Visual Object Recognition for Mobile Platforms

by

David Paul Meger

B.Sc, University of British Columbia, 2004

M.Sc, McGill University, 2006

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

in

THE FACULTY OF GRADUATE STUDIES
(Computer Science)

The University Of British Columbia
(Vancouver)

July 2013

Abstract

A robot must recognize objects in its environment in order to complete numerous tasks. Significant progress has been made in modeling visual appearance for image recognition, but the performance of current state-of-the-art approaches still falls short of that required by applications. This thesis describes visual recognition methods that leverage the spatial information sources available on-board mobile robots, such as the position of the platform in the world and the range data from its sensors, in order to significantly improve performance. Our research includes: a physical robotic platform that is capable of state-of-the-art recognition performance; a re-usable data set that facilitates study of the robotic recognition problem by the scientific community; and a three dimensional object model that demonstrates improved robustness to clutter. Based on our 3D model, we describe algorithms that integrate information across viewpoints, relate objects to auxiliary 3D sensor information, plan paths to next-best-views, explicitly model object occlusions and reason about the sub-parts of objects in 3D.

Our approaches have been proven experimentally on-board the Curious George robot platform, which placed first in an international object recognition challenge for mobile robots for several years. We have also collected a large set of visual experiences from a robot, annotated the true objects in this data and made it public to the research community for use in performance evaluation. A path planning system derived from our model has been shown to hasten confident recognition by allowing informative viewpoints to be observed quickly. In each case studied, our system demonstrates significant improvements in recognition rate, in particular on realistic cluttered scenes, which promises more successful task execution for robotic platforms in the future.

Preface

Although the work described in this thesis represents the novel contributions of the author, numerous portions of the work have been carried out in collaboration with others and have appeared in co-authored publications. This section will briefly list these collaborative works and the chapters in which their material can be found. Each of the relevant chapters will contain a more detailed description of the shared work.

- Chapter 3 describes the Curious George robot and material published as [MFL$^+$07], [FML$^+$08] and [MFL$^+$08].
- Chapter 4 describes the collection of a robotic dataset documented in [ML12].
- Chapter 6 describes a planning approach that is found in [MGL10].
- Chapter 7 describes an occlusion reasoning approach that is found in [ML11] and [MWSL11].
- Chapter 8 contains unpublished work carried out in collaboration with others.

Please note that throughout this thesis the “we” voice is used to denote work done primarily by the author of this thesis, with support from others as specified in each chapter.

Table of Contents

Abstract . . . . **ii**

Preface . . . . **iii**

Table of Contents . . . . **iv**

List of Tables . . . . **vii**

List of Figures . . . . **viii**

Glossary . . . . **x**

Acknowledgements . . . . **xiii**

1 Introduction **1**

1.1 Thesis Overview 1

1.2 Problem 4

1.3 Contributions and Outline 10

1.4 Chapter Summary 12

2 Related Work **13**

2.1 Single Image Object Recognition 14

2.2 Previous Robot Recognition Systems 23

2.3 Active Vision for Object Recognition 25

2.4 Recognition from Heterogeneous Data Sources 26

2.5 Applications of Object Recognition by a Mobile Platform 26

2.6 Evaluation Procedures for Object Recognition 27

2.7 Similar Object Datasets 29
2.8 Multiple Viewpoint Recognition and Tracking in Three Dimensions 31
2.9 Chapter Summary 34

3 The Curious George Visual Object Recognition Platform 35
3.1 Introduction 35
3.2 Goals and Design 36
3.3 Hardware 38
3.4 Attention System 39
3.5 Spatial Representation 44
3.6 Planning and Control 46
3.7 Visual Appearance Modeling 49
3.8 Experimental Results 50
3.9 SRVC Description 54
3.10 Chapter Summary 59

4 The UBC Visual Robot Survey Benchmark Dataset 64
4.1 Introduction 64
4.2 UBC Visual Robot Survey Dataset 66
4.3 Evaluation Protocol 75
4.4 Discussion 77
4.5 Chapter Summary 78

5 A Multiple Viewpoint 3D Object Model 79
5.1 Introduction 79
5.2 Model Components 80
5.3 Probabilistic Model 84
5.4 Parameters and Learning 90
5.5 Inference 91
5.6 Chapter Summary 92

6 Viewpoint Planning 94
6.1 Introduction 94
6.2 Sequential Category Recognition Model 97

6.3 Experimental Results 106
6.4 Chapter Summary 109

7 Object Parts and Occlusion Modeling 111
7.1 Introduction 111
7.2 Object Parts and Occlusions for 3D Object Modeling 114
7.3 3D Object Inference 119
7.4 Experimental Setup 120
7.5 Experimental Evaluation 123
7.6 Chapter Summary 126

8 Detailed Object Parts for 3D Scene Understanding 127
8.1 Introduction 127
8.2 A Scene Model with Detailed Object Parts 130
8.3 Scene Prior 132
8.4 Appearance Likelihood 133
8.5 Geometry Likelihood 137
8.6 MCMC-based Scene Inference 138
8.7 Experimental Setup 143
8.8 Experimental Results 148
8.9 Chapter Summary 154

9 Conclusions 156
9.1 Thesis Summary 156
9.2 Open Problems 158
9.3 Outlook and Final Remarks 162

Bibliography 164

List of Tables

Table 1.1 Robotic Recognition Taxonomy 7

Table 3.1 SRVC Result Summary 56
Table 3.2 SRVC 2007 Results 57
Table 3.3 SRVC 2008 Results 58
Table 3.4 SRVC 2009 Results 60

List of Figures

Figure 1.1 Example Recognition Scenarios 2
Figure 1.2 Sample Results 3
Figure 1.3 Illustrated Taxonomy 6

Figure 3.1 Curious George Hardware 37
Figure 3.2 Geometry-based Attention 41
Figure 3.3 Stereo Processing 42
Figure 3.4 Saliency Computation 43
Figure 3.5 Sample Paths 47
Figure 3.6 Spatial-semantic Maps 50
Figure 3.7 Attention Comparison 52
Figure 3.8 SRVC 2007 Images 61
Figure 3.9 SRVC 2008 Images 62
Figure 3.10 SRVC 2009 Images 63

Figure 4.1 UBC VRS Overview 66
Figure 4.2 UBC VRS Details 67
Figure 4.3 A Single Face of the Registration Target 69
Figure 4.4 Sample Registered Scenes 71
Figure 4.5 Sample Annotated Scenes 74
Figure 4.6 The *simulate-with-real-data* Protocol 76

Figure 5.1 Multiple Viewpoint Geometry 82
Figure 5.2 Object-Model Cartoon 85

Figure 6.1 Viewpoint-dependence Example 95

Figure 6.2 Viewpoint-Detection Functions per Class 101
Figure 6.3 Viewpoint-Detection Functions per Detector 102
Figure 6.4 Evolution of Posterior Example 103
Figure 6.5 Viewpoint Planning Results 106
Figure 6.6 Example Use-case for Viewpoint Planning 109

Figure 7.1 Sample Results from Parts-Model 112
Figure 7.2 Part-detector Results and Model Overview 115
Figure 7.3 3D Geometry Example 117
Figure 7.4 Performance Evaluation for Whole Object and Part Detectors 122
Figure 7.5 Qualitative Parts-Model Results 124
Figure 7.6 Result Comparison with State-of-the-art 2D Recognizer 125

Figure 8.1 3D Parts Model Overview 128
Figure 8.2 Geometric Constraint Illustrations 133
Figure 8.3 Sampling-along-ray Illustration 140
Figure 8.4 SSFM Dataset Training Examples 146
Figure 8.5 SSFM Test Examples 147
Figure 8.6 Qualitative Results for 3D Parts Model 150
Figure 8.7 3D Parts Model Results 152
Figure 8.8 Evaluation of Viewpoint Prediction Accuracy 155

Glossary

2D two dimensional.

3D three dimensional.

AP Average Precision.

AR Augmented Reality.

BANK OF DETECTORS a recognizer formed by composition from several sub-recognizers, each of which is typically tuned for a particular viewpoint.

CAD Computer Aided Design.

CATEGORY RECOGNIZER an algorithm that locates groups of objects with the same name, but varying visual appearances within an image.

DPM Deformable Parts Model.

