by letting the robot observe the teacher perform the task, either through vision [Kuniyoshi et al., 1994, Moeslund et al., 2006] or through motion capture devices [Tung and Kak, 1995, de Granville et al., 2006]. Alternately, motor parameters can be transferred directly to the robot, either through teleoperation [Billard et al., 2008], or by physically moving a compliant robot body [Ito et al., 2006].

Learning from exploration refers to scenarios where the robot explores the perception-action domains and autonomously discovers sensorimotor patterns which help in solving a task [Natale et al., 2005, Montesano and Lopes, 2009, Sigaud and Peters, 2010a]. Exploratory learning has the advantage of being autonomous, and of intimately linking the robot's body to the environment. On the other hand, learning from demonstration provides highly informative data to the robot in a much shorter time, but it requires the help of a teacher. Large-scale robot systems will usually work with a combination of both: Initial knowledge is transferred to the robot via a teacher; the robot then adapts to its own morphology and to environmental variations through autonomous exploration [Sigaud and Peters, 2010b].

Robot learning has been approached with many standard machine-learning frameworks [Sigaud and Peters, 2010a], including supervised learning (e.g., building classifiers, as in the work of Saxena et al. [2008]), unsupervised learning (e.g., building mixture models, as de Granville et al. [2006]), and reinforcement learning. These frameworks have been applied independently of the learning paradigm (exploration or demonstration), although reinforcement learning has often successfully been applied to exploration [Sigaud and Peters, 2010b].

Robotic manipulation usually involves a robotic arm with a manipulation-enabled end-effector. Real-world robotic manipulation is confronted to uncertainty through, e.g., objects in motion, clutter and obstacles, object variability, and noisy perceptions. Learning has been applied for addressing these issues in a number of sub-areas. For instance, learning has been applied for extracting arm control policies from movement demonstrations, in both fast-motion problems [Kober et al., 2010] and problems where arm movements need to be adapted to obstacles while performing a manipulation task [Pastor et al., 2009]. Learning has also proved very useful for reducing the dimensionality of complex controllers, such as those of human-like hands [Ciocarlie and Allen, 2009].

In a manipulation task, the selection of grasp parameters should not depend on object properties only. The environment also has an influence on grasp feasibility – for instance, clutter can make some grasps impossible to achieve. In recent work, Gienger et al. [2008] have presented an agent that learns a grasp model that allows for optimizing grasp selection with respect to both object- and environment-related constraints.

Learning has yielded impressive results in grasping. Learning from demonstration has led to the identification of clusters in hand poses [de Granville et al., 2006, Sweeney and Grupen, 2007] or in grasp preshape sequences [Ekvall and Kragic, 2004]. Visuomotor invariants have been identified from exploration [Montesano and Lopes, 2009] and demonstration [Sweeney and Grupen, 2007, Saxena et al., 2008], effectively allowing for the reproduction of grasps on both known and novel objects. Finally, demonstration and exploration have successfully been combined to actively learn physically optimal grasps [Kroemer et al., 2009].

As noted in the paragraphs above, motor-related learning most often encompasses some perceptual learning. However, learning has also extensively been applied to sensor data independently of a motor application. As this work has its share of impact on robotics, a short discussion is provided in the next section.

Sensor-related Learning

To date, the highest-impact sensor modality for manipulation is vision, with visual input acquired either with a camera [Daniilidis and Eklundh, 2008, Collet et al., 2009] or a range scanner [Fisher and Konolige, 2008, Rusu et al., 2009]. Vision plays a key role in many aspects of manipulation: Vision brings object detection and categorization, which may for instance help a robot find the object or tool it needs. Vision also brings object localization, which tells the robot where to place its gripper to grasp an object.

In the context of vision, real-world uncertainty manifests itself through motion, occlusion, clutter or appearance variability, to cite only a few. The vision community has been particularly active in addressing these issues with machine learning. Learning has been profusely applied to object detection, recognition and localization. So-called bag-of-features models [Csurka et al., 2004] are learned by computing the appearance frequency of a number of basic elements (e.g. pixel patches) within an object view [Julesz, 1981, Salton and McGill, 1983]. Bag-of-features models rapidly categorize an object amongst a set of classes. They are robust to clutter, object deformations, and, to some extent, to object variability. Another class of methods focuses on learning the geometric structure of objects, e.g., by extracting the relative configurations of elementary parts [Felzenszwalb and Huttenlocher, 2000, Fergus et al., 2003, Fidler and Leonardis, 2007]. These methods typically yield an estimate of an object's position. For manipulation, position estimation is especially useful when objects are modeled in 3D [Rothganger et al., 2006], as it potentially allows for computing the accurate object-gripper configurations that are essential to many manipulative tasks. Part-based approaches offer robustness to occlusions, and they lend themselves to generalization by identifying parts that are shared by multiple objects.

Learning has been applied to various other areas relevant to manipulation such as segmentation and grouping [Tu and Zhu, 2002, Boiman and Irani, 2006] or stereo and depth acquisition [Saxena et al., 2005].

Touch, which is yet another important modality for manipulation, is becoming increasingly popular as tactile sensors improve. Tactile sensing provides information on force and torque at contacts between the robot and the world [Lee and Nicholls, 1999, Cutkosky et al., 2008]; they are usually fixed to the robot's manipulator. Although tactile learning has been studied far less than visual learning, it has been shown that it can improve manipulation robustness, e.g., by detecting slip [Hosoda et al., 2002], by providing touch-based object recognition [Schneider et al., 2009] or by guiding grasp policy refinement [Argall et al., 2009].

1.3.2 Developmental Robotics

This work can be associated to the field of developmental robotics. Developmental robotics is an increasingly popular research field whose definition is seemingly not fully established yet. For this reason, this section will not attempt to discuss whether our work belongs to it. Instead, we emphasize a few key points of our approach which will hopefully help the reader make his own opinion. This discussion will also clarify our use of a few context-sensitive terms.

It is generally understood that developmental robotics is concerned with the application of developmental behaviors and models to robotics, and also concerned with the study of development through the synthesis of robotic agents with developing capabilities [Lungarella et al., 2003, Meeden and Blank, 2006]. To date, we haven't meant to study the developmental impact of our robotics work. However, our learning-based approach loosely follows the biological example. In contrast to classical robotics approaches that employ CAD object models and compute grasp parameters based on analytical physical models [Bicchi and Kumar, 2000, Borst et al., 2003, Miller et al., 2003], we *learn* gripper poses that lead to stable grasps. We start with very simple initial models, which may originate from a premature vision-based grasping mechanism providing only little bias towards stable grasp configurations. While this approach yields a rather low success rate, it is sufficient to bootstrap the acquisition of object-specific knowledge for skilled grasping. This procedure – feature-induced grasping refined by sensorimotor exploration – loosely resembles human acquisition of grasping skills during infancy, and constitutes a promising avenue towards viable robotic grasping, as it does for humans. Moreover, the employed visual methods (visual model and inference, vision-induced grasping) resemble their biological counterparts: The sparse-stereo descriptors of Pugeault et al. [2010] have been motivated by the concept of hypercolumns in the human visual system [Hubel and Wiesel, 1969]. As detailed by Piater et al. [2008], the visual model presented below in Chapter 3 is compatible with studies suggesting that the visual processing stream might perform Bayesian inference within an undirected Markov chain [Lee and Mumford, 2003].

When presenting our work to a classical robotics audience, we emphasize the autonomous and low-bias aspects of its learning methods. Indeed, from a classical viewpoint, the visuomotor learning scenario mentioned above (Section 1.2) is autonomous: data is collected largely without human intervention, and model parameters are computed by the agent from the data. It is also very little biased, as the models make (relatively) few assumptions on the shape of objects and object-grasp associations.

From a purely developmental viewpoint however, qualifying our learning as autonomous and low-bias may be inadequate. Although the scenario through which model parameters are learned proceeds with little human intervention, the learning algorithms do constitute an important developmental bias, and so do the representations we use for visual and grasp features. We also note that:

- The task (“play with objects”) is given to the robot.
- The robot knows about its body and its ability to use it to grasp objects (as opposed to, e.g., the work of Stoytchev [2005, 2008]).
- Motor skills are only learned in task space: The robot learns object-relative gripper configurations that lead to successful grasps. Inverse kinematics are given. (As opposed to, e.g., Demiris and Dearden [2005], Natale et al. [2005], Rolf et al. [2009].)

We do use the terms “autonomous” and ”low-bias” to qualify our approach in the text below. As explained here, these terms should be understood in the classical robotics sense.

1.3.3 Generative Model

When modeling an input-output system probabilistically, one is usually required to choose between a *generative* or *discriminative* approach: Given a dataset of input-output pairs (x, y), a probabilistic model $P(Y|X)$ can be constructed either by constructing a function approximation of $P(Y|X)$, or, following Bayes' rule, by approximating $P(X|Y)$ and $P(Y)$ (see, e.g., Ulusoy and Bishop [2005] for a discussion in the context of computer vision). An approximation of $P(Y|X)$ forms a *discriminative* model, as its construction can make use of all the training data to identify the input patterns that best discriminate output values. On the other hand, empirical representations of $P(X|Y)$ and $P(Y)$ form a *generative* model, as they can generate input values x conditioned on a particular output y. The models presented in this text are generative. Motives for using generative visual and grasp models will be given in respective chapters.

1.4 Structure of the Dissertation

This dissertation is organized as follows. Chapter 2 explains how we represent, evaluate and integrate over probability density functions defined on the Special Euclidean group $SE(3)$. This chapter provides a theoretical basis for the next chapters, which are dedicated to vision and grasping. Chapter 3 details how we encode the spatial distribution of 3D object edges and object surface points to model object structure. In Chapter 4, we discuss the experience-based refinement of initial grasp models built from visual cues or from human demonstrations. Finally, Chapter 5 presents means of transferring grasping knowledge to novel objects.

This dissertation is organized as a collection of articles. Chapter 3 consists of two publications. Chapter 4 consists of a single article under review at the time of this writing. Chapter 5 presents recent work that has not been compiled into a self-contained publication yet. While Chapter 3 and 4 are self-contained, Chapter 5 relies on concepts discussed throughout the entire dissertation. Brief summaries of the material presented in detail in Chapter 2 are included in Chapters 3 and 4.

Chapter 2

Nonparametric Densities and Monte Carlo Integration

Our work heavily relies on the modeling of the spatial distribution of vision and grasp-related parameters with density functions. This section reviews the theory behind density estimation and integration particularized to the function domains we work with. This section also discusses details related to a computer implementation where appropriate.

Within our visual model (Chapter 3), density functions model the spatial distribution of object poses – 3DOF position and 3DOF orientation – and the distribution of short edge segments or surface patches – 3DOF position and 2DOF orientation. Within the grasp model (Chapter 4), density functions model the distribution of object-relative gripper poses – 3DOF position and 3DOF orientation.

2.1 Parametrization

Three-DOF positions are naturally parametrized with $\mathbb{R}^3$ points. Two-DOF orientations are parametrized with 3D unit vectors. The set of 3D unit vectors forms the 2-sphere S^2. As we consider edge orientations and surface normals as axial data (a surface normal v is equivalent to $-v$), each 2DOF orientation is represented by exactly two unit vectors of S^2.

Three-DOF orientations, i.e., rotations around the origin of $\mathbb{R}^3$, form the rotation group. As 3DOF rotations can be uniquely parametrized by special orthogonal matrices, the rotation group is often referred to as the special orthogonal group $SO(3)$. The text below follows this convention.

There exist multiple representations of 3DOF orientations, such as rotation matrices, Euler angles, or unit quaternions. Unit quaternions form the 3-sphere S^3 (the set of unit vectors in $\mathbb{R}^4$). A rotation of α radians about a unit vector $v = (v_x, v_y, v_z)$ is parametrized by a quaternion q defined as

$$q = \left(\cos\frac{\alpha}{2}, v_x \sin\frac{\alpha}{2}, v_y \sin\frac{\alpha}{2}, v_z \sin\frac{\alpha}{2}\right).$$

Because a rotation of α radians about v is equivalent to a rotation of $-\alpha$ about $-v$, q and $-q$ correspond to the same rotation. Consequently, unit quaternions form a double cover of $SO(3)$ – every rotation exactly corresponds to two unit quaternions.

We parametrize 3DOF orientations with unit quaternions. Quaternions offer a clear formalism for the definition of position-orientation densities (see below). From a numerical viewpoint, they are stable and free of singularities. Finally, unit quaternions allow for the definition of a rotation metric that roughly reflects our intuitive notion of a distance between rotations. The distance between two rotations θ and θ' is defined as the angle of the 3D rotation that maps θ onto θ' [Kuffner, 2004]. This metric can be easily and efficiently computed using unit quaternions as twice the shortest path between θ and θ' on the 3–sphere,

$$\mathbf{d}\big(\theta, \theta'\big) = 2 \arccos \left| \theta^\top \theta' \right|, \tag{2.1}$$

where $\theta^\top \theta'$ is the inner (dot) product of θ and θ'. In this expression, we take the absolute value $|\theta^\top \theta'|$ to take into account the double cover issue mentioned above.

Our models rely on density functions defined on $\mathbb{R}^3 \times S^2$ and $\mathbb{R}^3 \times SO(3)$. The latter is generally called the special Euclidean group $SE(3)$. The next section details the definition of density functions on $\mathbb{R}^3 \times S^2$ and $SE(3)$.

2.2 Nonparametric Density Estimation

Density estimation generally refers to the problem of estimating the value of a density function from a set of random samples drawn from it. Density estimation methods can loosely be divided into two classes: parametric or nonparametric. Parametric methods model a density with a set of heavily parametrized kernels. The number of kernels is generally smaller than the number density samples available for computing the model. The price to pay for the smaller number of kernels is the substantial effort required to tune their parameters. The most famous parametric model is the Gaussian mixture, which is generally constructed by tuning the mean and covariance matrix of each Gaussian kernel with the Expectation-Maximization algorithm [Dempster et al., 1977, Alpaydin, 2004].

Nonparametric methods represent a density simply with the samples drawn from it. The probabilistic density in a region of space is given by the local density of the samples in that region. A density can be estimated by simple methods such as histograms, or more sophisticated methods like kernel density estimation [Silverman, 1986]. Kernel density estimation (KDE) works by assigning a kernel function to each observation; the density is computed by summing all kernels (see Figure 2.1). By contrast to parametric methods, these kernels are relatively simple, generally involving a single parameter defining an isotropic variance. Hence, compared to classical parametric methods, KDE uses a larger number of simpler kernels.