HDL High Definition Lidar.

HOG Histogram of Oriented Gradients.

ICP Iterative Closest Point.

INSTANCE RECOGNIZER an algorithm that recognize objects that are visual indistinguishable from one another within an image.

LOCAL IMAGE FEATURE a representation of visual content based on a (typically small) portion of an image.

MCMC Markov chain Monte Carlo.

MSER Maximally Stable Extremal Region.

OBJECT PERMANENCE the ability to maintain awareness of the existence and approximate location of an object even when it is outside of the current field-of-view.

ORIENTED BOUNDING VOLUME a cuboid represention of space including 3D position, scale and orientation.

PASCAL Pattern Analysis, Statistical Modelling and Computational Learning.

POSE ESTIMATOR a recognizer that predicts object pose along with location.

PR Precision and Recall.

PROTO-OBJECT the representation of a region that has been presented as a candidate object before full recognition reasoning has been completed.

PTU Pan-Tilt Unit.

RANSAC Randomized Sampling and Consensus.

RECOGNIZER an algorithm that locates instances of objects within an image.

RGB-D red, green, blue, depth.

SEMANTIC of or relating to meaning, especially meaning in language.

SIFT Scale Invariant Feature Transform.

SIMULATE FROM REAL DATA the process of re-creating a realistic visual experience using data that was collected in the past.

SLAM Simultaneous Localization and Mapping.

SRVC Semantic Robot Vision Challenge.

SSFM Semantic Structure From Motion.

SVM Support Vector Machine.

TRACKING-BY-DETECTION an algorithm that recovers the trajectory of an object using the results of a recognizer in images collected at numerous instants in time.

TRACKLET a sequence of compatible object detections in neighbouring images.

UBC University of British Columbia.

VOC Visual Object Categories.

VRS Visual Robot Survey.

Acknowledgements

I would like to thank my PhD committee for their support and advice during my studies at UBC. This group, along with numerous students and post-docs, provided a wonderful education and made the place feel like home for so many years. In particular, my supervisor, Prof. Jim Little, has been a great friend and mentor. His positivity, patient support and deep knowledge of the field were invaluable for the completion of this thesis.

My team mates for the Semantic Robot Vision Challenge deserve an extra portion of my gratitude. Our strong performance over the years was due to their outstanding commitment to success, tenacity and smarts. I have benefited greatly from the experience. Special thanks to Bruce Dow, an engineer at UBC, who was instrumental in supporting our work with the Curious George robot.

I would like to thank the hosts of my research visit to the Max-Planck Institute: Prof. Bernt Schiele, Dr. Christian Wojek and Dr. Michael Stark. Their group shared a wealth of expertise in the field of 3D visual understanding and collaborated with me for the final half of this thesis.

Thanks to my parents, family and friends for their support and encouragement. And finally, to Elizabeth for enduring the thesis writing process and the push to finish the final version.

Chapter 1

Introduction

Recognizing objects is a fundamental task for intelligent systems since it maps semantic concepts into physical action space. Reliable recognition is required for many robotic tasks such as responding to simple natural language commands and safely navigating in human-populated environments. Unfortunately, even after decades of study, the performance of methods that recognize objects within visual images is not sufficient to enable many desired tasks, especially in cluttered environments.

This thesis describes visual recognition methods that leverage spatial information sources that are uniquely available during object recognition on-board mobile robots, such as the position of the platform in the world and the range data from its sensors. Our research includes: a physical robotic platform that is capable of state-of-the-art recognition performance; a re-usable data set that facilitates study of the robotic recognition problem by the scientific community; and a three dimensional (3D) object model with accompanying algorithms that demonstrate improved robustness to clutter. Based on our 3D model, we describe algorithms that integrate information across viewpoints, relate objects to auxiliary 3D sensor information, plan paths to next-best-views, explicitly model object occlusions and reason about the sub-parts of objects in 3D.

1.1 Thesis Overview

This thesis describes the development of the Curious George robot: a physical platform that achieved state-of-the-art performance on the Semantic Robot Vision Challenge (SRVC)

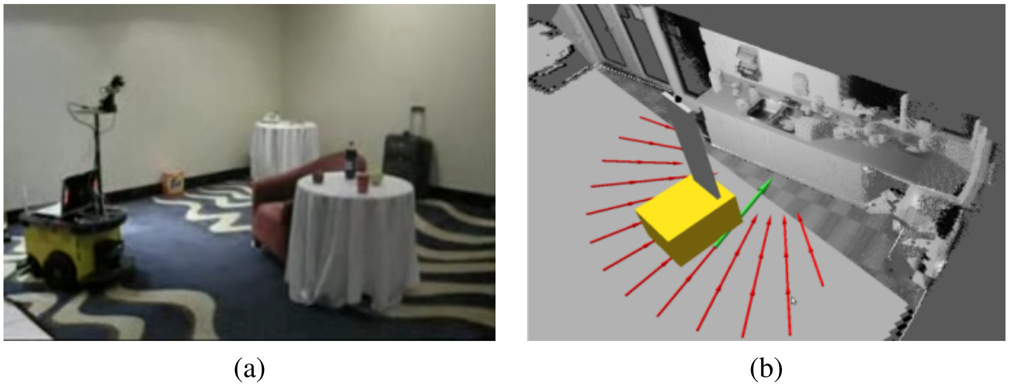
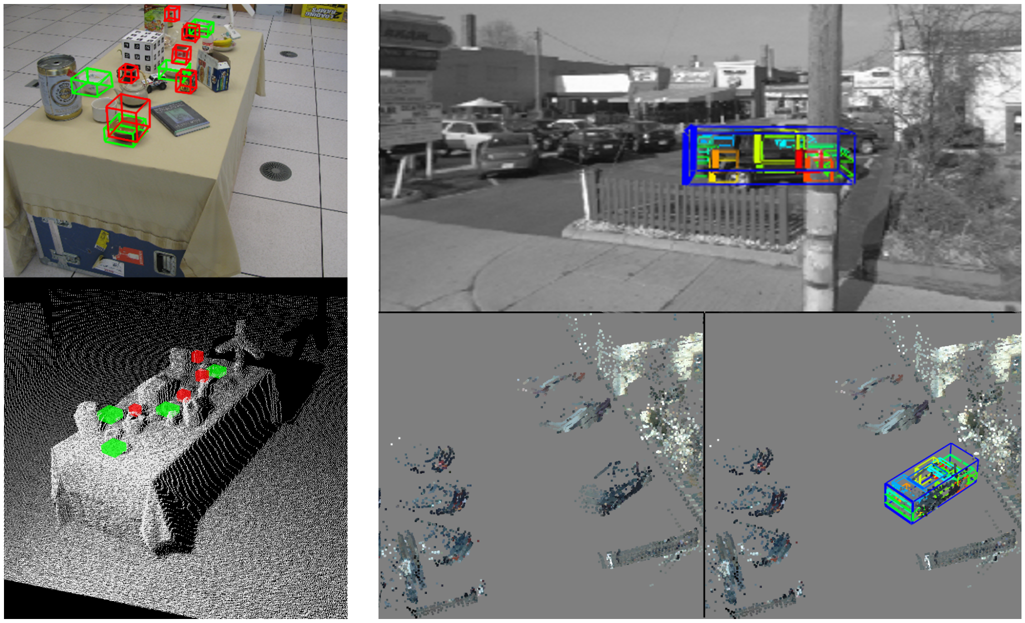

(a) (b)

Figure 1.1: Example Recognition Scenarios: (a) Our Curious George robot performing live object recognition at the 2009 SRVC contest, where it placed first. (b) A sample data collection trajectory that was used to build the UBC VRS dataset overlaid on the collected 3D laser point cloud.

contest, an international robotic recognition competition (see Figure 1.1(a)). During the contest, Curious George was required to autonomously explore a single room, to collect useful images, to recognize objects within these images, and to report results in an on-line fashion. By placing first in the SRVC contest in several years, sometimes out-scoring the competing teams by a large margin, we demonstrated the effectiveness of our recognition approach. In particular, we developed a targeted visual attention system to guide our robot and cameras. This enabled rapid success on visual tasks compared to competing approaches such as random motion.