In this work, densities are modeled nonparametrically with KDE. This choice is primarily motivated by the structure of the position–orientation domains on which they are defined: Capturing pertinent position-orientation correlations within a single parametric function is very complex, while these correlations can easily be captured by a large number of simple kernels. Also, the nonparametric approach eliminates

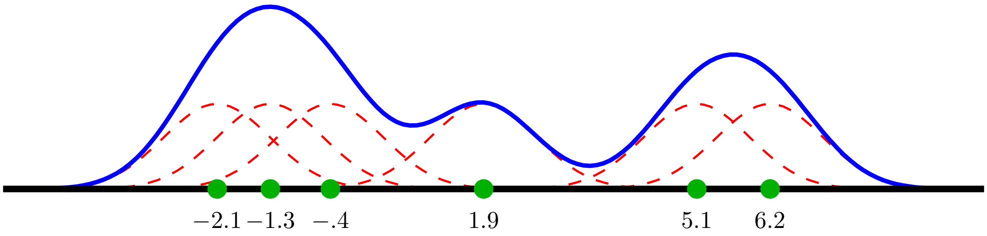


Figure 2.1: Kernel estimation (blue) of a density observed through 6 samples (green). The dashed red lines illustrate the gaussian kernels associated to each sample. The estimate is obtained by summing all kernels. To form a proper density, the function represented by the blue line should be normalized by dividing it by 6.

problems like mixture fitting, choosing a number of components, or having to make assumptions concerning density shape (e.g. normality). Kernel density estimation is thus ideal for modeling the highly multi-modal densities we are dealing with.

A density $d(x)$ is encoded by a set of observations $\hat{x}_i$ drawn from it, which we will refer to as *particles*. Density values are estimated with KDE, by representing the contribution of the i^{th} particle with a local kernel function $\mathbf{K}(\cdot\,; \hat{x}_i, \sigma)$ centered on $\hat{x}_i$. The kernel function is generally symmetric with respect to its center point; the amplitude of its spread around the center point is controlled by a bandwidth parameter σ. For conciseness, particles are often weighted, which allows one to denote, e.g., a pair of identical particles by a single particle of double mass. In the following, the weight associated to a particle $\hat{x}_i$ is denoted by w_i.

KDE estimates the value of a continuous density d at an arbitrary point x as the weighted sum of the evaluation of all kernels at x, i.e.,

$$d(x) \simeq \sum_{i=1}^{n} w_i \mathbf{K}(x\,; \hat{x}_i, \sigma)\,, \tag{2.2}$$

where n is the number of particles encoding d. Random variates from the density are generated as follows:

1. First, a particle $\hat{x}_i$ is selected by drawing i from

   $$p(i = \ell) \propto w_\ell. \tag{2.3}$$

   (This effectively gives a higher chance to particles with a larger weight.)

2. Then, a random variate x is generated by sampling from the kernel $\mathbf{K}(x\,; \hat{x}_i, \sigma)$ associated to $\hat{x}_i$.

2.2.1 Defining Densities on $SE(3)$

In order to define densities on $SE(3)$, a position-orientation kernel is required. We denote the separation of kernel parameters into position and orientation by

$$x = (\lambda, \theta) \qquad x \in SE(3), \quad \lambda \in \mathbb{R}^3, \quad \theta \in SO(3), \tag{2.4}$$

$$\mu = (\mu_t, \mu_r) \qquad \mu \in SE(3), \quad \mu_t \in \mathbb{R}^3, \quad \mu_r \in SO(3), \tag{2.5}$$

$$\sigma = (\sigma_t, \sigma_r) \qquad \sigma_t, \sigma_r \in \mathbb{R}_+. \tag{2.6}$$

The kernel we use is defined with

$$\mathbf{K}(x\,;\,\mu, \sigma) = \mathbf{\Lambda}(\lambda\,;\,\mu_t, \sigma_t)\,\mathbf{\Theta}(\theta\,;\,\mu_r, \sigma_r)\,, \tag{2.7}$$

where μ is the kernel mean point, σ is the kernel bandwidth, $\mathbf{\Lambda}$ is an isotropic location kernel defined on $\mathbb{R}^3$, and $\mathbf{\Theta}$ is an isotropic orientation kernel defined on $SO(3)$.

For $\mathbb{R}^3$, we can simply use a trivariate isotropic Gaussian kernel

$$\mathbf{\Lambda}(\lambda; \mu_t, \Sigma_t) = C_g(\Sigma_t) e^{-\frac{1}{2}(\lambda-\mu_t)^\top \Sigma_t^{-1}(\lambda-\mu_t)}, \tag{2.8}$$

where $C_g(\cdot)$ is a normalizing factor and $\Sigma_t = \sigma_t^2 \mathbf{I}$. The definition of the orientation kernel $\mathbf{\Theta}$ is based on the von Mises–Fisher distribution on the 3-sphere in $\mathbb{R}^4$ [Fisher, 1953]. The von Mises–Fisher distribution is a Gaussian-like distribution on S^3. It is defined as

$$\mathbf{F}(\theta\,;\,\mu_r, \sigma_r) = C_4(\sigma_r) e^{\sigma_r\, \mu_r^T \theta}, \tag{2.9}$$

where $C_4(\sigma_r)$ is a normalizing factor, θ and μ_r are unit quaternions, and $\mu_r^T \theta$ is a dot product. Because unit quaternions form a double cover of the rotation group, $\mathbf{\Theta}$ has to verify $\mathbf{\Theta}(q\,;\,\mu_r, \sigma_r) = \mathbf{\Theta}(-q\,;\,\mu_r, \sigma_r)$ for all unit quaternions q. We thus define $\mathbf{\Theta}$ as a pair of antipodal von Mises–Fisher distributions [Sudderth, 2006],

$$\mathbf{\Theta}(\theta\,;\,\mu_r, \sigma_r) = \frac{\mathbf{F}(\theta\,;\,\mu_r, \sigma_r) + \mathbf{F}(\theta\,;\,-\mu_r, \sigma_r)}{2}. \tag{2.10}$$

We note that the von Mises–Fisher distribution (2.9) involves the same dot product as the rotation metric defined above (2.1). The dot product $\mu_r^\top \theta$ is equal to 1 when $\mu_r = \theta$. The dot product decreases as θ moves further away from μ_r, to reach 0 when θ is a $180°$ rotation away from μ_r. In this range of values, the von Mises–Fisher kernel thus varies between $C_4(\sigma_r)e^{\sigma_r}$ and $C_4(\sigma_r)$. While e^{σ_r} grows rapidly with σ_r, $C_4(\sigma_r)$ rapidly becomes very small. This makes the computation of $\mathbf{F}$ numerically difficult. A robust approximation of $\mathbf{F}$ can be obtained with

$$\mathbf{F}(\theta\,;\,\mu_r, \sigma_r) \simeq e^{\sigma_r\, \mu_r^T \theta + C_4'(\sigma_r)}, \tag{2.11}$$

where $C_4'(\sigma_r)$ approximates the logarithm of $C_4(\sigma_r)$ [Abramowitz and Stegun, 1965, Elkan, 2006]. However, since σ_r is common to all kernels forming a density, using

$$\tilde{\mathbf{F}}(\theta\,;\,\mu_r, \sigma_r) = e^{-\sigma_r\left(1-\mu_r^T \theta\right)} \tag{2.12}$$

instead of $\mathbf{F}$ in the expression of $\mathbf{\Theta}$ (2.10) will yield density estimates equal to d (2.2) up to a multiplicative factor, while allowing for efficient and robust numerical computation. This alternative will be preferred in all situations where d need not integrate to one.

2.2.2 Defining Densities on $\mathbb{R}^3 \times S^2$

Turning to densities defined on $\mathbb{R}^3 \times S^2$, let us define

$$\begin{aligned} x &= (\lambda, \theta) & & x \in \mathbb{R}^3 \times S^2, \quad \lambda \in \mathbb{R}^3, \quad \theta \in S^2, & (2.13)\\ \mu &= (\mu_t, \mu_r) & & \mu \in \mathbb{R}^3 \times S^2, \quad \mu_t \in \mathbb{R}^3, \quad \mu_r \in S^2, & (2.14)\\ \sigma &= (\sigma_t, \sigma_r) & & \sigma_t, \sigma_r \in \mathbb{R}_+. & (2.15) \end{aligned}$$

The $\mathbb{R}^3 \times S^2$ kernel is defined as

$$\mathbf{K_3}(x\,;\,\mu, \sigma) = \mathbf{N}(\lambda\,;\,\mu_t, \sigma_t)\,\mathbf{\Theta_3}(\theta\,;\,\mu_r, \sigma_r)\,, \tag{2.16}$$

where $\mathbf{\Theta_3}$ is a mixture of two antipodal S^2 von Mises–Fisher distributions, i.e.,

$$\begin{aligned} \mathbf{\Theta_3}(\theta\,;\,\mu_r, \sigma_r) &= \frac{\mathbf{F_3}(\theta\,;\,\mu_r, \sigma_r) + \mathbf{F_3}(\theta\,;\,-\mu_r, \sigma_r)}{2}, & (2.17)\\ \mathbf{F_3}(\theta\,;\,\mu_r, \sigma_r) &= C_3(\sigma_r) e^{\sigma_r\,\mu_r^T \theta}. & (2.18) \end{aligned}$$

In the expression above, $C_3(\sigma_r)$ is a normalizing factor, which can be written as

$$\frac{\sigma_r}{2\pi\,(e^{\sigma}_{r} - e^{-\sigma_r})}. \tag{2.19}$$

$\mathbf{F_3}$ can thus be written as

$$\mathbf{F_3}(\theta\,;\,\mu_r, \sigma_r) = \frac{\sigma_r}{2\pi\,(1 - e^{-2\sigma_r})} e^{-\sigma_r \left(1 - \mu_r^T \theta\right)}, \tag{2.20}$$

which is easy to numerically evaluate. As for $SE(3)$ densities, when d need not integrate to one, the normalizing constant can be ignored.

2.2.3 Evaluation and Simulation

As the kernels $\mathbf{K}$ and $\mathbf{K_3}$ factorize to position and orientation factors, they are simulated by drawing samples from their position and orientation components independently. Efficient simulation methods are available for both normal distributions [Box and Muller, 1958] and von Mises–Fisher distributions [Wood, 1994]. The bandwidths σ_t and σ_r are computed by ad-hoc methods that depend on the application; these methods are discussed in Chapter 3 and Chapter 4.

From an algorithmic viewpoint, density evaluation is linear in the number of particles n supporting the density. Asymptotically logarithmic evaluation can theoretically be achieved with kd-trees and slightly modified kernels: Considering for instance $SE(3)$ densities and the notation introduced above (2.4–2.7), let us define a truncated kernel $\mathbf{K}'$ as

$$\mathbf{K}'(x\,;\,\mu, \sigma) = \begin{cases} \mathbf{K}(x\,;\,\mu, \sigma) & \text{if } \mathbf{d}(\lambda, \mu_t) < \lambda_\ell \text{ and } \mathbf{d}(\theta, \mu_r) < \theta_\ell, \\ 0 & \text{else,} \end{cases} \tag{2.21}$$

where λ_ℓ and θ_ℓ are fixed position and orientation thresholds. The value at (λ, θ) of a density modeled with $\mathbf{K}'$ only depends on particles whose distance to (λ, θ) is smaller than λ_ℓ in the position domain, and smaller than θ_ℓ in the orientation domain. These particles can theoretically be accessed in near-logarithmic time with a kd-tree.

However, traversing a kd-tree is computationally more expensive than traversing a sequence. Hence, kd-trees only become profitable for n larger than a certain threshold. In the case of our 5DOF or 6DOF domains and sets of 500–2000 particles, we have observed best performances when organizing particle positions in a kd-tree, while keeping orientations unstructured. Density evaluation is always sub-linear in the number of particles, and it approaches a logarithmic behavior as n increases.

2.2.4 Resampling

In the next chapters, certain operations on densities will yield very large particle sets. When the number of particles supporting a density becomes prohibitively high, a sample set of n elements will be drawn and replace the original representation. This process will be referred to as *resampling*. For efficient implementation, *systematic sampling* [Douc et al., 2005] can be used to select n kernels from the distribution defined in Eq. 2.3. In the following, n will generally denote the number of particles per density.

2.3 Integration

The models presented in the next chapters make extensive use of approximate integration of density functions. In this work, approximate integration is generally carried out through Monte Carlo integration, which is briefly introduced below. The remainder of the section then details the convolution of $SE(3)$ and $\mathbb{R}^3 \times S^2$ densities, as this operation is instrumental in the models of the next chapters.

2.3.1 Monte Carlo Integration

Integrals over $SE(3)$ and $\mathbb{R}^3 \times S^2$ are solved numerically with Monte Carlo methods. Monte Carlo integration is based on random exploration of the integration domain. By contrast to classical numerical integration algorithms that consider integrand values at points defined by a rigid grid, Monte Carlo integration explores the integration domain randomly.

Integrating the product of two density functions $f(x)$ and $g(x)$ defined on the same domain is performed by drawing random variates from g and averaging the values of f at these points [Caflisch, 1998]

$$\int f(x)g(x)\mathrm{d}x \simeq \frac{1}{n}\sum_{i=1}^{n} f(x_i) \quad \text{where} \quad x_i \sim g(x). \tag{2.22}$$

2.3.2 Cross-Correlation

Chapter 3 and Chapter 5 make extensive use of density convolutions. An in-depth description of their approximate computation is given below.

Let f and g be two density functions with domain D, where D is either $SE(3)$ or $\mathbb{R}^3 \times S^2$. Let also

$$t_x(y) : D \to D \tag{2.23}$$

denote the rigid transformation of y by x, with $y \in D$ and $x \in SE(3)$. The $SE(3)$ cross-correlation of f and g is written as

$$c(x) = \int f(y)\, g(t_x(y))\mathrm{d}y. \tag{2.24}$$

As both f and g have unit integrals, Fubini's theorem guarantees that c also integrates to one.

The cross-correlation of f and g is approximated with Monte Carlo integration as

$$c(x) \simeq \frac{1}{n}\sum_{\ell=1}^{n} g(t_x(y_\ell)) \quad \text{where} \quad y_\ell \sim f(y). \tag{2.25}$$

Sampling from $c(x)$ can be achieved by simulating $h(x) = g(t_x(y_f))$, where $y_f \sim f(y)$. The simulation of $h(x)$ depends on the domain D on which f and g are defined. We first consider the case $D = SE(3)$. In this case, drawing a sample from $h(x)$ amounts to computing the (unique) transformation x_* that maps y_f onto y_g, where $y_g \sim g(y)$. When $D = R^3 \times S^2$, the transformation between y_f and y_g is not unique anymore; sampling $h(x)$ is done by selecting one transformation from a uniform distribution on the transformations that map y_f onto y_g.

Chapter 3

Visual Model

This chapter describes a probabilistic, generative model of 3D object structure and appearance. A competitive generative 3D model is a very useful asset in the context of visuomotor grasping, as it allows for precise alignment of a grasp model to arbitrary object configurations.