In order to continue our empirical study of recognition methods, we collected a dataset that allows repeated replication of the sensory experience of a robot performing visual recognition. Curious George collected visual survey trajectories, which covered a dense set of the possible viewing directions of a scene. The information from the robot's cameras, spatial mapping system and laser range finder were archived for each scene (see Figure 1.1(b)). We developed a *simulate from real data* protocol that enables the repeated testing of recognizers by drawing specific subsets of the recorded data. We annotated the ground truth object information, to allow quantitative analysis of results. The resulting data, called the UBC Visual Robot Survey (UBC VRS) was made available to the scientific community. This dataset represents a more genuinely robotic visual experience than can be easily replicated with existing data sources, which are often collected using hand-held sensors with



(a)

(b)

Figure 1.2: Sample Results: (a) A kitchen scene where our method's 3D object recognition results, displayed over a 2D image (top), are a close approximation to the 3D ground-truth objects, displayed over a point cloud (bottom) (mugs shown in red and bowls shown in green). (b) An urban driving scenario where our method recognizes a partially occluded car and its composite parts in 3D, projected into one image (top). The same scene is shown over a 3D point cloud (bottom) both with the result overlaid (right) and without (left).

weak spatial registration between viewpoints. As such, it provides a new opportunity for other researchers to more easily study the robot recognition problem.

Our algorithmic contributions are built upon a probabilistic 3D object model that relates an object's position and shape to images collected along a robot's trajectory. An object, represented in 3D, is explained by the visual appearance of its corresponding region in every image, which we determine using registration of sensor positions (e.g., from structure-from-motion or laser range finder based localization). Since images are inherently 2D, the process of locating objects in 3D requires reconstructing missing depth information, similar to stereo-vision. We have implemented several statistical inference procedures that reliably perform this 3D object localization.

We have demonstrated the effectiveness of our model and associated algorithms for a number of applications. First, viewpoint-aware visual appearance models allow next-best-view planning that leads the platform to informative views. We utilize sensed range data to explicitly capture occlusion for each viewpoint, again leveraging the recovered 3D object locations. This occlusion model has demonstrated state-of-the-art performance on recognizing kitchen objects in clutter (e.g., Figure 1.2(a)). Finally, we model the spatial information of an object's sub-parts to produce more detailed output and to enable yet stronger parts-based occlusion reasoning. This has been demonstrated by our system's strong performance on recognizing occluded automobiles on a standard dataset (e.g., Figure 1.2(b)).

In combination, the techniques within this thesis form a mobile visual object recognition system that achieves state-of-the-art performance for a number of evaluation tasks. The potential for fusing information from many sources within a probabilistic model gives the promise for ever-stronger performance as robots are equipped with new sensor types and their mapping approaches become more accurate. This direction is likely to be a part of semantic awareness systems that are able to perform tasks in the real world.

1.2 Problem

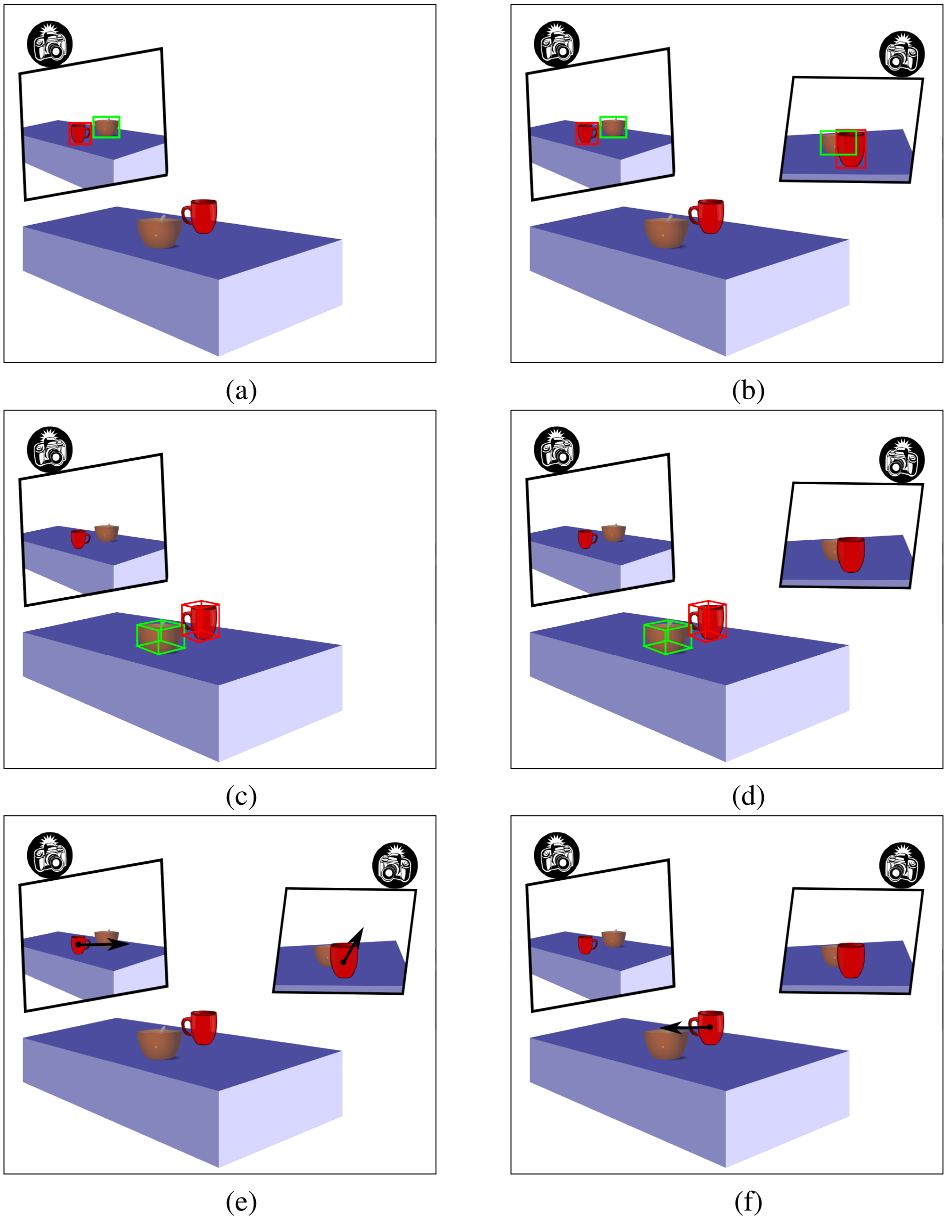
We define the visual object recognition problem for mobile platforms as recovering an object's name in the form of a semantic category label, where *semantic* is defined as "of or relating to meaning, especially meaning in language" [dic00], using images from an automated camera. A method that solves this problem, which we call a *recognizer*, must locate objects by recovering their center and size either within a visual image or, with additional challenge, in a 3D coordinate system. An algorithm that predicts object pose along with location will be referred to as a *pose estimator*.

Note that our definition allows recognition of both *generic object categories* (e.g., everything a human would call *bottle*) and *specific object instances* (e.g., the bottle of a single type and size of soft-drink). The name chosen as a recognition target implies a level of specificity based on a human's mapping of that term to a set of instances. Throughout this thesis, we consider recognizing *generic object categories* and methods that generalize across the appearance of different instances, except where we compete in tasks defined by the research community, such as the SRVC contest, which has a specific instance recognition component.

- *Input:*
  - Visual images from a camera that moves between frames.
  - Registration information, either accurate or approximate, that relates the positions of the camera over its trajectory to a global frame.
  - (optional) Sensed depth data corresponding to the images (e.g., from laser scanner or stereo camera).
- *Output:* List of hypothesized objects, each composed of:
  - A semantically meaningful category label.
  - A geometric location in 2D or 3D.
  - A geometric scale in 2D or 3D.
  - (optional) Orientation information, sometimes referred to as pose, in 2D or 3D.

Beyond this simple description, several additional details are needed to specify the exact nature of the recognition problem to be solved. Table 1.1 lists numerous problem dimensions. Each selection of one item for each dimension represents a potential specific problem that can be studied. For every particular application and intelligent system, the system designer can choose the nature of problem that most naturally fits the requirements.