The model works on perceptual input consisting of a constellation of points whose geometric arrangement conveys information about object shape and possibly appearance. A model is learned from percepts emerging from an object. A model can serve to locate and recognize the object in an arbitrary scene. This chapter considers the use of the model with two different types of percepts. We first consider surface-point data, obtained with a 3D scanner. Laser scanners densely sample surfaces to produce a set of 3D points that accurately reconstruct surface shapes (Figure 3.1). Scanners are usually slow – capturing a range image often takes several minutes. Faster rangefinders have recently emerged, but the price to pay for a higher frame rate is a higher noise level, which makes these devices difficult to use for shape retrieval. The second perceptual source discussed in this chapter corresponds to edge-point data (Figure 3.2). This data is obtained with a sparse-stereo method which reconstructs 3D edges from stereo imagery [Krüger et al., 2004, Pugeault, 2008]; a reconstruction is composed of a set of short edge segments that bear geometric information (position, orientation) and

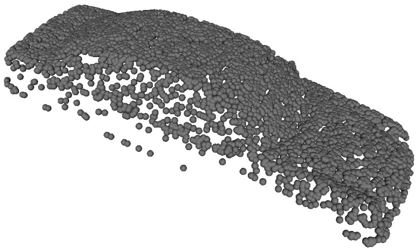


Figure 3.1: A 3D scan of a toy car. Each ball represents a 3D point recorded by the scanner.

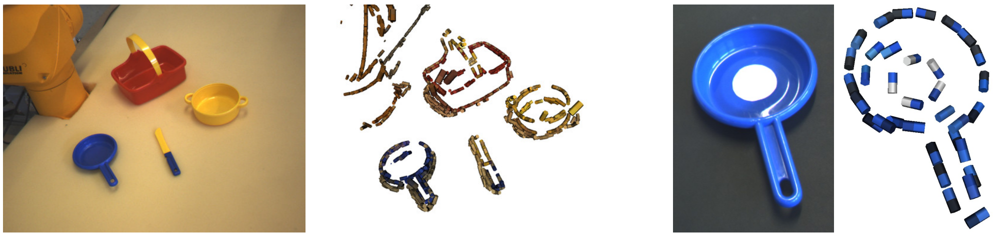


Figure 3.2: Sparse-stereo reconstructions of object edges [Krüger et al., 2004, Pugeault, 2008]. Each cylinder corresponds to an edge-segment descriptor. The axis of a cylinder is aligned with the direction of the modeled edge. Each cylinder bears the two colors found on both sides of the edge in 2D images. For clarity, the reconstruction of the toy pan (right side of the figure) only shows a fraction of the descriptors available for this object.

photometric information (including edge colors). Compared to range scanning, the computation of an edge reconstruction is fast – it usually takes a few seconds. Unlike range scanners, this sparse-stereo method can be motivated biologically, which makes it a good candidate for cognitive architectures.

The main idea behind our visual model is to encode the spatial distribution of local percepts (surface points or edge segments) with a density function. Perceptual evidence is turned into a density through kernel density estimation (KDE). Intuitively, a density is estimated by assigning a kernel function to each perceptual observation (each surface point or edge segment) and summing them. KDE is discussed in detail in Chapter 2.

Our visual model represents an object as a hierarchy of object parts. In a simple two-level hierarchy, an object is represented by a set of parts and by their relative geometric configuration. A basket, for instance, could be represented with two parts representing its bucket and handle (see Figure 3.3). The shape of each part is modeled with a density function, as described above. Relative part configurations are encoded by defining all density functions in a common reference frame, so that their sum provides a complete reconstruction of the object. Modeling low-level percepts probabilistically allows us to model observational uncertainty. In our nonparametric representation, uncertainty is modeled by adjusting kernel sizes. Also, a continuous model of object structure allows us to perform detection without explicit model-to-scene correspondences, as described below.

More sophisticated part combinations can be obtained by defining taller hierarchies of parts (Figure 3.4). The structure of high-level parts is still modeled with density functions. However, instead of representing the distribution of perceptual observations (e.g., points from a range scanner), these densities model the spatial distribution of their child parts.

One key aspect of our vision solution is its building on a generative model. Denoting by O an object name and by D perceptual observation, a probabilistic object model can be defined with object-specific sensor data distributions $P(D|O)$ (generative) or by object probabilities $P(O|D)$ (discriminative). A purely discriminative model solely concentrates on the recognition task, i.e., discriminating between the objects available

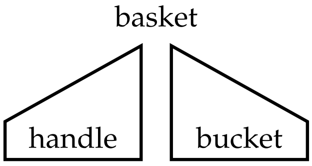


Figure 3.3: Illustration of a two-level hierarchy for a basket.

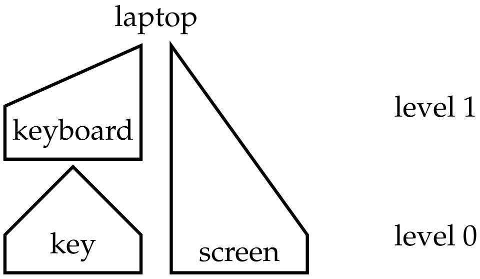


Figure 3.4: Illustration of a three-level hierarchy for a laptop.

during the learning process. Discriminative object models will generally ignore any information common to the objects of the training set, which makes them efficient at the recognition task. By contrast, a generative model represents sensor data for each object independently. They can serve to locate objects and segment them from the background. Also, when adding a new object to a library, previously existing models need not be re-learned. Finally, generative models can also yield object recognition, as demonstrated below. Generative and discriminative vision models have been compared in detail by Ulusoy and Bishop [2005].

Our model can serve to accurately compute the 3D position and 3D orientation (i.e., the *pose*) of an object in a novel scene (Figure 3.5). Pose estimation can be easily described for a simple hierarchy that contains a single part: In this case, the object model simply consists of a single density, which we denote by $\psi(x)$. An illustration of

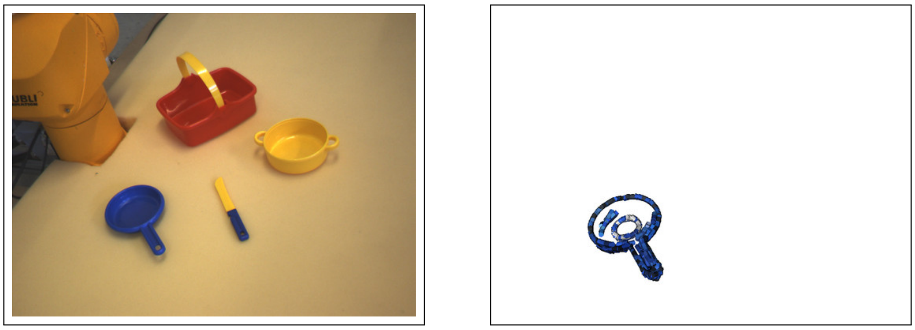
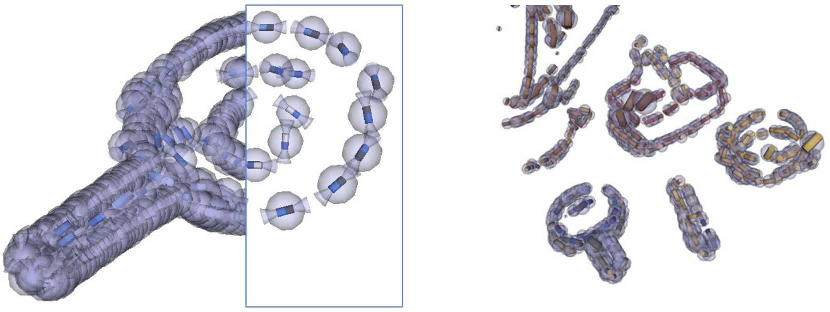


Figure 3.5: Pose estimation. Pose estimation amounts to recovering the relative configuration between the camera and an object. The relative configuration estimated from the left image (and the other image of the stereo camera) is illustrated in the right image.



(a) Object model $\psi(x)$ (b) Novel scene $\phi(x)$

Figure 3.6: Edge densities. Translucent shapes illustrate the kernels associated to edge-segment observations. They indicate one standard deviation in position (translucent spheres) and one standard deviation in orientation (cones). In Figure (a), within the blue frame, a limited number of kernels is rendered in order to improve clarity.

such a model for the toy pan of Figure 3.2 is shown in Figure 3.6a. Estimating the pose of this object in a novel scene (e.g., the scene shown on the left of Figure 3.2) works by first extracting a sparse-stereo reconstruction of the scene (Figure 3.2), then turning this reconstruction into an edge-density through KDE (Figure 3.6b). Pose estimation then amounts to computing the cross-correlation of the model and the scene, which effectively yields the pose likelihood of the toy pan over the scene. A single pose estimate can finally be obtained from the largest mode of the pose likelihood. Cross-correlations are approximated by Monte Carlo integration. The maximum of the pose likelihood can be obtained through simulated annealing on a Markov chain whose invariant distribution is an increasing power of the scene-object cross-correlation.

For general hierarchies, inference is implemented with belief propagation (BP). The computation of a single BP message corresponds to the cross-correlation described above.

A visual model is learned from a point-cloud or edge reconstruction of an object. We currently learn models in a bottom-up fashion, by first defining bottom-level parts through observation clustering and KDE, then iteratively combining parts together to form a hierarchy. A complete 3D model can be learned from a set of unregistered views of an object; novel views are incrementally aligned and fused with the model. Model learning and model exploitation are thus seamlessly integrated.

Early tridimensional models with methods for pose estimation were parametric wire-frame models created through computer-aided design [Lowe, 1991]. However, models that can be autonomously acquired from sensor data quickly appeared, and lead to robust solutions for both 3D range data [Johnson and Hebert, 1999, Mian et al., 2006] or stereo imagery [Rothganger et al., 2006, Savarese and Fei-Fei, 2007, Liebelt et al., 2008]. An interesting aspect of our method is that it can be applied to both kinds of sensor data, as demonstrated below.

Technical details are presented in the two papers included below. The first paper, entitled "Continuous Surface-point Distributions for 3D Object Pose Estimation and Recognition" (R. Detry and J. Piater. In *Asian Conference on Computer Vision*, 2010; included at pages 24–24), demonstrates maximum-likelihood pose estimation and object detection/recognition on 3D-scan data using simple two-level models. The second paper, "A Probabilistic Framework for 3D Visual Object Representation" (R. Detry, N. Pugeault, and J. Piater. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(10):1790–1803, 2009; included at pages 24–24), discusses the general hierarchical model on sparse-stereo data. Although the latter chronologically comes before the former, we believe the order in which they are included below allows for a smoother read.

We have contributed all the technical developments presented in the two articles cited above. Discussions with our collaborators (Norbert Krüger and Nicolas Pugeault from the University of Southern Denmark) were focused on scientific strategy, on the use of the sparse stereo reconstruction method developed by them [Pugeault et al., 2010], and on the adaptation of this reconstruction method to our purpose (short-range object pose estimation). We note that, while the two articles included in this chapter provide a summary of our contributions, we have also contributed to preliminary work and to related projects, which yielded the following publications:

1. R. Detry, N. Pugeault, and J. H. Piater. Probabilistic pose recovery using learned hierarchical object models. In *International Cognitive Vision Workshop (Workshop at the 6th International Conference on Vision Systems)*, pages 107–120, Berlin, Heidelberg, 2008. Springer-Verlag. doi: 10.1007/978-3-540-92781-5_9

2. D. Kraft, E. Başeski, M. Popović, A. M. Batog, A. Kjær-Nielsen, N. Krüger, R. Petrick, C. Geib, N. Pugeault, M. Steedman, T. Asfour, R. Dillmann, S. Kalkan, F. Wörgötter, B. Hommel, R. Detry, and J. Piater. Exploration and planning in a three-level cognitive architecture. In *International Conference on Cognitive Systems (Workshop at the IEEE International Conference on Robotics and Automation)*, 2008. Extended Abstract

3. J. Piater, F. Scalzo, and R. Detry. Vision as inference in a hierarchical markov network. In *International Conference on Cognitive and Neural Systems*, 2008. Extended Abstract

4. J. Piater and R. Detry. 3D probabilistic representations for vision and action. In *Robotics Challenges for Machine Learning II (Workshop at the IEEE/RSJ International Conference on Intelligent Robots and Systems)*, 2008. Extended Abstract

5. R. Detry and J. H. Piater. Hierarchical integration of local 3D features for probabilistic pose recovery. In *Robot Manipulation: Sensing and Adapting to the Real World (Workshop at Robotics, Science and Systems)*, 2007

The technical contents of those publications of which I am first author (1, 5) are covered in the articles included below. The other articles (2, 3, 4) go beyond the scope of this thesis and are not discussed here.

For copyright reasons, the publications that form this chapter cannot be included here. These publications are:

- R. Detry and J. Piater. Continuous surface-point distributions for 3D object pose estimation and recognition. In *Asian Conference on Computer Vision*, 2010
- R. Detry, N. Pugeault, and J. Piater. A probabilistic framework for 3D visual object representation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(10):1790–1803, 2009c. doi: 10.1109/TPAMI.2009.64

Please obtain copies from their respective publishers. Copies are also available at `http://orbi.ulg.ac.be/`.

Chapter 4

Grasp Model

In classical robotics, grasp parameter computation generally relies on contact force analysis [Bicchi and Kumar, 2000]. Recently, methods that instead *learn* how to grasp have become increasingly popular [Coelho et al., 2000, de Granville et al., 2006, Sweeney and Grupen, 2007, Saxena et al., 2008, Montesano and Lopes, 2009]. These methods provide the agent with means of representing relations between its manipulator and its environment, and means of learning these from experience – exploration or imitation. Learned manipulator-environment relations are typically formalized through the concept of grasp *affordances*. Affordances have become a popular formalization for cognitive control processes, while bringing valuable insight on how cognitive control can be done [Gibson, 1979, Sahin et al., 2007]. In the context of grasping, an affordance model represents the success of grasping solutions applied to an object.

This chapter develops means for a robotic agent to learn three-dimensional accurate grasp affordance models through interactive experience. Our aim is to provide robotic agents with means of acquiring and modeling object grasping properties in order to facilitate reasoning on grasping solutions and their feasibility. The model encodes relative object-gripper poses (3D positions and orientations) that yield stable grasps. The feasibility of object-relative grasps is represented probabilistically with a density function defined on the space of 6D gripper poses. These functions are referred to as *grasp densities*.

Grasp densities are linked to visual stimuli through registration with a visual model of the object they characterize, which allows the robot to grasp objects lying in arbitrary poses: to grasp an object, the object's model is visually aligned to the correct pose. The aligned grasp density is then combined to reaching constraints to select the maximum-likelihood achievable grasp. Grasp densities are learned and refined through exploration: grasps sampled randomly from a density are performed, and an importance-sampling algorithm learns a refined density from the outcomes of these experiences. Initial grasp densities are computed from the visual model of the object, or acquired from a human teacher.