The work in this thesis covers a sub-set of specific problems that are most relevant to the needs of mobile intelligent platforms. For some problem dimensions, this means that we have considered only a single option. For others, we have achieved a more complete coverage of the potential options. We will begin by discussing the problem dimensions that are depicted in Figure 1.3, as these are the most significant for our work: the number of viewpoints available, whether objects must be localized in image space or in three dimensions, and whether the object's pose should be predicted. The following sections will then continue to describe all additional problem dimensions listed in Table 1.1.



(a)

(b)

(c)

(d)

(e)

(f)

Figure 1.3: Illustrated Taxonomy: The first row represents (a) single-viewpoint and (b) multiple-viewpoint *image space object localization*. The second row shows (c) single-viewpoint and (d) multiple-viewpoint *three dimensional object localization*. In the third row, *object pose estimation* is depicted as an independent per-viewpoint task in (e) and as a cross-view task relative to the world frame object in (f).

Problem Dimension	**Examined Instances**
Number of Viewpoints	single, **two (wide-baseline stereo)**, **greater than two**
Number of Localization Dim.	image space, **three dimensional**
Pose Estimation	none, per-image, **global frame**
Registration Information	inferred, **known**
Sensor modalities	stereo range, RGB-D, **vision**, **laser range**
Control	**passive**, **active**
Objects vs Scenes	**independent object**, **joint scene model**
Real-time requirements	hard real-time, **near real-time**, **off-line**

Table 1.1: Robotic Recognition Taxonomy: several dimensions upon which recognition approaches can be compared. The problem instances that are most directly considered within this thesis are indicated by the bold text descriptions. Several of the plain text items are discussed in minor detail.

Number of Viewpoints

As is depicted by Figure 1.3(a) and (b), we distinguish between *single viewpoint recognition*, where only one image of a scene is available, and *multiple viewpoint recognition*, where several views can be processed jointly (e.g., after they have been collected by a moving platform). The ability to reason about scenes through the motion of a platform is a primary concern of this thesis. Some of our results do represent algorithms that treat views independently or that are only able to handle a single image of a scene. However, the large majority of our technical discussion and novel contributions deal with multiple viewpoint solutions to object understanding.

Number of Dimensions for Localization

The first and second rows of Figure 1.3 contrast object locations provided in image space with those that exist in three dimensions. 3D localization allows for our visual recognition outputs to be used by robotic systems to perform tasks such as grasping. While we produce 2D localization systems for comparison purposes and to standardize our results with the scientific community, a primary goal of this thesis is to demonstrate capable 3D localization from multiple images.

To appreciate the fundamental difference between the two tasks, the reader is asked to consider the task of translating the 2D detections shown in the right-most simulated image of Figure 1.3(b) into the 3D object descriptions of Figure 1.3(d). Since the bowl and mug are overlapping in the image, the image space results of many 2D object recognizers will produce overlapping bounding boxes. Even if sensed depth values (e.g., by an RGB-D camera) are available, they will not give a consistent measurement of an object's missing depth dimension, since a range of distracting depth values will be present within each bounding box. Correctly recovering 3D information requires inference of the type described later in this thesis.

Object Pose Estimation

The final row of Figure 1.3 depicts two of the possible formats for recovering the pose information. First, a method can predict a viewing direction local to each single image (e.g, "side-view in the first image and back-side view in the second"), as in Figure 1.3(e). While this information can be translated after-the-fact into a global frame common to all images using registration information, the numerous estimates from each image may not agree precisely and the process may need to be somewhat sophisticated. Therefore, the second approach, directly predicting a pose in the global frame as shown in Figure 1.3(f), is primarily considered in this thesis.

Registration Information

Our problem formulation considers information about where the camera has moved as an input to our algorithms, rather than hidden information that must be recovered by the method. The existing literature on this problem is somewhat split on whether registration information should be assumed known while inferring semantic objects. However, the majority of modern robots already employ highly capable localization systems that are typically based on matching sensor readings to existing maps comprised of linear segments or point features. There has been little evidence in the research literature that the addition of semantic object information is able to improve localization performance. Therefore, in this thesis, we will separate localization from other tasks, assuming it is solved by an external tool.

Sensing Modality

In this thesis we assume the inputs to a recognition system may be visual images only, or that sensed range information such as from a laser range finder might be available. We will not consider systems where *only* range information is available. Rather, visual appearance will be the primary signal that will be used by our systems to relate sensory information to objects in the world. Although 3D data has become much more readily available during the course of this thesis, for example using Microsoft's Kinect sensor or the Velodyne High Definition Lidar (HDL), cameras are still the most ubiquitous sensors to be available on intelligent systems. They operate in a larger range of environments (e.g., underwater and in bright outdoor sun) and are inherently passive as opposed to systems that radiate energy into the world. Consequently, throughout this thesis, we base our description of semantic objects on visual appearance in images. Many of our techniques could be adapted to utilize range-based or hybrid object appearance models. This is a promising direction for future work.

Control

Many intelligent systems provide an opportunity for automated control. Examples include intelligent mobile robots and security cameras with pan-tilt mounts. In other domains, an intelligent system can passively perceive but not guide its motion. These include passive driving assistants, smart phones and fixed surveillance systems. Our probabilistic model is suitable both for situations where a stream of data is obtained after a camera is externally controlled and also for systems that simultaneously perceive and give feedback to control the sensor. Therefore, this thesis achieves a fairly broad coverage along this problem axis.

Objects vs Scenes

Numerous authors have considered an independence assumption between the objects that appear together in a scene, which allows for simple and efficient inference of each object independent of all others. This independence assumption is often not a faithful representation of reality, since objects in real scenes overlap one another, influencing the local visual appearance and affecting any features or models that are extracted from the images. We have considered both sides of this problem dimension by treating objects as independent for the majority of the thesis and then relaxing this assumption by considering a more sophisticated inference approach in Chapter 8.

Real-time Requirements

We have implemented a near real-time recognition system in the Curious George robot that is described in Chapter 3. However, the remainder of the thesis has not been evaluated on a physical robot platform, and therefore we have not focused on real-time execution. Many of our approaches depend on tools developed by other authors to scan single images and score likely object locations based on visual appearance models. In particular, the Deformable Parts Model (DPM) of [FGMR10] is used in Chapters 6, 7 and 8. This method can take up to a full second per input image, at typical resolutions. We treat this mainly as an external component separate from the methods in this thesis, and so if a faster image space recognizer were used, our approaches would have more potential to be near real-time. In particular, the viewpoint planner of Chapter 6 executes much faster than real-time if the image detection step is ignored, although we have run this off-line. The simple parts-based multiple viewpoint 3D object inference method of Chapter 7 is close to real-time, but its run time scales with the number of objects present in a scene. It becomes slower than real-time with on the order of tens of objects present. The scene inference technique presented in Chapter 8 is currently much slower than real-time. That approach involves four distinct image recognition methods in each image, as well as the 3D inference approach, which requires several additional seconds to process each pair of images. In general, for all thesis components except the Curious George robot, we have taken the approach that there are numerous potential speed optimizations that could be made, but we have left these for future work when the approaches are implemented on a real robot system.

1.3 Contributions and Outline

The work contained in this thesis has been published in an international journal and several conferences. We group the research contributions contained into several distinct categories and describe the refereed publications that have resulted from each:

- *Visual robot search platform:* The Curious George robot platform represents a contribution both as an autonomous recognition platform that integrates existing components in a successful way and also by contributing several novel components such as the embodied attention system. The platform development efforts that are a part of this thesis were published in an early form at a workshop [MFL$^+$07] and the completed system was described at the International Conference on Robotics and Automation (ICRA) 2008 [FML$^+$08] and in a journal [MFL$^+$08].

- *Robotic recognition dataset:* The UBC VRS dataset is a large corpus of data collected by a real robot moving through many environments. The objects in these environments were annotated carefully. Simulation and evaluation software allows for repeatable testing of the performance of recognition approaches. This work was published in 2012 at the International Symposium for Experimental Robotics (ISER) as [ML12].

- *Active Vision for Category Recognition:* This thesis contributes a control strategy that guides the motion of a robot's base through an environment using learned viewpoint-detection models of the target object categories so that informative viewpoints are obtained. This work described at ICRA in 2010 as [MGL10].

- *Occlusion reasoning multi-view 3D category recognition:* Occlusion is one of the most common failure modes for current object recognizers when images of objects have a realistic level of clutter. We produced a method to improve the performance of a 3D multiple-viewpoint recognition by explicitly modeling occlusion. This work appeared at the International Conference on Robotics and Intelligent Systems (IROS), 2011 [ML11].

- *Parts-based multi-view 3D object reasoning*: We have considered part decompositions based on pre-determined image space geometry and on semantically meaningful decompositions of objects into their 3D component parts. These methods lead to a significant improvement in the accuracy of our multi-view 3D object detection method. The first portion of this work was presented at the British Machine Vision Conference (BMVC) in 2011 [MWSL11]. The most recent version of this work appears in Chapter 8 and has not yet been published at the writing of this thesis. We plan to submit this content to an international conference or journal in the future.

In addition, the work described in Chapter 3 has been evaluated during an international robotics competition named the Semantic Robot Vision Challenge (SRVC) [SRV]. In three years of participation in this contest, the Curious George platform twice achieved first place standing in the robot division of the contest. In the third year, the associated object recognition system placed first in the software-only division.

1.3.1 Thesis Outline

This thesis can be viewed as four parts:

- (Part 1) includes the introductory material, in this chapter, and a discussion of related work in Chapter 2.
- (Part 2) describes the systems and data contributions including the Curious George robot, in Chapter 3, and the UBC VRS data set, in Chapter 4.
- (Part 3) contains the primary technical contributions of the thesis including a high-level model description, in Chapter 5, application of the model to active viewpoint planning, in Chapter 6, occlusion and parts-aware object inference for indoor objects, in Chapter 7, and scene-level reasoning with detailed object parts, in Chapter 8.
- (Part 4) concludes the thesis with a summary and discussion of open research problems, in Chapter 9.

The remainder of this thesis will be quite accessible if read in a linear fashion and the material in each chapter builds upon the previous in non-trivial ways. However, readers seeking to understand only one of the novel technical contributions are encouraged to begin reading at Chapter 5, which describes the generic form of our model, and then to continue to choose the one of the three subsequent chapters. Material specifically focusing on the active nature of the robotic recognition problem is primarily found in Chapter 3 and Chapter 6, where two planners for the robot and its camera can be found. Some readers may come to this thesis seeking our contributions to home robotics, which could be found by reading about the Curious George robot in Chapter 3, our dataset of kitchen images in Chapter 4 and the results of our technique on that data in Chapter 7. Results for automobile recognition in urban driving can be found in Chapter 8.

1.4 Chapter Summary

This chapter has provided an overview of the robotic object recognition task and has given a preview of the work that will be described in the remainder of this thesis. We described motivations and background, provided a taxonomy to sub-divide the various related subproblems. The thesis will continue with a detailed discussion of related work.

Chapter 2

Related Work

The research in this thesis has been conducted during a time of rapid exploration in both the computer vision and robotics communities. Large strides were made in the performance of visual appearance models for the single-image category recognition task. Fast and reliable range sensors (e.g., the Microsoft Kinect) became available and these inspired numerous methods that investigate the use of 3D features, models and abstractions. A number of new standards were established by the research community to measure progress on recognition tasks, such as the Pattern Analysis, Statistical Modelling and Computational Learning (PASCAL) Visual Object Classes (VOC) [EVW$^+$12], ImageNet [DDS$^+$09] and multi-view Kinect [LBRF11] datasets.

In this chapter, we describe the research developed by other authors that had the largest impact on the methods we describe within this thesis. Broadly, we will organize the discussion into a summary of visual object recognition history and the current state of the art for a number of recognition sub-problems that are closest to the work in this thesis: classifiers that operate strictly in the space of visual appearance, pose estimators that add the ability to predict or model the object's orientation, recognizers that are aware of the parts-layout of an object, and recognizers that explicitly handle some level of visual occlusion. The problem of viewpoint control will be briefly described. We then describe existing experimental benchmarks along with current results as well as a number of physical robot systems that are capable of embodied visual recognition. Finally, we will discuss visual tracking and multiple-viewpoint recognition techniques that fuse information across numerous images, as these are the most similar to our own approaches.

2.1 Single Image Object Recognition

The study of object recognition in visual images has a long history, with notable early work by Helmholtz [Von67], Hubel *et al.* [HW79] and Gibson [Gib79] contributing to the understanding of human vision, among countless others. Automated object recognition using digital images is a classification process, where the correct label must be assigned to a portion of the image based on analysis of the image content. A major challenge for visual recognition techniques has always been the plethora of viewing conditions and camera properties that affect each image. A single static scene can produce a wide variety of different images as the lighting and camera properties change. This means that simply describing objects based on the values of raw image pixels is inadequate.

Recently, significant progress has been made through the study of invariant features, which can be extracted from images in order to produce new representations of the visual content that remain consistent with respect to some number of these disturbances. Gradients, which are formed by taking differences of nearby pixels, provide greater illumination invariance than the raw intensity values. The scale-space theory of Lindeberg [Lin90] has been used by a number of authors to produce features that give a nearly consistent response as the image resolution changes (e.g., Witkin [Wit84]). The scale-space of an image is constructed by repeatedly blurring an original image (i.e., applying convolution with a Gaussian filter), and down-sampling to produce a pyramid of images at successively lower resolution.

Distance in the space of invariant features is typically a better indicator of similar content than distance in the space of raw intensities, but this still does not produce capable recognizers. Note that the appearance of an object will change as it rotates in the image, as parts of the object are deformed, and as the background that surrounds the object changes. There are several approaches to overcome these changes in order to produce a successful recognition technique. Roughly, these can be grouped into methods that extract informative sub-regions of the image in a bottom-up fashion to focus further processing and those that scan a template for the entire object across the image in some fashion (e.g., exhaustively as a sliding window search or in some adaptive fashion). We will briefly describe the first approach, which is known as *local feature* image description, and will then continue by describing approaches for scanning regions within the image.

2.1.1 Recognition with Local Features

The first method that we will describe is based on the observation that even as many aspects of an image change with viewpoint, lighting and object motion, a sub-set of meaningful local appearance regions are likely to remain largely unchanged. Numerous methods describe an image based on a collection of local features including [MTS$^+$05, KB01, BMP00, BTV]. The Scale Invariant Feature Transform (SIFT) of Lowe [Low04] is perhaps the most common scale-invariant local feature and an implementation made available by the author[1] is widely used. SIFT features describe small patches of image content in a fashion that is invariant to scale, lighting and in-plane rotation, and tolerates a moderate amount of affine deformation. Feature points are located at the extrema of the difference of Gaussians over neighboring levels in scale-space, where extrema-finding can be performed with precise interpolation to produce sub-pixel and sub-scale accurate feature locations. Several heuristics prune points that are less informative and the local dominant orientation is computed at each remaining point. Finally, a feature descriptor is constructed, respecting the orientation that was estimated, by rotating local gradients and binning the resulting vectors into eight discrete orientations. The patch is spatially decomposed into a four by four grid and a separate orientation histogram is produced in each grid cell, which produces a 128-dimensional description of the local image region.

Authors have also studied stronger invariance, for example the work of Mikolajczyk *et al.*, which achieves invariance to affine image transformations [MS04]. The basic principle of this approach is to estimate the local surface orientation so that the feature can be appropriately transformed. This makes matching possible over a wider range of deformations. Numerous other authors have considered techniques for enhancing the viewpoint invariance that can be achieved by local features (e.g., [MY09, WCL$^+$08]).

Regardless of the type of feature, the process for recognition using local features is somewhat similar. It involves searching for large sets of matching feature pairs that are close in descriptor space so that the set respects geometric constraints. For simplicity, consider now that only a single training image of an object is available, and we seek to recognize (locate and label) another instance of that object in our run-time processing. Local features are extracted at the keypoints located in each of the images, and feature descriptors are computed. For the remaining discussion we will assume that *N* features are found in the

[1] http://www.cs.ubc.ca/ lowe/keypoints/

training image. The next step is search for the best match to every feature in the test image, where we assume two features that are close in descriptor-space are candidate matches that may explain the same image content. Brute-force search for the closest match can be performed naively with complexity $O(N)$, by computing the distance to every feature in the training image. Several more efficient techniques are available to find neighbours in low dimensions, such as the k-d tree [FBR77], with complexity that scales as $O(log(N))$ with the number of features. However, the so-called "curse of dimensionality" [MS79] describes that performance of any such exact method is dominated by the dimensionality of the underlying space. In high dimensions, such as the 128 of SIFT, the naive approach is more efficient. Approximate matching techniques such as [BL97, ML09] are often employed in practice and allow for efficient matching at the cost of rarely locating a sub-optimal neighboring point.