Combining the visual model described in the previous chapter to the grasp-densities framework yields a largely autonomous visuomotor learning platform. We present below an experiment in which this platform is used to learn and refine grasp densities for a set of three objects presenting large differences in shape and structure. The experiment demonstrates that through learning, the robot becomes increasingly efficient at inferring grasp parameters from visual evidence. The experiment also yields conclusive results in practical scenarios where the robot needs to repeatedly grasp an object lying in an arbitrary pose, where each pose imposes a specific reaching constraint, and thus forces the robot to make use of the entire grasp density to select the most promising achievable grasp. This work led to publications in the fields of robotics [Detry et al., 2010a,b] and developmental learning [Detry et al., 2009a]. We have integrated these publications together to form the article "Learning Grasp Affordance Densities" (R. Detry, D. Kraft, O. Kroemer, L. Bodenhagen, J. Peters, N. Krüger, and J. Piater, submitted to *Paladyn. Journal of Behavioral Robotics*). This article is included below (page 28–28).

We have contributed most of the technical developments presented in the article included below (nonparametric representation, learning). Discussions with our collaborators (the University of Southern Denmark and the MPI Tübingen) were focused on scientific strategy, and on experimental design. We received great practical help from our collaborators, which provided the robot platforms and greatly helped us perform the experiments. We note that, while the article included in this thesis provides a summary of our contributions, we have also contributed to preliminary work and to related projects, which yielded the following publications:

1. R. Detry, E. Başeski, M. Popović, Y. Touati, N. Krüger, O. Kroemer, J. Peters, and J. Piater. Learning continuous grasp affordances by sensorimotor exploration. In O. Sigaud and J. Peters, editors, *From Motor Learning to Interaction Learning in Robots*, pages 451–465. Springer-Verlag, 2010a. doi: 10.1007/978-3-642-05181-4_19

2. R. Detry, D. Kraft, A. G. Buch, N. Krüger, and J. Piater. Refining grasp affordance models by experience. In *IEEE International Conference on Robotics and Automation*, pages 2287–2293, 2010b. doi: 10.1109/ROBOT.2010.5509126

3. A. Erkan, O. Kroemer, R. Detry, Y. Altun, J. Piater, and J. Peters. Learning probabilistic discriminative models of grasp affordances under limited supervision. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010. doi: 10.1109/IROS.2010.5650088

4. D. Kraft, R. Detry, N. Pugeault, E. Başeski, F. Guerin, J. Piater, and N. Krüger. Development of object and grasping knowledge by robot exploration. *IEEE Transactions on Autonomous Mental Development*, 2010. doi: 10.1109/TAMD.2010.2069098

5. O. Kroemer, R. Detry, J. Piater, and J. Peters. Adapting preshaped grasping movements using vision descriptors. In *From Animals to Animats 11 – International Conference on the Simulation of Adaptive Behavior*, 2010a. doi: 10.1007/978-3-642-15193-4_15

6. O. Kroemer, R. Detry, J. Piater, and J. Peters. Grasping with vision descriptors and motor primitives. In *International Conference on Informatics in Control, Automation and Robotics*, 2010b

7. O. Kroemer, R. Detry, J. Piater, and J. Peters. Combining active learning and reactive control for robot grasping. *Robotics and Autonomous Systems*, 2010c. doi: 10.1016/j.robot.2010.06.001

8. J. Piater, S. Jodogne, R. Detry, D. Kraft, N. Krüger, O. Kroemer, and J. Peters. Learning visual representations for perception-action systems. *International Journal of Robotics Research*, 2010. doi: 10.1177/0278364910382464

9. R. Detry, E. Başeski, N. Krüger, M. Popović, Y. Touati, and J. Piater. Autonomous learning of object-specific grasp affordance densities. In *Approaches to Sensorimotor Learning on Humanoid Robots (Workshop at the IEEE International Conference on Robotics and Automation)*, 2009b

10. R. Detry, E. Başeski, N. Krüger, M. Popović, Y. Touati, O. Kroemer, J. Peters, and J. Piater. Learning object-specific grasp affordance densities. In *IEEE International Conference on Development and Learning*, pages 1–7, 2009a. doi: 10.1109/DEVLRN.2009.5175520

11. D. Kraft, R. Detry, N. Pugeault, E. Başeski, J. Piater, and N. Krüger. Learning objects and grasp affordances through autonomous exploration. In *International Conference on Computer Vision Systems*, volume 5815/2009, pages 235–244, 2009. doi: 10.1007/978-3-642-04667-4_24

12. O. Kroemer, R. Detry, J. Piater, and J. Peters. Active learning using mean shift optimization for robot grasping. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2610–2615, 2009. doi: 10.1109/IROS.2009.5354345

13. J. Piater, S. Jodogne, R. Detry, D. Kraft, N. Krüger, O. Kroemer, and J. Peters. Learning visual representations for interactive systems. In *International Symposium on Robotics Research*, 2009

The technical contents of the publications of which I am first author (1, 2, 9, 10) are covered in the article below. The other articles (3, 4, 5, 6, 7, 8, 11, 12, 13) go beyond the scope of this thesis and are not discussed here.

For copyright reasons, the publication that forms this chapter cannot be included here. This publication is:

- R. Detry, D. Kraft, O. Kroemer, L. Bodenhagen, J. Peters, N. Krüger, and J. Piater. Learning grasp affordance densities. *Paladyn. Journal of Behavioral Robotics*, (submitted)

Please obtain a copy by contacting us or the publisher. A copy will also be available at `http://orbi.ulg.ac.be/`.

Chapter 5

Generalizing Grasp Densities

This section describes means of generalizing and transferring grasping experience across objects. Many objects share similarities in shape and grasping properties, and both are often correlated. It seems only natural to make use of these similarities when trying to grasp a novel object: when a novel object appears, instead of starting grasping exploration from scratch, the robot can try to apply the knowledge it has acquired for partly similar objects.

In the context of grasp densities, our aim is to make use of previously-acquired models for creating the initial densities of novel objects. Our approach relies on the discovery of tight associations between grasps and object parts. We propose to learn object part models – where a part model is composed of a visual model (edge/face density) and a grasp model (grasp density) – for which the visual component robustly predicts the associated grasps. To this end, we start by defining part candidates by arbitrarily segmenting regions from known object models (e.g., we take the handle of the pan). The ability of each part to robustly map visual structure to grasp parameters is measured on the set of known object models: For each candidate part model p, we detect the visual model of p in all object models, and see if, throughout all detected poses of p, we find a strong correlation between the grasp density of p and the grasp density of the object model at the detected pose. Parts that successfully predict grasp densities are then used to form initial densities for new objects.

5.1 Visuomotor Generalization

Learning a 3D model from experience is an important step towards the development of robots that can work in human environments. Yet, in order to implement this behavior effectively, it is crucial to allow robots to generalize grasping experience across objects that share similar visual properties. This skill, largely mastered by the human visuomotor system, allows us to quickly adapt to smooth environmental changes. For instance, although knives, spoons and dishes are usually different from one kitchen to another, preparing a dish in somebody else's kitchen is not dramatically more difficult for us than preparing one in our own kitchen.

The visuomotor models presented in previous chapters are object-specific – they associate grasp strategies to a visual representation of the whole object. This produces accurate visual alignments, which in turn permits precise grasp executions. It also allows our system to suggest grasps onto occluded or partly-occluded object parts, and it makes it very robust to visual noise. The downside of this approach is that the visual model of an object A typically contains a lot of information that is not directly relevant to *predicting* grasp applicability. Hence, using A's model to grasp a new, *partly* similar object B is not directly possible. In order to allow the agent to generalize its acquired knowledge to new objects, we are developing means of finding within known object models *minimal* recurring visuomotor patterns, i.e., patterns for which the visual component is a robust predictor of the associated grasps. To this end, we define a formal measure of pattern generality. The generality of a pattern is measured by its ability to robustly predict grasps across the library L of known object models. This measure relies on a function $f(p, o)$ that yields a high value if the visual component of a pattern p successfully identifies regions of an object o associated to grasping strategies that are similar to its own, and discards those that are not. The generality measure of p is then defined from the statistics of $\{f(p, o) : o \in L\}$. As described below, the agent will systematically evaluate the generality of patterns randomly segmented from existing models, yielding a set of patterns ordered by their ability to generalize. Those that yield a high measure will be selected to form the initial grasp models of new objects. Naturally, exploratory refinement of these initial models will still be required. It should however converge faster than the vision-based bootstrap method of Chapter 4. We note that the process of discovering recurring patterns (i.e. generalization) is run offline – it is entirely based on acquired models and it does not require the robot to execute grasps. The process which still requires exploration is the adaptation of generalization-based models to specific objects.

Our visuomotor model offers powerful and elegant means of implementing f. Its probabilistic representation of visual structure and grasp strategies with density functions defines a convenient abstraction which allows us to think of solutions in terms of generic probability and machine-learning tools. In particular, we detail below how the grasping correlation between a generic pattern and an object model can be approximated with the Bhattacharyya distance [Bhattacharyya, 1943].

The next section discusses related work. Section 5.3 details the learning of recurring visuomotor patterns, and the construction of experience-based initial densities. Section 5.4 shows encouraging preliminary results obtained in both real-world and simulated environments.

5.2 Related Work

The grasping community has studied grasp generalization trough a number of different approaches. Goldfeder et al. [2009] have presented a data-driven approach, which consists in accumulating a large database of object–grasp pairs. Grasp parameters for a novel object A are recovered by selecting from the database the object that best matches A's shape. As the size of the database increases, the likelihood of finding an object similar to A grows.

Concurrently, a number of groups have developed means of sharing grasps across objects by linking grasps to object parts that predict their applicability, which is also the approach considered in this chapter.

Part-grasp associations have been defined mainly (1) by learning a mapping between low-level visual features and grasp parameters or (2) by making use of explicit shape models for describing parts. The first approach has been exploited, e.g., by Saxena et al. [2008] and by Montesano and Lopes [2009], who have learned a mapping from local image features to grasp parameters. Linking grasps to low-level visual features facilitates the emergence of generic visuomotor associations, as it limits the chance of linking grasps to unrelated visual percepts. However, local features usually imply poor geometric resolution. Associating, e.g., precise-pinch grasps to these is usually difficult.

Another approach is to make use of explicit shape models for describing parts, as they allow for precise alignment to the shape of a novel object. In the work of Miller et al. [2003], a set of (hand-defined) shape primitives annotated with grasp approaches allowed the authors to generate grasp parameters by fitting the primitives to a shape model of the novel object. Sweeney and Grupen [2007] have demonstrated the learning of similar part-shape–grasp associations, yet still using a rather simple shape model consisting of a single ellipsoid.

In this chapter, we present means of representing associations between grasps and part-shape models. Contrary to Miller et al. [2003], we learn these associations through visuomotor exploration. We go further than Sweeney and Grupen [2007] by using much finer shape models. Also, the learning methods presented below are novel means of extracting visuomotor correlation.

5.3 Recurring Visuomotor Patterns

As described above, our goal is to identify a set of visuomotor patterns for which the visual component is a robust predictor of the grasping component. These generic patterns are discovered as follows:

1. Randomly segment a set of P object parts $\{p^{(i)}\}_{i\in[1,P]}$ from the library of known objects $L = \{o^{(i)}\}_{i\in[1,N]}$ (described below in Section 5.3.1).

2. For each part p, compute a generality measure $m(p, L)$ with respect to the set of known objects L (described below in Section 5.3.2).

The parts that yield a high measure will be selected for creating the initial grasp models of new objects (described below in Section 5.3.3).

5.3.1 Generating Candidates

In the following, we consider that visual object models are made up of a single edge-segment or surface-patch density (by contrast to the multi-part models discussed in Chapter 3). Segmenting one object part p from the object library $L = \{o^{(i)}\}_{i\in[1,N]}$ works as follows:

1. Select one object model $o = (o_v, o_g)$ from L, where
   - The visual component of o (single edge-segment density or surface-point density) is denoted by o_v, with
   $$o_v : \mathbb{R}^3 \times S^2 \to \mathbb{R}.$$
   - The grasping component of o (grasp density) is denoted by o_g, with
   $$o_g : SE(3) \to \mathbb{R}.$$
2. Let d be the maximum distance between two particles supporting o_g, which approximately corresponds to the diameter of o's bounding sphere.
3. Select r uniformly in $[0, d]$.
4. Let a be the position of a particle randomly selected from o_g.
5. The grasp model of p, denoted by p_g, is defined from the particles of o_g which lie within the sphere of radius r centered at a. The visual model of p, denoted by p_v, is defined from the particles of o_v which lie within the sphere of radius r centered at a.

5.3.2 Generality Measure

This section defines a measure $m(p, L)$ of the generality of part p with respect to the set of known objects L. We start by defining the ability of a pattern p to predict the grasp model of an object o. Let us consider a visuomotor pattern $p = (p_v, p_g)$, and an object $o = (o_v, o_g)$ selected from L, where:

- p_v is the visual model (single edge-segment density or surface-point density) of p.
- p_g is the grasp model (grasp density) of p.
- o_v is the visual model of o.
- o_g is the grasp model of o.

We note that the visual models p_v and o_v contain no color information. They only consist of a $\mathbb{R}^3 \times S^2 \to \mathbb{R}$ density.

Let $t_x(\cdot)$ denote a rigid transformation by $x \in SE(3)$, and let $t_x^{-1}(\cdot)$ denote the inverse of $t_x(\cdot)$, such that
$$(t_x \circ t_x^{-1})(y) = y \tag{5.1}$$
for all y in $SE(3)$ or $\mathbb{R}^3 \times S^2$. For clarity, in the equations below, $t_x(y)$, which gives the transformation of y by x, will be denoted by $y + x$. Similarly, $t_x^{-1}(y)$ will be denoted by $y - x$.

The $SE(3)$ cross-correlation of p_v with o_v gives the pose likelihood of p in o (see Chapter 3 for more details). It can be written as
$$d_v(x\,;\, p_v, o_v) = \int p_v(y - x) o_v(y) \mathrm{d}y. \tag{5.2}$$

Intuitively, $d_v(x\,;\,p_v, o_v)$ is computed by "shifting" $p_v(y)$ to pose x, then computing its overlap with $o_v(y)$. Likewise, the cross-correlation of p_g with o_g

$$d_g(x\,;\,p_g, o_g) = \int p_g(y-x)o_g(y)\mathrm{d}y \tag{5.3}$$

is a measure of similarity between p_g "shifted" by x and o_g. The models p_v, p_g, and o_v allow for the definition of a grasp density h for the object modeled by o, as

$$h(x\,;\,p_v, p_g, o_v) = \frac{1}{Z}\int [d_v(y\,;\,p_v, o_v)]^c\, p_g(x-y)\mathrm{d}y, \tag{5.4}$$

where Z is a normalizing factor, and c controls the trade-off between robust prediction and generalization. The expression $p_g(x-y)$ corresponds to $p_g(x)$ "shifted" by y. Intuitively, the integral (5.4) considers all the different ways to "align" the pattern p with the object. The density h is computed as the weighted sum of all possible alignments of p_g, where weights – given by $d_v(y\,;\,p_v, o_v)$ – are computed from visual correlation. The constant c controls the trade-off between robust prediction and generalization. If $c = 0$, h represents random grasps. As c grows, h converges towards the transformation of p_g by $\arg\max_x d_v(x\,;\,p_v, o_v)$, i.e., the transformation of p_g by the maximum-likelihood pose of p_v in o_v. In the experiments below, c is set to 5.

The ability of p to predict the grasping properties of the object modeled by o can be measured by the similarity of h and o_g. Using the Bhattacharyya coefficient [Bhattacharyya, 1943], this similarity is written as

$$f(p, o) = \int \sqrt{h(x\,;\,p_v, p_g, o_v)o_g(x)}\mathrm{d}x, \tag{5.5}$$

where $f(p, o) = 1$ if $h(x\,;\,p_v, p_g, o_v) = o_g(x)$ for all x.

The generality of p with respect to the object library L is computed from the statistics of $\{f(p, o) : o \in L'\}$, where L' corresponds to L minus the object from which p was segmented. In the experiments below, the generality of p is computed as the arithmetic mean of $\{f(p, o) : o \in L'\}$

$$m(p, L) = \frac{1}{N-1}\sum_{o\in L'} f(p, o). \tag{5.6}$$

5.3.3 Creating Initial Grasp Densities

The procedure described above is run offline to produce a large number of parts characterized by a generality measure. The k parts that generalize best are selected to create the initial models of novel objects. We denote by $K = \{p^{(i)}\}_{i\in[1,k]}$ the set of selected parts, and by o_v the visual model of a novel object. An initial density for this object is created as

$$h(x\,;\,K, o_v) = \frac{1}{Z}\int \sum_{p\in K} [d_v(y\,;\,p_v, o_v)]^c\, p_g(x-y)\mathrm{d}y, \tag{5.7}$$

where Z is a normalizing factor. If $k = 1$, the expression above correspond to Eq. 5.4. If $k > 1$, $h(x\,;\,K, o_v)$ is constructed from a combinations of the parts in K, and the

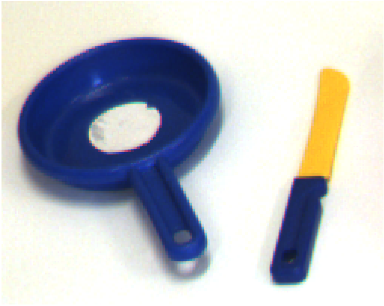


Figure 5.1: Object library: toy pan an knife

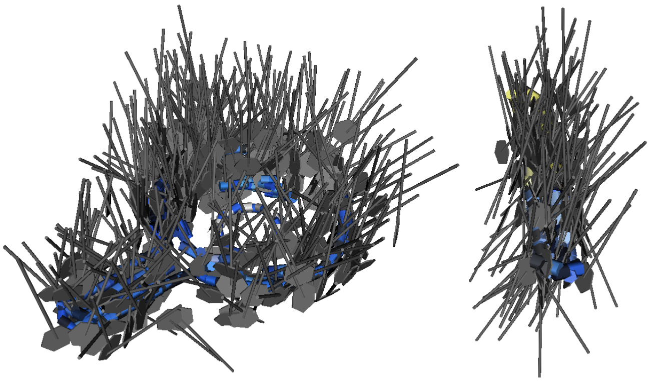


Figure 5.2: Samples from empirical grasp densities learned with the objects of Figure 5.1.

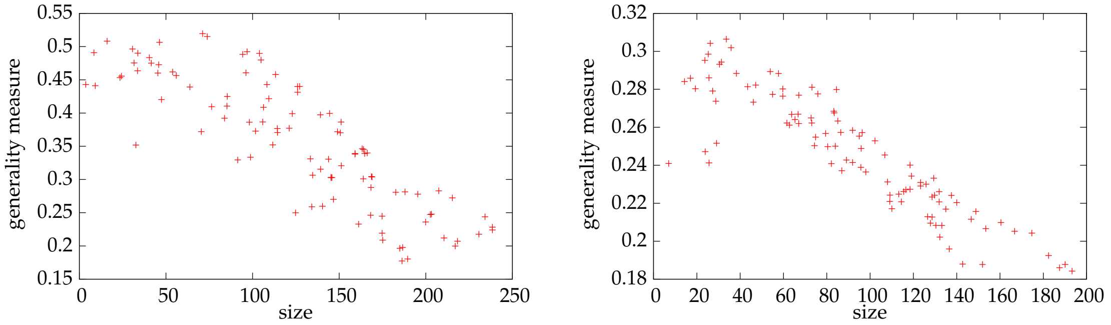
contribution of each part p is weighted by its visual (or shape) resemblance with o_v. For instance, let us consider a set K containing two parts that model a cylinder ($p^{(1)}$) and a handle ($p^{(2)}$), and let o_v correspond to a mug. Through the integral above (5.7), the region of $h(x\,;\,K, o_v)$ surrounding the cylinder of the mug will be computed from $p^{(1)}$ only, while the region surrounding the handle will be computed form $p^{(2)}$ only. Means of numerically approximating the integrals defined above are detailed in Section 2.3.

5.4 Experimental Results

This section presents preliminary results on part generalization. Section 5.4.1 illustrates the generality measure defined above on two object models learned in Chapter 4. Section 5.4.2 presents an experiment in which initial densities are learned from generic parts. Results obtained in simulation show that these densities have a higher success rate than vision-based initial densities.

5.4.1 Generalization with Models Learned by a Robot

This section illustrates the generality measure defined above on two object models learned in Chapter 4. These objects are the pan and knife of Figure 5.1. Each model is



(a) Parts segmented from the pan

(b) Parts segmented from the knife

Figure 5.3: Generality measure of the parts of the pan with respect to the model of the knife (Figure (a)) and of the parts of the knife with respect to the model of the pan (Figure (b)). The measure of generality is plotted as a function of part sizes. See text for details.

composed of a visual model in the form of a single edge-segment density (see Chapter 3), and a grasp model in the form of an empirical grasp density. They are illustrated in Figure 5.2.

One hundred parts were randomly segmented from the model of the pan, and one hundred from the model of the knife. The ability of each part p of the pan to predict the empirical density of the knife was then computed using the generality measure $m(p, \{\text{knife}, \text{pan}\}) = f(p, \text{knife})$ defined above (5.6). Likewise, the generality measure of the pan model and each of the one hundred parts segmented from the knife was computed. An interesting way to plot these results is to show the generality measure as a function of the spatial size of the corresponding parts. In Figure 5.3a, each point corresponds to one of the 100 candidate parts segmented from the pan. The abscissa a point gives the spatial size of the corresponding part; the size of a part corresponds to the diameter of the smallest sphere that encloses all the particles from the part's visual and grasp models. The ordinate of a point gives the generality measure of the corresponding part. Figure 5.3b shows a similar plot for parts segmented from the knife.

The first observation to make is that as parts grow larger, the generality measure eventually decreases. This result is rather natural, as the two objects we are considering have different overall shapes. It is also interesting to note that, at least in Figure 5.3b, the parts with the highest generality measures are not the smallest ones. Figure 5.4 shows the knife part with the highest generality measure – it corresponds to the highest point of Figure 5.3b. This part corresponds to a segment of about 1cm from the handle of the knife.

Figure 5.5 shows three parts of the pan. The generality measure of the first two is high. The part of Figure 5.5a is at coordinates $(71.2, 0.519)$ in Figure 5.3a – its generality measure is 0.519, and the radius of its bounding sphere is 71.2mm. The part of Figure 5.5b is at coordinates $(16, 0.508)$. By contrast, the part of Figure 5.5c, which is rather large (186mm), has a low generality measure (0.177). These results correspond

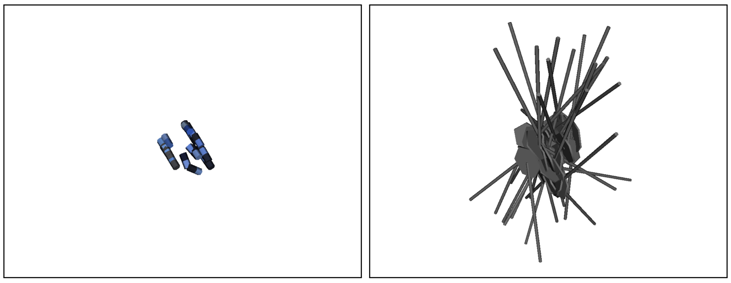


Figure 5.4: Visuomotor model of a part segmented from the knife. The part corresponds to a small segment of the handle of the knife. The left image shows the visual component p_v of the part. The right image shows the associated grasp component p_g.

to what we would expect from a generality measure: the parts of Figure 5.5a and Figure 5.5b seem to contain information relevant to the knife, while the part of Figure 5.5c couldn't be properly fitted to it.

5.4.2 Generalization in a Simulated Environment

The experiments of the previous section illustrate the behavior of the generality measure presented above (5.6). In the grasp learning context, the purpose of identifying parts with a high generality measure is to exploit them to form initial densities for novel objects (5.7). When used in exploratory learning (see Chapter 4), these initial densities should hopefully yield success rates higher than the rates obtained with vision-based initial densities. In order to show this, we should construct initial densities, e.g., from the parts of Figure 5.4, Figure 5.5a, and Figure 5.5b, and repeat a grasp-learning experiment similar to that of Chapter 4. This experiment is beyond the time frame of this dissertation. Instead, this section provides results obtained in simulation.

In this section, we compare the success rate of grasp densities created from visual cues to the success rate of densities created from generic object parts. This experiment works as follows:

i. We create visual models (surface-point densities) for four virtual objects (Figure 5.6).

ii. We create initial grasp densities from the visual models of the four objects.

iii. We simulate the learning of empirical grasp densities for these four objects. Empirical densities are required for learning the visuomotor patterns shared by the objects. Learning empirical densities also provides us with an estimate of the success rate of the initial densities created from visual models.

iv. We evaluate, for each subset of exactly three objects, the generality measures of arbitrarily segmented parts.

v. We create an initial density for each object. The initial density of an object o is created as defined by Eq. 5.7, with K containing the part that shows the highest generality measure across the other three objects. As a result, the selection of the

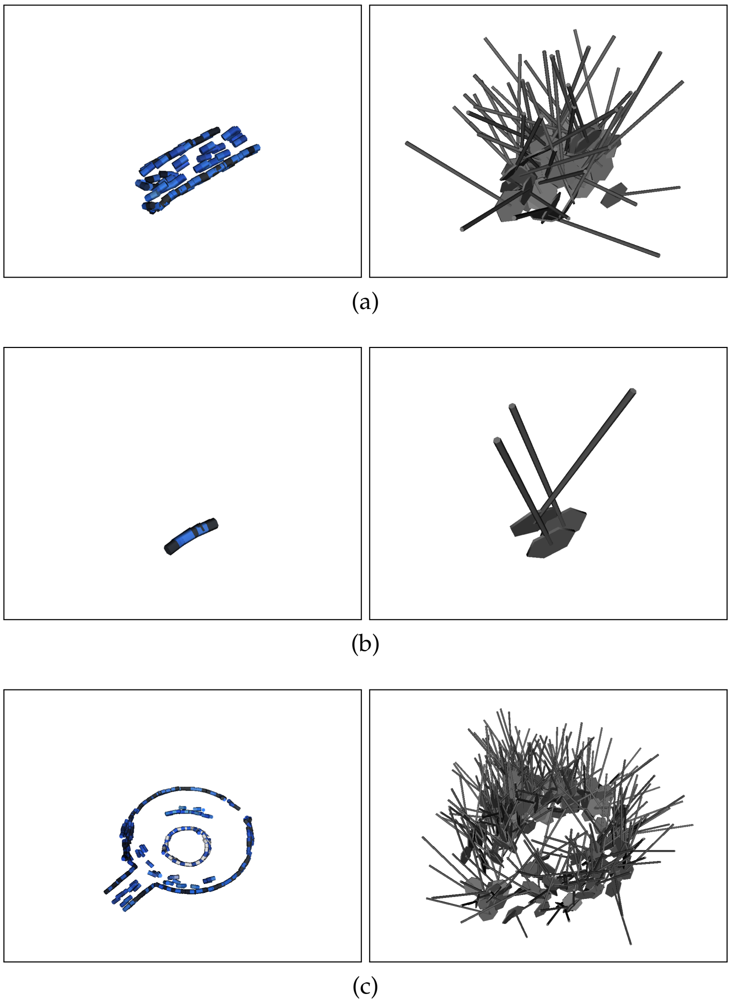


Figure 5.5: Visuomotor models of parts segmented from the pan. Figure (a) corresponds to a small segment of the handle of the pan. Figure (b) corresponds to an even smaller segment of the handle, while Figure (c) is a model of almost all of the object. Images on the left show the visual component of each model, while images on the right show the grasp component.

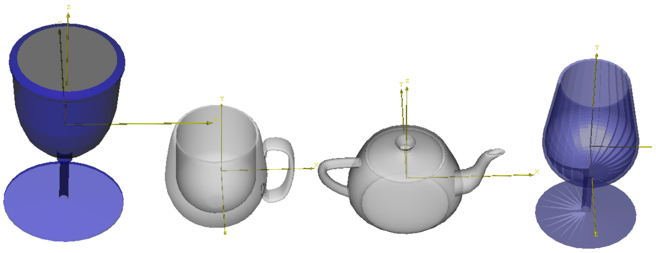


Figure 5.6: Mesh models of the four objects used in this experiment: a goblet, a mug, a teapot, and a wine glass.

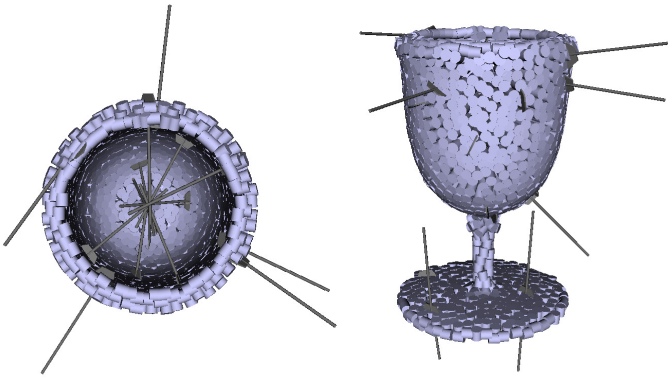


Figure 5.7: Illustration of the visual model and vision-based initial density of the goblet, viewed from the top (left) and from the side (right). The blue cylinders correspond to the particles supporting the visual model of the goblet. The axis of a cylinder represents the orientation of the associated particle. The figure also shows in gray ten of the particles of the initial density created from the visual model.

part that is used for constructing the initial density of o is completely independent of o (o can be considered as a “novel” object that initially has only a visual model).

vi. We simulate the learning of empirical densities using the generalization-based initial models. Our results show that the success rate of this process is about three times the success rate of vision-based initial densities.

Vision-based Initial Densities

To create a visual model (see point i above) from a mesh model (Figure 5.6), we sample a set of points from the meshed surface, and we turn these points into a single surface-point density as described in Chapter 3. The resulting model is supported by a set of $\mathbb{R}^3 \times S^2$ particles distributed along object faces (see Figure 5.7). Each particle has a 3D position, and a 2DOF orientation which corresponds to the local surface normal. For clarity, the text below refers to these 5DOF particles as *surface patches*.