Any set of candidate feature matches that are found solely based on appearance similarity is likely to contain some matches that truly correspond to the same positions on objects and other matches that are accidental (e.g., due to image texture and repeating patterns). Typical approaches model the spatial layout of visual features in the training image and search for a sub-set of the matches that agrees upon a consistent geometric model of some form that explains a feasible motion of the object between images. One example spatial model is the space of all affine transformations of the assumed-rigid object. A robust model-fitting method such as Randomized Sampling and Consensus (RANSAC) [FB81] can be used to locate sets of features that agree upon the same geometric model. More flexible spatial models have also been considered, for example by [FPZ03] and [LLS08].

The output of a local feature-based recognizer is a set of object hypotheses. This set may be empty for an image, if zero or few matching feature points are mutually consistent. For every large group of compatible features which are inliers for a particular object placement within an image, the common geometric transformation can be used to estimate the image location of the object in the query image, which forms an object hypothesis.

2.1.2 Recognition with Appearance Templates

The second major approach for object recognition in images is based upon the idea that each position in a test image can be compared to a full-object template, which is typically learned from numerous training examples. Methods based on learned appearance templates are particularly appropriate for recognition of object categories, where numerous training images might be needed to capture the variation related to single semantic label. This variation can be caused by the presence of sub-categories (e.g., cars, trucks, vans and convertibles are all automobiles) or simply because instances with the same name show variation (e.g., each human face has a unique appearance). Numerous authors have proposed constructing appearance templates using similar gradient-based image processing techniques to those employed for the local features described above (e.g., [SC00, DT05, BMP00, BTV]). Many of these approaches are suitable to describe the appearance within each image window in a fashion that also has a number of useful invariance properties.

Each of the considered windows, once described by a feature representation, can be classified or scored by an appearance model that is learned from training data. Classification may be through a straightforward application of a generic technique from machine learning such as a Support Vector Machine (SVM) as in Dalal *et al.* [DT05] or boosting as in Torralba *et al.* [TMF04]. Alternatively, object parts (e.g., [FH05, LLS08, FPZ03]), viewpoints (e.g., [TFL^{+}09, SSFFS09]) or occlusion reasoning (e.g., [GFM11]) can be incorporated within this classification step. Here we will specifically focus on two recognition approaches that we have used extensively as inputs to the methods within this thesis: Felzenszwalb *et al.* [FGMR10] and Stark *et al.* [SGS10].

The Deformable Parts Model (DPM) of [FGMR10]

During the course of this thesis, the method that has displayed the most consistently strong performance on the PASCAL VOC contest has been the Deformable Parts Model (DPM) of Felzenszwalb *et al.* [FGMR10]. DPM has gained wide use by the computer vision community and the source code, which is publicly available[2], has been used in numerous places throughout this thesis. This method is based on the Histogram of Oriented Gradients (HOG) feature that was developed by [DT05] for detecting pedestrians. HOG features are similar to the SIFT descriptor, in that they compute discretized gradient information pooled over spatial bins. Rather than being computed only at feature points, they are typically computed over a dense grid that covers the entire image. Also, local normalization over nearby groups of histogram bins provides improved repeatability. HOG features are not, by default, rotationally invariant as they do not benefit from the local orientation estimate

[2] http://www.cs.berkeley.edu/ rbg/latent/

that we described above for local feature methods. This makes the unmodified HOG feature suitable for detecting objects that always appear with the same viewpoint and in-plane orientation.

The DPM approach extends on basic HOG templates in several ways. Perhaps most important is the addition of numerous part templates that can deform relative to each entire object (i.e., a part can take a variety of positions within the object's outline). These parts, in addition to the so-called root filter, are each composed of a HOG-like descriptors, modified to have lower dimensionality and made invariant to reflection. The part filters are applied at twice the resolution of the entire-object filter, to capture the finer detail that makes up an object's components. Numerous sets of entire and part templates are used in a mixture model with the intuition that each mixture is able to model a different mode or viewpoint of the object's appearance.

A primary contribution of the DPM approach is the use of a Latent Variable Support Vector Machine (LVSVM) representation to form the training objective function. The SVM approach seeks to maximize the margin, which in this case is the number of misclassified training examples. The latent variables refer to assignments of training examples to mixture components and the placement of parts on each positive example. The LVSVM is trained using stochastic gradient descent. To make best use of the extensive set of negative data that is available for most recognition tasks (it is easy to find images that do not contain the object), and particularly the PASCAL VOC challenge, only an initial sub-sample of the negative data is initially provided to the method. In subsequent training rounds, data-mining of hard negatives is performed. Hard negatives are the windows of negative data that obtain the most positive scores, indicating regions where the current model is most likely to make an error. By focusing the training process on the most error-prone background regions, convergence occurs much faster than it would if all negative windows were used in each round.

After training completes, the learned appearance model can be used for detecting objects in a test set, or on-board a physical robot as we discuss in this thesis. Sliding the object template exhaustively across the entire image is an expensive process, especially when an optimal part configuration must be associated with each detection hypothesis. In practice, an efficient detection method based on distance transforms was proposed in [FGMR10]. [FGM10] proposed a so-called cascade detector which rapidly rejects large areas of the image with initial filtering before running detailed analysis only in a small number of promising locations. In any case, the detection outputs are a set of object hypotheses in the form of bounding boxes in image space, with associated normalized margin scores that determine the confidence of the classifier in each result.

The Part-Constellations of Stark *et al.* [SGS10]

A second approach that we have utilized within this thesis is the viewpoint-aware constellation of semantically meaningful parts that were described by Stark *et al.* [SGS10]. While being somewhat similar to DPM in overall motivation (i.e., both approaches attempt to model objects as collections of parts related by loose spatial relations), the training process and model specifics have significant differences. Stark attempts to perform detailed viewpoint estimation jointly with detection, and in each viewpoint attempts to accurately localize a set of semantically meaningful object parts that remain consistent across views. Note that this contrasts with the DPM object parts which are selected simply to optimize a discriminative classification training objective and may not correspond to the part decomposition that a human would perform.

In order to train such detailed spatial information, the method begins with 3D Computer Aided Design (CAD) models of many instances of an object category. In order to obtain features that can be later related to visual images, these CAD models are rendered from a 36 viewpoints (every ten degrees) for the paper's results, but this is an adjustable parameter of the method. Object part labels that are present in the CAD models are transfered into each image, along with viewpoint information. Shape context features developed by Belongie *et al.* [BMP00] are extracted densely over the rendered image patches. Independent part appearance models are training using Ada-Boost [FS97], which forms a strong classifier by combining a number of weaker classifiers. Platt Scaling [NMC05] is applied to the raw output of the boosted classifier in order to produce well-scaled detection scores. A so-called constellation model, which is a specific form of probabilistic graphical model encoding variation in part placement with respect to the object's centre, is learned for each viewpoint. While all positive training examples are derived from rendered CAD imagery, in order to produce an object detector that is robust to real-world image effects, a large number of real visual images are used to form a negative set. The result of the training process are the weights and local boosted classifiers that form a viewpoint and part-aware appearance model.

At testing time, each image region would ideally be compared to every learned viewpoint and the optimal part layout would be found. This is prohibitively expensive for the number of viewpoints and parts in the model, so instead a sampling-based search process using the Metropolis-Hastings (MH) [GSS96] algorithm is used to approximate the maximum *a posteriori* (MAP) object configurations in an image. Local non-maxima suppression is needed, since MH results in many likely samples associated with every true object with a slightly shifted image location or viewpoint. The final result of the Stark object detector is a set of object hypotheses. Each has a bounding box in image space, a viewpoint estimate precise to the number of views rendered during training, an estimated part-layout that describes where each of the human-understandable parts is likely to fall within the entire object's region and a confidence score. Chapter 8 demonstrates the utility of the detailed viewpoint and object-part information available from the Stark detector for the purposes of multiple viewpoint object reasoning.