Mathematically, surface-oriented visual models are identical to those created from sparse-stereo data. A bootstrapping procedure (point ii above) similar to the one of Chapter 4 can thus be applied: Constructing an initial density works by defining a large set of grasps approaching normally to the surface patches supporting the object’s visual model. The patches supporting the visual object model have a 2–degree-of-freedom orientation (modeling the local surface normal), whereas a grasp orientation has 3DOF. Each surface patch thus yields a set of grasps approaching normally to the patch, with the rotation of the hand around the wrist selected from a uniform distribution on $[0, 2\pi[$. The resulting grasps are directly used as particles supporting the initial density. Figure 5.7 shows some of the particles supporting the initial density created for the goblet of Figure 5.6.

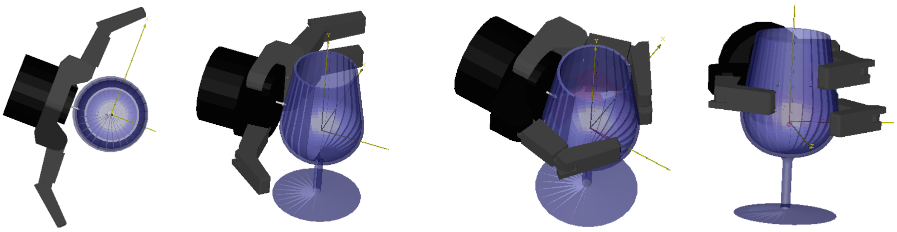


Figure 5.8: Grasping an object in *GraspIt!*. According to the ϵ measure described in the text, the quality of this grasp is 0.24.

Vision-based Exploratory Learning

From a theoretical viewpoint, the simulated learning procedure (iii) is identical to the one followed in Chapter 4: grasps sampled randomly from an initial density are executed, and an empirical density is constructed from the successful grasps using an importance-sampling algorithm.

Simulated learning of grasp densities differs from the experiment of Chapter 4 in the way grasps are executed. Grasp are executed in the *GraspIt!* simulator [Miller and Allen, 2004] using a Barrett hand model (Figure 5.8). The simulated environment contains only the object and the hand. The object's pose is fixed and known to the robot. The hand is free-floating; the rest of the robot is not modeled. The simulator does not model dynamics: When closing the hand, fingers stop as soon as they make contact with the object. There is no gravity. By contrast to the experiment of Chapter 4, there is no need for pose estimation or path planning.

An empirical density is learned by executing a set of grasp trials. Each trial involves the following operations:

1. Draw a grasp sample from the object's initial density.

2. Place the fully open hand at the pose defined by the grasp sample. If the model of the hand intersects with the object, the grasp is a failure.

3. Try to move the hand 5mm forward, in the direction of the wrist. If the hand makes contact with the object, it stops. (The height of the wine glass of Figure 5.6 is 10cm.)

4. Close the fingers until they make contact with the object.

5. Evaluate the quality of the grasp.

The success of a grasp is computed in *GraspIt!* using the "ϵ" force-closure quality measure formulated by Ferrari and Canny [1992]. The ϵ force-closure quality measure studies contact forces to characterize the effort the robot has to make to maintain force-closure under a worst-case external disturbance. The value of ϵ can vary between 0 and 1, with $\epsilon = 1$ corresponding to a very good grasp. In our experiment, a grasp is successful if $\epsilon > 0.05$. Force-closure grasps are difficult to achieve with few contact points. This explains why the hand is moved forward during grasp executions: it

Object	Successful Grasps	Tot. N. Grasps	Success Rate
Goblet	9236	157282	5.9%
Mug	15119	119913	12.6%
Teapot	11453	136415	8.4%
Wine Glass	8096	126254	6.4%

Table 5.1: Learning with a vision-based initial density: success statistics.

Object	Successful Grasps	Tot. N. Grasps	Success Rate
Goblet	4948	23645	21%
Mug	5756	15191	37.9%
Teapot	4715	17712	26.6%
Wine Glass	3265	18267	17.9%

Table 5.2: Learning with a generalization-based initial density: success statistics.

increases the chance of creating an additional contact point, for instance by allowing the palm of the hand to touch the object.

Empirical densities were learned for the four objects shown above through the simulation of 539864 grasps. Success rates are shown in Table 5.1.

Generalization-based Initial Densities and Exploratory Learning

The rest of the experiment is organized as a leave-one-out cross-validation. We consider three of the four objects of Figure 5.6. The fourth object, let us denote it by o, is left out for testing. From the three objects we generate a set of one hundred visuomotor patterns (Section 5.3.1), and we compute their generality measure (Section 5.3.2). We then select the pattern with the highest generality measure, and we use it to construct an initial density for the fourth object o (iv, v, illustrated in Figure 5.11). We then simulate in *GraspIt!* a number of grasps sampled from this initial density (vi). This process is repeated four times, so that each of the four objects is used for testing once. The four parts which present the highest generality measure across one subset of three objects are shown in Figure 5.9 and Figure 5.10.

The statistics of generalization-based learning are shown in Table 5.2. As shown in Table 5.3, the success rates are on average three times higher than those of Table 5.1.

	Goblet	**Mug**	**Teapot**	**Wine Glass**
Vision-based Exploration	5.9%	12.6%	8.4%	6.4%
Generalization-based Exploration	21%	37.9%	26.6%	17.9%

Table 5.3: Success rates for vision-based and generalization-based exploratory learning.

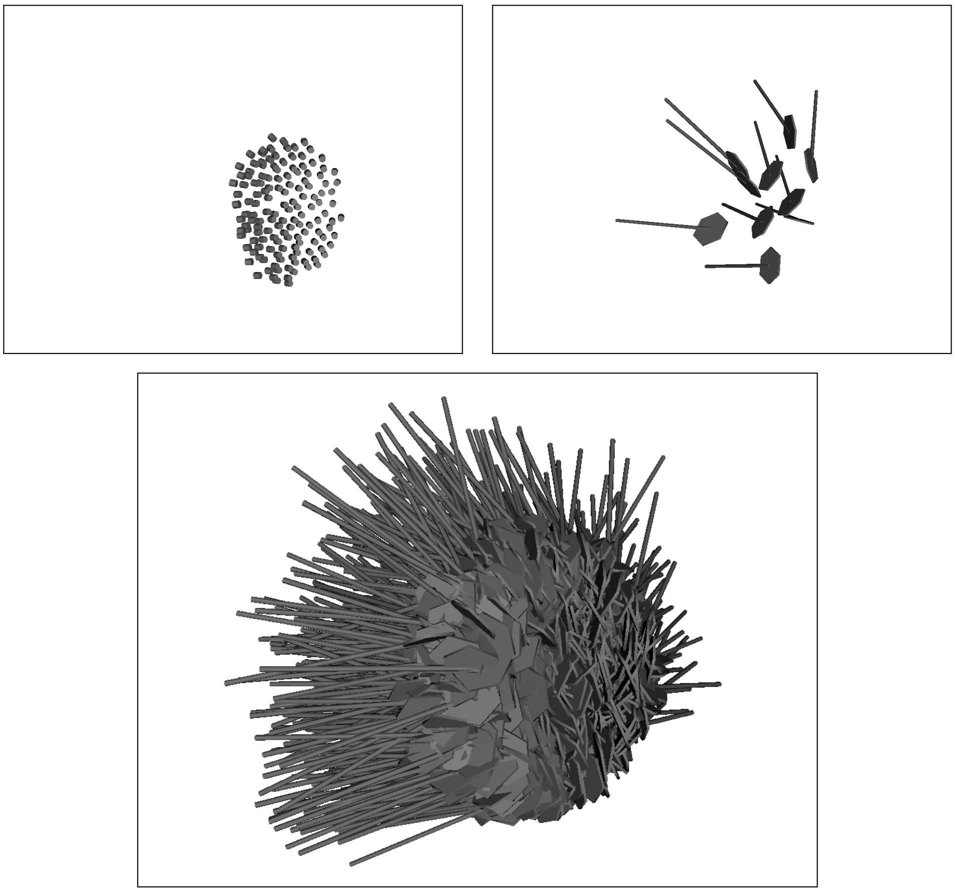
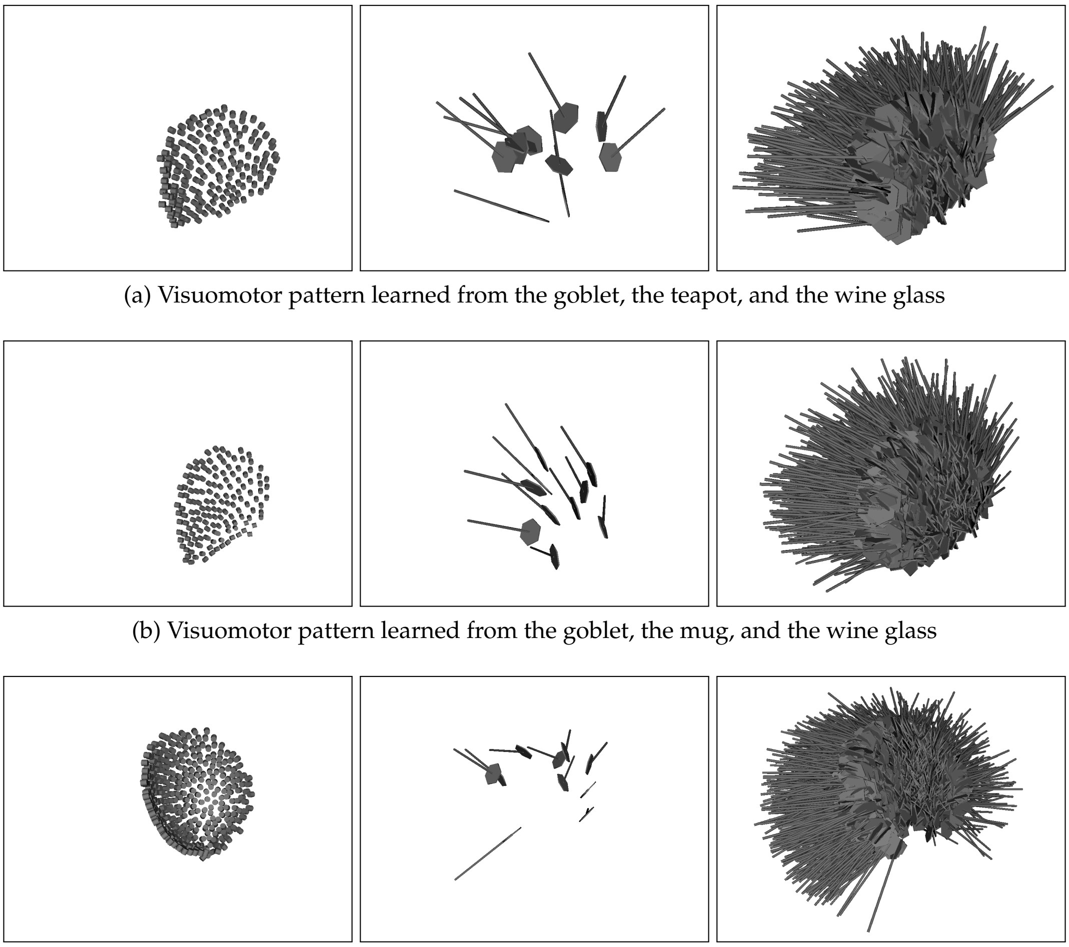


Figure 5.9: Visuomotor pattern learned from the mug, the teapot, and the wine glass. The top-left image corresponds to the visual component of the part. The other two images illustrate the grasp component. The bottom image shows a large number of samples; the top-right image shows only ten. This part is the one that yields the highest generality measure across the mug, the teapot, and the wine glass.



(a) Visuomotor pattern learned from the goblet, the teapot, and the wine glass

(b) Visuomotor pattern learned from the goblet, the mug, and the wine glass

(c) Visuomotor pattern learned from the goblet, the mug, and the teapot

Figure 5.10: Generic parts. Each triplet of images illustrates a generic part. The left image corresponds to the visual component of the part. The middle and right-side images illustrate the grasp component. The right-side image shows a large number of samples; the middle image shows only ten. The three parts illustrated in this figure correspond to the parts of highest generality measure across each of the corresponding groups of three training objects. The fourth part is shown in Figure 5.9.

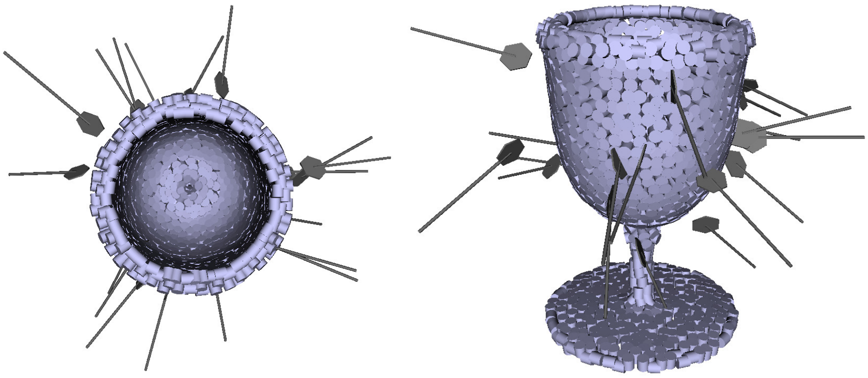


Figure 5.11: Illustration of the generalization-based initial density of the goblet. The figure shows in gray ten grasps sampled from the initial density created from the pattern of Figure 5.9.

Discussion

In several respects, this experiment is rather simple. The number of objects is rather small, and they share one obvious common part – a round-shaped "bowl". Also, initial densities are constructed from a single generic pattern. Nonetheless, our results make explicit two strong points of the approach. First, the patterns selected by the generality measure of Section 5.3.2 seem intuitively pertinent. Their visual models include enough structure to encode the curvature of the underlying surface and correctly align grasps to similar shapes, while excluding structures that are not common to all objects. Second, even if each initial density is built from a single pattern, the success rates of Table 5.3 demonstrate the applicability of the bootstrapping procedure defined by Eq. 5.7.

5.5 Conclusion

Generalizing grasps across objects is a crucial part of grasp learning, as it allows us to quickly adapt to smooth environmental changes. We have presented a method that allows an autonomous agent to efficiently grasp novel objects by applying grasps related to parts shared by previously-acquired objects. Parts that are likely to help with grasping novel objects are identified offline, by extracting arbitrary visuomotor patterns from known models, and counting how often they are observed in other known models. The grasps associated to these patterns are transferred to a novel object by instantiating them onto object regions that visually (or structurally) predict their applicability. They are eventually adapted to the precise morphology of the object through autonomous exploration.

Our experiments demonstrate that the parts emerging from the generalization process are conform to what we would expect: they include enough visual support to be robustly applied, and they exclude patterns that are not common to several objects.