2.1.3 Recognition of Objects Joint with their Spatial Properties

We attempt to recover 3D information about the objects in a scene so that our hypotheses are useful for a mobile robot performing tasks based on the objects. Numerous authors have previously considered recovering spatial information from single images of a scene, such as: object pose, occlusion, scene layout, depth maps (e.g., [SCN08]), segmentation (e.g,. [AMFM11, LSD12]), and material or contextual properties of regions (e.g., [PG11, FEHF09]). The remainder of this section will discuss several examples from each of the first three areas listed above, since these are most relevant to our work.

Pose Estimation

We consider estimation of the pose of an object, along with its category label, location and size in Chapters 6 and 8. This task is related to approaches for building object category recognizers that perform well over all viewpoints [SSFFS09, TFL$^+$09, LSS08, SK00], a challenge since many objects exhibit large variation in appearance across their aspects. Multi-view recognition methods typically require training data in which the object's pose has been annotated in each image or a video sequence that captures the object from many directions as in [SSFFS09]. The most standard approach for recognition from multiple viewpoints is to discretize the space of viewing directions (e.g., by defining 8 views spaced at 45 degree increments) and to model each direction as either independent or related to its neighbors. If an independent recognizer is trained for each viewpoint, it is common to call the approach a *bank-of-detectors*. Final answers from this bank are obtained by evaluating the detector for each viewpoint on each image and performing non-maxima suppression to pick the viewpoint model with the highest local response. Several authors have shown that introducing links between viewpoint models, such as enforcing shared features during training as [TMF04, OPZ06] or some minimal appearance continuity as [TFL$^+$09], allows for improved multi-view recognition performance over the simpler independent detector bank.

The above methods differ from our work in that their motivation for exploring viewpoint is primarily to improve the performance of the recognition method. They have typically not evaluated the accuracy of their techniques for estimating pose, and do not explicitly model the variation or distribution of appearance with respect to pose. Several authors have also explored the variation of object category models with respect to viewing direction (e.g., [LA06]). These approaches build viewpoint-aware models that predict an object's pose with relative accuracy and can be used within a planning framework. Our viewpoint selection method in Chapter 6 uses a similar approach, and we will discuss related approaches further in the context of viewpoint planning in Section 2.3.

Occlusion

Occlusion is perhaps the most dramatic effect that can alter the contents of an object's image patch, since the occluded portion is independent of the object and its properties. This effect can alter the score obtained by a recognizer on an occluded object instance, as the features will typically appear more similar to the background than for unoccluded instances. One solution is for a recognizer to be inherently robust to occlusion, so that even heavily occluded instances are distinguishable from background, but this is unfortunately not the case for many current methods.

Numerous authors have considered explicitly reasoning about occlusion while recognizing objects in single images. Vedaldi *et al.* [VZ09] employed a structural SVM based on HOG-like features to predict object occlusion or truncation jointly with the usual dimensions of object location and category label. Wang *et al.* [WHY09] study a similar problem but show that a second complementary image region descriptor is useful for the task. Schindler *et al.* [SD10] have considered occlusion-aware appearance models in the context of reasoning about large-scale reconstructions of entire city scenes. In each of these cases, if the correct occlusion for a particular image patch can be determined, the techniques are able to preserve a meaningful score that assigns confidence in the detected region normalized by the amount of visual area.

Occlusion has been studied in the context of other computer vision problems, such as determining optical flow [JFB02] and segmentation [YHRF10]. These approaches show promising bottom-up cues that can be used as an indication of the presence and location of occlusion. Fusion of such information with appearance-occlusion models for objects and 3D inference is an interesting direction for future work.

In a paper that appeared after all of the technical contributions in this thesis were completed, Hsiao *et al.* [HH12] discuss occlusion reasoning performed from a variety of viewpoints for specific instance recognition. Their method includes detailed geometric reasoning, priors about likely types and frequencies of occlusions and processing of each object's silhouette. Their method shows strong performance on occluded object instances, demonstrated on highly cluttered kitchen data, which is similar in many respects to the UBC Visual Robot Survey dataset that we will describe in Chapter 4. Our methods are not directly comparable, because we study object category recognition and [HH12] considers specific instances. However, several aspects of their technique, especially the occlusion boundary and silhouette reasoning, are suitable to be incorporated into future versions of our method.

The most similar previous approaches to ours are those which combine occlusion reasoning with tracking in video, as they both fuse information over space and infer a depth layout of the scene. Pedestrian tracking methods, such as those developed by [EESG10] and Wojek *et al.* [WWRS11], have inspired the mixture-of-experts formulation to combine partial object detectors that we will describe in detail in Chapter 7. The primary difference is that we utilize sensed depth information, such as from an RGB-depth camera, while [EESG10] uses motion discontinuities to pre-segment regions and [WWRS11] utilizes inter-object reasoning that requires all occluders to be detected with an appearance model (i.e., if the background creates occlusion, this is not modeled).

Scene and Object Layout

Our work places 3D objects within a scene, which allows for reasoning about geometric constraints between pairs of objects as well as between objects and elements of the environment such as walls and support surfaces. Numerous authors have attempted to infer the 3D layout of a scene either based on generic geometric cues derived from the image (e.g., [TXLK11]), or by joint reasoning with the recognition process (e.g., [WGK10, BSS11, FDU12]). Hoiem *et al.* [HEH06] recover a rough overall scene geometry by estimating the orientation of surfaces in the image and by placing priors over the likely orientation of the camera. This allows filtering and re-scoring of object detections made in image space, which significantly improves recognition performance.

Fidler *et al.* [FDU12] is the approach that uses the most similar layout reasoning to our methods. They have demonstrated that the DPM approach can be extended with 3D geometric information. Specifically, the object's overall shape in 3D is well approximated by a cuboid. Their approach estimates the 3D position, orientation and scale of objects in the scene from single images, which they refer to as cuboids and our later chapters refer to as *oriented bounding volumes*. As such, both the inter-object mutual exclusion and also expected ground plane constraints that we apply later can be formulated identically in both approaches. Fidler's work has been published after all technical contributions in this thesis were complete and the authors have not evaluated their approach on any of the datasets that we have used for results in this thesis. Direct comparison, or fusing of the methods appears to be a promising future direction.

2.2 Previous Robot Recognition Systems

An important aspect of this thesis is the assembly and programming of a mobile robot system named Curious George that demonstrates the capability to recognize objects in real-world scenarios. Numerous other authors have also developed embodied object recognition systems. Those approaches often consider elements similar to our work, in particular when the systems are aimed towards home robotics (e.g., [SBV07, MFS$^+$07]) or urban driving (e.g., [SBFC03]). This section will describe a sub-set of the platforms that are most related to our methods.

Ye *et al.* [YT99] describes *Playbot*, a robot which models the variation in viewpoint when observing a specific object, learns this model from training data, and uses the learned model to guide its behaviour. We have been inspired by this approach, and have performed a similar analysis for the response of object category detectors over many instances of each category in Chapter 6.

Sjö *et al.* [SLP$^+$08] have constructed a recognition system that was demonstrated to successfully locate a number of objects in a realistic home environment. We note that they consider object categories with little variation between instances, that object viewpoint is not modeled in their work at present and that their example scenes appear less cluttered than the ones we study in this work. However, their system considers a wider range of the problems facing a domestic robot, when compared to the work of this thesis. This includes interaction with humans in the environment using natural language dialogue in order to create plans based on the users requests. Combining our perceptual techniques into a larger integrated system of this type is a direction for future work.

A number of problems related to those in this thesis were studied during the construction of the Stanford Artificial Intelligence Robot (STAIR) [GAK$^+$07]. They have considered an information-based visual guidance routine for peripheral-foveal vision and estimated accurate object pose during recognition [SDN09], for example. The family of robots named *Personal Robot* (PR1 [WBDS08] and PR2 developed commercially by Willow Garage) are also capable recognition platforms and have been used by several authors for tasks such as the construction of semantic object maps [PTPB12].

In the domain of visual understanding within the workspace of a robot arm, Rasolzadeh *et al.* [RBHK09] describe a vision system that enables the arm to interact with a set of objects. Their use of visual saliency and object segmentation in this work is quite similar to the visual attention system of the Curious George robot. However, their relatively simple object models and overall goal of achieving successful object grasps for simple objects differs significantly from recognition of generic objects within unstructured environments.