Chapter 6

Conclusion

The amplitude of the uncertainty inherent to human environments motivates robots that learn and adapt to their environment. The robotics community is presently moving in this direction. Adaptive components are progressively being developed and deployed throughout the entire robot control stack.

In this work, we focused on the problem of grasping visually-observed objects. We presented a 3D object model for autonomous, visuomotor interaction. The model is learned autonomously by experiencing the correlation between successful grasps and visual structure. With time, it becomes increasingly efficient at inferring grasp parameters from visual evidence. This behavior relies on (1) a grasp model representing the grasp success likelihood of relative hand-object configurations, and (2) a model of visual object structure, which aligns the grasp model to arbitrary object poses (3D positions and orientations).

Our visual object model is defined as a hierarchy of parts. Low-level parts encode sensor data with probability density functions. Each low-level part models a number of object edges or faces by representing the spatial distribution of short segments or patches from these edges or faces, respectively. Higher-level parts encode the spatial distribution of more elementary parts. Evidence for edges and faces is obtained from a 3D scanner or a sparse-stereo setup; evidence is turned into probability distributions through kernel density estimation. The model readily allows one to compute the 6DOF pose distribution of an object within an arbitrary scene by propagating scene evidence through the model with the belief propagation algorithm. BP messages and local maxima of the posterior pose distribution are computed through Monte Carlo simulation. A model is learned from a point-cloud or edge reconstruction of an object in a bottom-up fashion, by first defining bottom-level parts through observation clustering and kernel density estimation, and then iteratively combining parts together to form a hierarchy. A complete 3D model can be learned from a set of unregistered views of an object – novel views are incrementally aligned and fused with the model.

Our grasp affordance model represents the likelihood of success of relative object-gripper poses with a kernel-based density function defined on the space of 6DOF poses. Grasp densities are learned through a combination of cross-object generalization, visual inference, autonomous exploration, and imitation. The density of a novel object is initially constructed from visual cues or by observing a teacher. This initial density is then closely adapted to the object's morphology through autonomous exploration: grasps sampled randomly from the initial model are performed, and an importance-sampling algorithm learns an empirical density from the outcomes of these experiences. If a new object resembles objects the robot has already worked with, its model can also be constructed by applying grasps related to parts shared by the new objects and the objects the robots knows already. Parts that are likely to help with grasping novel objects are identified offline, by extracting arbitrary visuomotor patterns from known models, and counting how often they are observed in other known models. To robustly grasp an object, its model is visually aligned to the correct object pose. The aligned grasp density is then combined to reaching constraints to select the maximum-likelihood achievable grasp. Figure 6.1 shows a system diagram of the visual and grasp models working together.

The applicability of our model is supported by numerous examples of pose estimates in cluttered scene, and by a robot platform that shows increasing grasping performances as it explores its environment.

The main contributions of our work are summarized below:

1. We have formulated and efficiently implemented a nonparametric representation of $SE(3)$ probability density functions.

2. We have defined density-based models of low-level visual descriptors and kinematic grasp parameters, along with methods for object pose estimation and robotic grasping.

3. We have developed means of learning both models from experience.

4. We have developed a software library that implements our visual and grasp models. We have contributed to the deployment of this program onto two robotic platforms. These platforms have empirically demonstrated the feasibility of our approach through an experiment that integrates all aspects of our work.

Chapters 3 and 4 establish two links between the visual and grasp models: initial densities are computed from the visual model, and visual pose estimation aligns the grasp model to the correct object pose. Despite these links, results similar to those of Chapter 4 could be obtained without our visual model, replacing it with a standard pose estimation method and using human-demonstrated initial densities. It is principally in Chapter 5 that both models become integrated into a coherent visuomotor solution, as the generative, probabilistic approach to modeling visual structure and grasp parameters is instrumental to the proposed grasp generalization method.

Future work has been discussed in detail in the previous chapters. The most promising perspectives are summarized below.

1. Parsing objects into parts, which allow for efficient and robust modeling of large object libraries, is a promising research avenue. Chapter 3 presents a flexible hierarchical approach to object modeling. However, the hierarchies used in Chapter 3 are rather simple, and the model is potentially capable of hosting more complex part combinations. We are interested in using this model as a basis for studying novel means of parsing objects. One possible avenue would be to use the visual

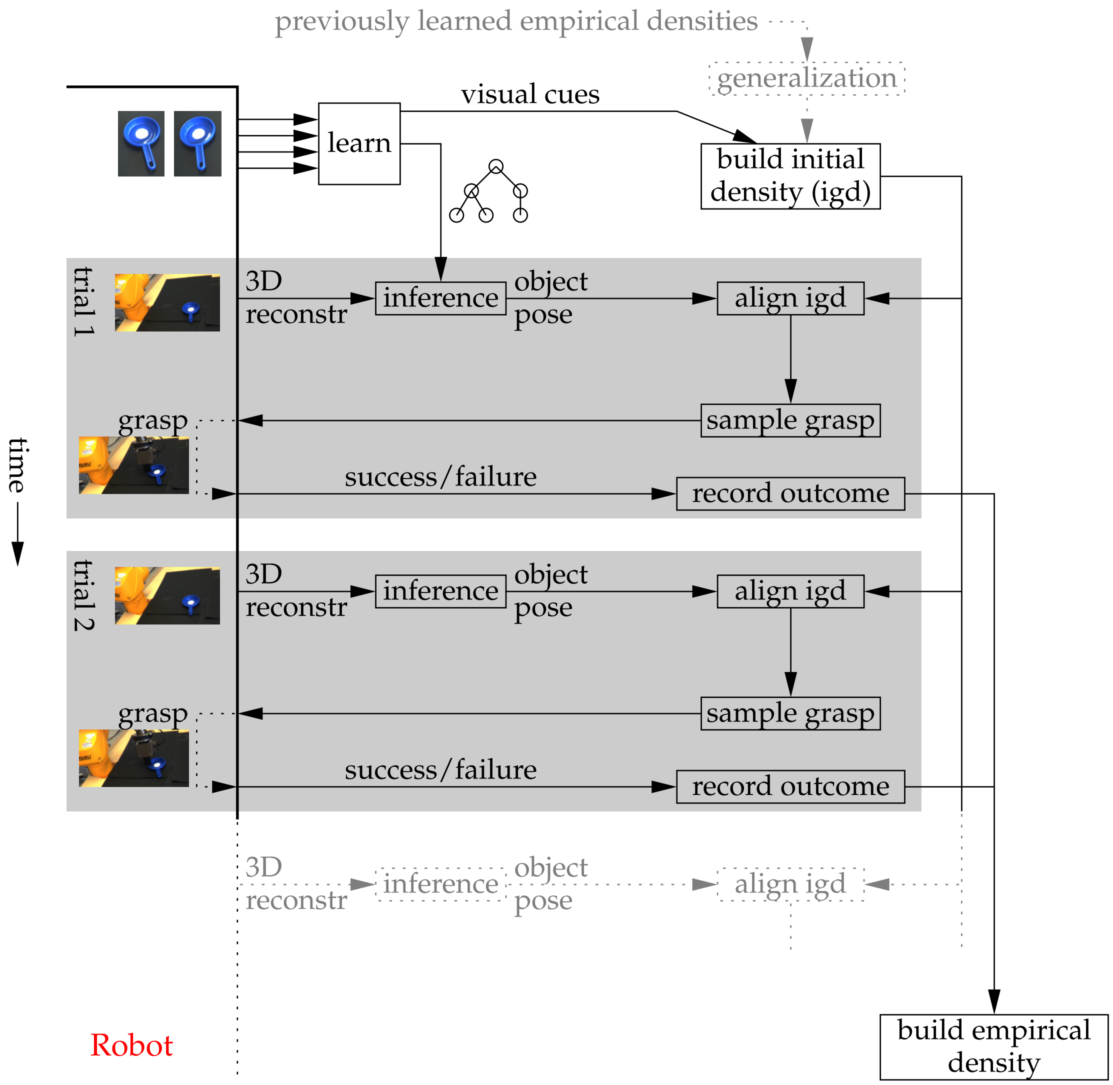


Figure 6.1: Visual and grasp models working together to form a grasp learning system. The left side of the figure represents the robot, through which the agent interacts with the world. The right side of the figure corresponds to the grasp model. Visual components are scattered in-between. The agent initially acquires a set of images of an object and builds a visual hierarchical model of it. Also, visual cues from the object help the agent build a vision-based initial grasp density. The agent then starts interacting with the object. The agent repeatedly executes grasps, by estimating the pose of the object, aligning the initial grasp density to it, sampling a grasp from the density, and performing the grasp with the robot. The set of tested grasps is eventually turned into an empirical density. When several empirical densities are available, subsequent initial densities can be constructed through generalization (see Chapter 5).

components of the parts learned during grasp generalization as an elementary part vocabulary for visual object models.

2. Currently, grasp densities exclusively model gripper poses. We are interested in extending the representation to preshape information, e.g., using preshape dimensionality reduction to limit the number of additional parameters [Ciocarlie and Allen, 2009].

3. An interesting aspect of grasp affordance models is their representation of a large variety of grasp strategies. This type of information becomes very useful in grasp planning, when both object- and environment-related constraints need to be satisfied. For instance, as illustrated in Chapter 4, a planner may combine a grasp model with reaching constraints to select the most promising achievable grasp. Continuing in this direction can potentially lead to important results, as demonstrated by Gienger et al. [2008].

4. Chapter 4 focuses on a generative model, which is then exploited in Chapter 5 for grasp generalization. The data collected during the experiments of Chapters 4 and 5 could potentially be used to learn discriminative grasp models – i.e., using the notation of Chapter 4, $P(O = o|X = x)$. It would be interesting to see how these compare in practice to the results we have obtained.

5. The experimental validation of Chapter 5 is rather limited. We are planning to extend it to more objects. We would also like to study whether grasp densities learned in simulation can be transferred to a real robot.

Bibliography

M. Abramowitz and I. Stegun. *Handbook of mathematical functions: with formulas, graphs, and mathematical tables*. Courier Dover Publications, 1965.

P. Allen, A. Miller, P. Oh, and B. Leibowitz. Integration of vision, force and tactile sensing for grasping. *Int. J. Intelligent Machines*, 4(1):129–149, 1999.

E. Alpaydin. *Introduction to machine learning*. The MIT Press, 2004.

B. Argall, E. Sauser, and A. Billard. Demonstration, tactile correction and multiple training data sources for robot motion control. In *Learning from Multiple Sources with Applications to Robotics (Workshop at Neural Information Processing Systems)*, 2009.

M. Beetz, O. Brock, G. Cheng, and J. Peters. 09341 summary – cognition, control and learning for robot manipulation in human environments. In M. Beetz, O. Brock, G. Cheng, and J. Peters, editors, *Cognition, Control and Learning for Robot Manipulation in Human Environments*, number 09341 in Dagstuhl Seminar Proceedings, Dagstuhl, Germany, 2010. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany.

A. Bhattacharyya. On a measure of divergence between two statistical populations defined by their probability distributions. *Bulletin of the Calcutta Mathematical Society*, 1943.

A. Bicchi and V. Kumar. Robotic grasping and contact: a review. In *IEEE International Conference on Robotics and Automation*, 2000.

A. Billard, S. Calinon, R. Dillmann, and S. Schaal. Robot programming by demonstration. In Siciliano and Khatib [2008], pages 1371–1394. ISBN 978-3-540-23957-4.

O. Boiman and M. Irani. Similarity by composition. In *Neural Information Processing Systems*, 2006.

C. Borst, M. Fischer, and G. Hirzinger. Grasping the dice by dicing the grasp. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 4, pages 3692–3697, 2003.

G. E. P. Box and M. E. Muller. A note on the generation of random normal deviates. *The Annals of Mathematical Statistics*, 29(2):610–611, 1958.

R. Caflisch. Monte carlo and quasi-monte carlo methods. *Acta Numerica*, 7:1–49, 1998.

M. T. Ciocarlie and P. K. Allen. Hand posture subspaces for dexterous robotic grasping. *Int. J. Rob. Res.*, 28(7):851–867, 2009. ISSN 0278-3649. doi: http://dx.doi.org/10.1177/0278364909105606.

J. Coelho, J. Piater, and R. Grupen. Developing haptic and visual perceptual categories for reaching and grasping with a humanoid robot. In *Robotics and Autonomous Systems*, volume 37, pages 7–8, 2000.

A. Collet, D. Berenson, S. Srinivasa, and D. Ferguson. Object recognition and full pose registration from a single image for robotic manipulation. In *IEEE International Conference on Robotics and Automation*, 2009.

G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *Statistical Learning in Computer Vision (Workshop at the European Conference on Computer Vision)*, volume 1, page 22, 2004.

M. R. Cutkosky, R. D. Howe, and W. R. Provancher. Force and tactile sensors. In Siciliano and Khatib [2008], pages 455–476. ISBN 978-3-540-23957-4.

K. Daniilidis and J.-O. Eklundh. 3-D vision and recognition. In Siciliano and Khatib [2008], pages 543–562. ISBN 978-3-540-23957-4.

C. de Granville, J. Southerland, and A. H. Fagg. Learning grasp affordances through human demonstration. In *IEEE International Conference on Development and Learning*, 2006.

Y. Demiris and A. Dearden. From motor babbling to hierarchical learning by imitation: a robot developmental pathway. In *International Workshop on Epigenetic Robotics: Modeling Cognitive Development in Robotic Systems*, pages 31–37, 2005.

A. Dempster, N. Laird, D. Rubin, et al. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39 (1):1–38, 1977.

R. Detry and J. Piater. Continuous surface-point distributions for 3D object pose estimation and recognition. In *Asian Conference on Computer Vision*, 2010.

R. Detry and J. H. Piater. Hierarchical integration of local 3D features for probabilistic pose recovery. In *Robot Manipulation: Sensing and Adapting to the Real World (Workshop at Robotics, Science and Systems)*, 2007.

R. Detry, D. Kraft, O. Kroemer, L. Bodenhagen, J. Peters, N. Krüger, and J. Piater. Learning grasp affordance densities. *Paladyn. Journal of Behavioral Robotics*, (submitted).

R. Detry, N. Pugeault, and J. H. Piater. Probabilistic pose recovery using learned hierarchical object models. In *International Cognitive Vision Workshop (Workshop at the 6th International Conference on Vision Systems)*, pages 107–120, Berlin, Heidelberg, 2008. Springer-Verlag. doi: 10.1007/978-3-540-92781-5_9.

R. Detry, E. Başeski, N. Krüger, M. Popović, Y. Touati, O. Kroemer, J. Peters, and J. Piater. Learning object-specific grasp affordance densities. In *IEEE International Conference on Development and Learning*, pages 1–7, 2009a. doi: 10.1109/DEVLRN.2009.5175520.

R. Detry, E. Başeski, N. Krüger, M. Popović, Y. Touati, and J. Piater. Autonomous learning of object-specific grasp affordance densities. In *Approaches to Sensorimotor Learning on Humanoid Robots (Workshop at the IEEE International Conference on Robotics and Automation)*, 2009b.

R. Detry, N. Pugeault, and J. Piater. A probabilistic framework for 3D visual object representation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(10):1790–1803, 2009c. doi: 10.1109/TPAMI.2009.64.

R. Detry, E. Başeski, M. Popović, Y. Touati, N. Krüger, O. Kroemer, J. Peters, and J. Piater. Learning continuous grasp affordances by sensorimotor exploration. In O. Sigaud and J. Peters, editors, *From Motor Learning to Interaction Learning in Robots*, pages 451–465. Springer-Verlag, 2010a. doi: 10.1007/978-3-642-05181-4_19.

R. Detry, D. Kraft, A. G. Buch, N. Krüger, and J. Piater. Refining grasp affordance models by experience. In *IEEE International Conference on Robotics and Automation*, pages 2287–2293, 2010b. doi: 10.1109/ROBOT.2010.5509126.

R. Douc, O. Cappe, and E. Moulines. Comparison of resampling schemes for particle filtering. *International Symposium on Parallel and Distributed Processing and Applications*, 2005:64, 2005.

S. Ekvall and D. Kragic. Interactive grasp learning based on human demonstration. In *IEEE International Conference on Robotics and Automation*, 2004.

C. Elkan. Clustering documents with an exponential-family approximation of the Dirichlet compound multinomial distribution. In *International Conference on Machine Learning*, 2006.

A. Erkan, O. Kroemer, R. Detry, Y. Altun, J. Piater, and J. Peters. Learning probabilistic discriminative models of grasp affordances under limited supervision. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010. doi: 10.1109/IROS.2010.5650088.

P. F. Felzenszwalb and D. P. Huttenlocher. Efficient matching of pictorial structures. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2066–, 2000.

R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 264–271, June 2003.

C. Ferrari and J. Canny. Planning optimal grasps. In *IEEE International Conference on Robotics and Automation*, pages 2290–2295, 1992.

S. Fidler and A. Leonardis. Towards scalable representations of object categories: Learning a hierarchy of parts. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

R. A. Fisher. Dispersion on a sphere. In *Proc. Roy. Soc. London Ser. A.*, 1953.

R. B. Fisher and K. Konolige. Range sensors. In Siciliano and Khatib [2008], pages 521–542. ISBN 978-3-540-23957-4.

J. J. Gibson. *The Ecological Approach to Visual Perception*. Lawrence Erlbaum Associates, 1979.

M. Gienger, M. Toussaint, and C. Goerick. Task maps in humanoid robot manipulation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2758–2764, 2008.

C. Goldfeder, M. Ciocarlie, H. Dang, and P. Allen. The Columbia grasp database. In *IEEE International Conference on Robotics and Automation*, 2009.

K. Hosoda, Y. Tada, and M. Asada. Internal representation of slip for a soft finger with vision and tactile sensors. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 1, pages 111–15, 2002.

D. Hubel and T. Wiesel. Anatomical demonstration of columns in the monkey striate cortex. *Nature*, 221:747–750, 1969.

M. Ito, K. Noda, Y. Hoshino, and J. Tani. Dynamic and interactive generation of object handling behaviors by a small humanoid robot using a dynamic neural network model. *Neural Networks*, 19(3):323–337, April 2006. ISSN 08936080. doi: 10.1016/j.neunet.2006.02.007.

A. E. Johnson and M. Hebert. Using spin images for efficient object recognition in cluttered 3D scenes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21:433–449, 1999.

B. Julesz. Textons, the elements of texture perception, and their interactions. *Nature*, 290(5802):91–97, 1981.

C. Kemp, A. Edsinger, and E. Torres-Jara. Challenges for robot manipulation in human environments. *IEEE Robotics and Automation Magazine*, 14(1):20, 2007a.

C. C. Kemp, A. Edsinger, R. Platt, and N. E. Sian, editors. *Proceedings of the Robotics: Science & Systems 2007 Manipulation Workshop – Sensing and Adapting to the Real World*, 2007b.

J. Kober, K. Mülling, O. Krömer, C. Lampert, B. Schölkopf, and J. Peters. Movement templates for learning of hitting and batting. In *IEEE International Conference on Robotics and Automation*, 2010.

D. Kraft, E. Başeski, M. Popović, A. M. Batog, A. Kjær-Nielsen, N. Krüger, R. Petrick, C. Geib, N. Pugeault, M. Steedman, T. Asfour, R. Dillmann, S. Kalkan, F. Wörgötter, B. Hommel, R. Detry, and J. Piater. Exploration and planning in a three-level cognitive architecture. In *International Conference on Cognitive Systems (Workshop at the IEEE International Conference on Robotics and Automation)*, 2008. Extended Abstract.

D. Kraft, R. Detry, N. Pugeault, E. Başeski, J. Piater, and N. Krüger. Learning objects and grasp affordances through autonomous exploration. In *International Conference on Computer Vision Systems*, volume 5815/2009, pages 235–244, 2009. doi: 10.1007/978-3-642-04667-4_24.

D. Kraft, R. Detry, N. Pugeault, E. Başeski, F. Guerin, J. Piater, and N. Krüger. Development of object and grasping knowledge by robot exploration. *IEEE Transactions on Autonomous Mental Development*, 2010. doi: 10.1109/TAMD.2010.2069098.

D. Kragic, A. T. Miller, and P. K. Allen. Real-time tracking meets online grasp planning. In *IEEE International Conference on Robotics and Automation*, pages 2460–2465, 2001.

O. Kroemer, R. Detry, J. Piater, and J. Peters. Active learning using mean shift optimization for robot grasping. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2610–2615, 2009. doi: 10.1109/IROS.2009.5354345.

O. Kroemer, R. Detry, J. Piater, and J. Peters. Adapting preshaped grasping movements using vision descriptors. In *From Animals to Animats 11 – International Conference on the Simulation of Adaptive Behavior*, 2010a. doi: 10.1007/978-3-642-15193-4_15.

O. Kroemer, R. Detry, J. Piater, and J. Peters. Grasping with vision descriptors and motor primitives. In *International Conference on Informatics in Control, Automation and Robotics*, 2010b.

O. Kroemer, R. Detry, J. Piater, and J. Peters. Combining active learning and reactive control for robot grasping. *Robotics and Autonomous Systems*, 2010c. doi: 10.1016/j.robot.2010.06.001.

N. Krüger, M. Lappe, and F. Wörgötter. Biologically Motivated Multi-modal Processing of Visual Primitives. *The Interdisciplinary Journal of Artificial Intelligence and the Simulation of Behaviour*, 1(5):417–428, 2004.

N. Krüger, J. Piater, C. Geib, R. Petrick, M. Steedman, F. Wörgötter, A. Ude, T. Asfour, D. Kraft, D. Omrčen, A. Agostini, and R. Dillmann. Object-action complexes: Grounded abstractions of sensorimotor processes. *Robotics and Autonomous Systems*, 2010a. (submitted).

N. Krüger, J. Piater, C. Geib, M. Steedman, F. Wörgötter, A. Ude, T. Asfour, and R. Dillmann. Object-action complexes: Grounded abstractions of sensorimotor processes. In *International Conference on Cognitive Systems*, 2010b. (extended abstract).

J. Kuffner. Effective sampling and distance metrics for 3D rigid body path planning. In *IEEE International Conference on Robotics and Automation*. IEEE, May 2004.

Y. Kuniyoshi, M. Inaba, and H. Inoue. Learning by watching: Extracting reusable task knowledge from visual observation of human performance. *IEEE Transactions on Robotics and Automation*, 10(6):799–822, 1994.

M. Lee and H. Nicholls. Review article tactile sensing for mechatronics–a state of the art survey. *Mechatronics*, 9(1):1–31, 1999.

T. S. Lee and D. Mumford. Hierarchical Bayesian inference in the visual cortex. *Journal of the Optical Society of America*, pages 1434–1448, 7 2003.

J. Liebelt, C. Schmid, and K. Schertler. Viewpoint-independent object class detection using 3D feature maps. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

D. G. Lowe. Fitting parameterized three-dimensional models to images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 13:441–450, 1991.

M. Lungarella, G. Metta, R. Pfeifer, and G. Sandini. Developmental robotics: a survey. *Connection Science*, 15(4):151–190, 2003.

L. Meeden and D. Blank. Introduction to developmental robotics. *Connection Science*, 18(2):93–96, 2006.

A. S. Mian, M. Bennamoun, and R. A. Owens. Three-dimensional model-based object recognition and segmentation in cluttered scenes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(10):1584–1601, 2006.

A. Miller and P. Allen. Graspit! a versatile simulator for robotic grasping. *IEEE Robotics & Automation Magazine*, 11(4):110–122, 2004.

A. T. Miller, S. Knoop, H. Christensen, and P. K. Allen. Automatic grasp planning using shape primitives. In *IEEE International Conference on Robotics and Automation*, volume 2, pages 1824–1829, 2003.

T. Moeslund, A. Hilton, and V. Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer vision and image understanding*, 104(2-3):90–126, 2006.

L. Montesano and M. Lopes. Learning grasping affordances from local visual descriptors. In *IEEE International Conference on Development and Learning*, 2009.

L. Natale, F. Orabona, F. Berton, G. Metta, and G. Sandini. From sensorimotor development to object perception. In *IEEE/RAS International Conference on Humanoid Robots*, pages 226–231, 2005.

P. Pastor, H. Hoffmann, T. Asfour, and S. Schaal. Learning and generalization of motor skills by learning from demonstration. In *IEEE International Conference on Robotics and Automation*, pages 1293–1298, Piscataway, NJ, USA, 2009. IEEE Press. ISBN 978-1-4244-2788-8.

J. Piater and R. Detry. 3D probabilistic representations for vision and action. In *Robotics Challenges for Machine Learning II (Workshop at the IEEE/RSJ International Conference on Intelligent Robots and Systems)*, 2008. Extended Abstract.

J. Piater, F. Scalzo, and R. Detry. Vision as inference in a hierarchical markov network. In *International Conference on Cognitive and Neural Systems*, 2008. Extended Abstract.

J. Piater, S. Jodogne, R. Detry, D. Kraft, N. Krüger, O. Kroemer, and J. Peters. Learning visual representations for interactive systems. In *International Symposium on Robotics Research*, 2009.

J. Piater, S. Jodogne, R. Detry, D. Kraft, N. Krüger, O. Kroemer, and J. Peters. Learning visual representations for perception-action systems. *International Journal of Robotics Research*, 2010. doi: 10.1177/0278364910382464.

M. Popović, D. Kraft, L. Bodenhagen, E. Başeski, N. Pugeault, D. Kragic, T. Asfour, and N. Krüger. A strategy for grasping unknown objects based on co-planarity and colour information. *Robotics and Autonomous Systems*, 2010.

N. Pugeault. *Early Cognitive Vision: Feedback Mechanisms for the Disambiguation of Early Visual Representation*. Vdm Verlag Dr. Müller, 2008.

N. Pugeault, F. Wörgötter, and N. Krüger. Visual primitives: Local, condensed, and semantically rich visual descriptors and their applications in robotics. *International Journal of Humanoid Robotics*, 2010. (to appear).

M. Rolf, J. Steil, and M. Gienger. Efficient exploration and learning of whole body kinematics. In *IEEE International Conference on Development and Learning*, 2009.

F. Rothganger, S. Lazebnik, C. Schmid, and J. Ponce. 3D object modeling and recognition using local affine-invariant image descriptors and multi-view spatial constraints. *Int. J. Comput. Vision*, 66(3):231–259, 2006. ISSN 0920-5691. doi: http://dx.doi.org/10.1007/s11263-005-3674-1.

R. Rusu, N. Blodow, Z. Marton, and M. Beetz. Close-range scene segmentation and reconstruction of 3D point cloud maps for mobile manipulation in domestic environments. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1–6. IEEE Press, 2009.

E. Sahin, M. Cakmak, M. R. Dogar, E. Ugur, and G. Ucoluk. To afford or not to afford: A new formalization of affordances towards affordance-based robot control. *Adaptive Behavior*, 2007.

G. Salton and M. McGill. *Introduction to modern information retrieval*. McGraw-Hill New York, 1983.

S. Savarese and L. Fei-Fei. 3D generic object categorization, localization and pose estimation. In *IEEE International Conference on Computer Vision*, pages 1–8, Oct. 2007. doi: 10.1109/ICCV.2007.4408987.

A. Saxena, S. Chung, and A. Ng. Learning depth from single monocular images. In *Neural Information Processing Systems*, 2005.

A. Saxena, J. Driemeyer, and A. Y. Ng. Robotic Grasping of Novel Objects using Vision. *International Journal of Robotics Research*, 27(2):157, 2008.

A. Schneider, J. Sturm, C. Stachniss, M. Reisert, H. Burkhardt, and W. Burgard. Object identification with tactile sensors using bag-of-features. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 243–248, 2009.

B. Siciliano and O. Khatib, editors. *Springer Handbook of Robotics*. Springer, 2008. ISBN 978-3-540-23957-4.

O. Sigaud and J. Peters, editors. *From Motor Learning to Interaction Learning in Robots*, volume 264 of *Studies in Computational Intelligence*. Springer, 2010a. ISBN 978-3-642-05180-7.

O. Sigaud and J. Peters. From motor learning to interaction learning in robots. In O. Sigaud and J. Peters, editors, *From Motor Learning to Interaction Learning in Robots*, pages 1–12. Springer, 2010b.

B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman & Hall/CRC, 1986.

A. Stoytchev. Toward learning the binding affordances of objects: A behavior-grounded approach. In *AAAI Symposium on Developmental Robotics*, 2005.

A. Stoytchev. Learning the affordances of tools using a behavior-grounded approach. In E. Rome et al., editors, *Affordance-Based Robot Control*, volume 4760 of *Lecture Notes in Artificial Intelligence (LNAI)*, pages 140–158. Springer, Berlin / Heidelberg, 2008.

E. B. Sudderth. *Graphical models for visual object recognition and tracking*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2006.

J. D. Sweeney and R. Grupen. A model of shared grasp affordances from demonstration. In *International Conference on Humanoid Robots*, 2007.

Z. Tu and S. Zhu. Image segmentation by data-driven markov chain monte carlo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 657–673, 2002.

C. P. Tung and A. C. Kak. Automatic learning of assembly tasks using a dataglove system. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1995. doi: http://dx.doi.org/10.1109/IROS.1995.525767.

I. Ulusoy and C. Bishop. Generative versus discriminative methods for object recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

A. T. A. Wood. Simulation of the von Mises-Fisher distribution. *Communications in Statistics—Simulation and Computation*, 23(1):157–164, 1994.

F. Wörgötter, A. Agostini, N. Krüger, N. Shylo, and B. Porr. Cognitive agents – a procedural perspective relying on the predictability of Object-Action-Complexes (OACs). *Robot. Auton. Syst.*, 57(4):420–432, 2009. ISSN 0921-8890. doi: http://dx.doi.org/10.1016/j.robot.2008.06.011.