The robots described in this section use a wide variety of approaches for each system component, but all have been demonstrated to be reasonably successful at recognizing objects in a number of visual environments. This demonstrates the diversity of available solutions in this domain. Our efforts to participate in the Semantic Robot Vision Challenge and to publish a public dataset collected by our robot are motivated by the need to compare between platforms of this type.

2.3 Active Vision for Object Recognition

Control of the robot and its sensor pose is a critical aspect of the robotic object recognition problem. The robot's camera must not only continue to observe the object within the field-of-view, but it should also ideally find the viewpoints that most assist with recognition. This is closely related to the so-called *Active Vision* [AWB88] problem from computer vision, where the only distinction is whether constraints from the robot's control system are considered. Early work by Whaite *et al.* [WF94] identified the importance of modeling uncertainty, or the lack of information about certain pieces of information, and to find control actions that minimize this uncertainty. This concept has been applied to the problem in a wide variety of work including: to combine object-related planning with a visual attention process [DCTO97], to guide recognition based on optical flow features [AF99] and to find objects jointly with recognizing indoor place-types [GAP$^+$11], among numerous others.

One challenge for a viewpoint-control system is that planning trajectories of many steps is a costly task, where each single step can require significant simulation from a probabilistic object viewpoint model. Vogel *et al.* [VM07, VdF08] have proposed several computational methods to allow multi-step planning for viewpoint control. These approaches are similar to the related problem of finding multi-step trajectories that allow for efficient robot localization [PR09, FBT98, SJC00].

Another approach to achieving efficiency in the planning process is considered by Laporte *et al.* [LA06], who consider efficient computation over single actions. We will describe this method is detail as it motivates our approach in Chapter 6. Laporte uses a Bayesian model to integrate object information over the collected views and models several latent variables such as the object's orientation and the scene lighting. Principle Component Analysis (PCA) is used to produce simple object appearance models for a number of viewpoints. A necessary step in many viewpoint selection algorithms is computing the distribution of the next unseen observation, conditioned on potential motions under consideration as well as the latent variables. In the case that there are many unknown dimensions, computing this distribution can be time consuming and often requires sampling to approximate the desired marginal (e.g., with Metropolis-Hastings). Laporte demonstrates that the observation model can be factorized and that several components may be pre-computed after the object model is known, to minimize test-time computation. Our work adopts the Bayesian evidence integration formulation, but we have used modern object recognizers such as the DPM technique, which is largely invariant to illumination. So, we have not found the need to model lighting as a latent variable.

Active Vision and robotic viewpoint planning are often considered in small workspaces, as this allows for systematic evaluation and restricts the planning dimensions. However, it is also important to evaluate viewpoint planning approaches on robots in less constrained environments. Our Curious George robot planned motions through a contest area on the order of 100 m^2 during the SRVC contest. Other examples of physical robots that use viewpoint planning for physical robots over relatively large spaces include the work of [Eid10].

2.4 Recognition from Heterogeneous Data Sources

In several sections within this thesis, we fuse visual appearance models with sensed geometric information, such as that from a laser range finder. Other authors have also considered fusing information between depth sensors and images [GBQ$^+$08, SXBS10, QBG$^+$09, FSD10, HL10, RLXF11]. In some cases, depth information has been used as an indirect input, which guides the recognition process but does not directly appear in the feature representation. For example, Sun *et al.* developed a depth-aware Hough Transform [SXBS10] and several authors have considered using depth to reduce the set of scales to be searched at each image pixel [FSD10, HL10], during visual recognition. In contrast, both Lai *et al.* [RLXF11] and Quigly *et al.* [QBG$^+$09] extract features from both visual and depth imagery, so that the resulting appearance models have primary contributions from both sensor types. Our work can be classified with those that make indirect use of sensed ranges. We use depth both to reason about occlusion and to locate objects in three dimensions, but do not extract appearance features from the range data.

2.5 Applications of Object Recognition by a Mobile Platform

A number of authors have considered the use of detected objects, along with minimal spatial information for each object, in order to perform the task of place recognition. Vasuvedan *et al.* [VGNS07] use expected co-occurrence distributions between object counts and the type of room to label places (e.g., office, hallway, or kitchen). Ranganathan *et al.* [RD06] also model a place by the set of objects located there, but study the task of localization using sets of detected objects.

In some cases, the interaction between the spatial properties of objects and humans in the world is the primary concern. For example, Grabner *et al.* [GGV11] parse the geometry of a scene by analyzing what locations are likely to be "sittable" by a human as part of their procedure to locate chairs in the environment. Kulić *et al.* [KC05] have developed a system for safe interaction between a robot arm and a human working in close proximity that leverages visual information about the human within the planning algorithm.

In the case of autonomous driving safety, it is essential to recognize and locate nearby cars, people and traffic signs. Numerous methods have used object recognition from visual images, range data or the combination to solve this task. Prisacariu *et al.* [PTZ$^+$10] have used the recognition of street signs from multiple viewpoints to assist in driver safety. Dolson *et al.* [DBPT10] consider the important problem of interpolation between sparse points of range data in order to produce detailed geometric object descriptions while driving, which could be incorporated into our work on car recognition in Chapter 8 in future work.

A number of authors have combined object information with video cues such as motion and moving segmented regions in order to automatically build an understanding of space. Swadzba *et al.* [SBWK10] use recognized objects along with motion information (i.e., foreground and background segmentation) to construct a so-called "articulated scene model". Xu *et al.* [XK10] use moving objects as part of their object semantic hierarchy which is a general description of the world that can be used during developmental learning.

2.6 Evaluation Procedures for Object Recognition

Evaluation procedures are required to judge the quality of various object recognition approaches and to describe their expected performance on new tasks. Realistic data is required for this analysis, and we will discuss a number of relevant data sources shortly. Recognizers can describe object locations in a variety of ways (e.g., with 2D bounding boxes, 3D bounding volumes, with or without poses) and this must be respected during performance evaluation. Here we will briefly discuss some of the most common evaluation algorithms for several types of object descriptions.

Evaluating 2D Bounding Boxes

The standard evaluation method for algorithms that localize objects as a 2D bounding box is precision and recall (PR) analysis and the average precision (AP) statistic. Each object box hypothesized by the recognizer is determined to be either true or false, based on whether it overlaps sufficiently with an annotated object region. A point on a PR curve represents a precision, the ratio of true positives to number of hypothesized objects, and a recall, the ratio of true positives to number of annotated objects. Each point corresponds to a thresholding of the recognizer's results at a single confidence value. A curve is formed by simulating a range of threshold values. Average precision is a summary statistic that captures performance across all possible thresholds with a single number by measuring the area under the PR curve. Perfect performance on the task would give all of precision, recall, and average precision equal to 1.0 which indicates every object instance is correctly located without any false positives.

Note that computing a PR curve for an arbitrary recognizer can be quite a complicated task, as each hypothesis must be labeled as a true or false positive for a variety of confidence thresholds. Also, multiple object categories, multiple instances of each category, and multiple hypothesized detections from the recognizer may appear in the same image. Decisions such as how closely a hypothesis must agree with the ground truth, how to handle multiple hypotheses for the same true object, and how to handle the image boundary can greatly change the final result. In the computer vision community, the evaluation methods of the PASCAL VOC challenge [EVW$^+$12] are the most widely accepted, and we have followed these closely throughout this thesis. In particular, the VOC criteria requires the ratio between intersection and union of bounding boxes to exceed 0.5 in order for there to be a true positive match, allows only a single object hypothesis to be a true detection for each labeled object and provides for truncated and difficult flags that allow the annotator to exclude problematic object instances from consideration.

Evaluating 3D Object Volumes

Recognition algorithms that estimate the position of objects in 3D can be evaluated against datasets with 3D object annotations. In principle, the PR analysis describe above is still an appropriate quality metric. In practice, computing the overlap between pairs of oriented volumes can produce misleading results, as this measure confounds errors of several types including: position, scale and orientation. When 3D volumes are derived from image data, there is also an inherent non-uniformity in resolution (i.e., close objects are measured at higher resolution and can be localized more accurately). To address these issues, other authors have previously considered the elements of 3D localization independently. Specifically, they evaluate the accuracy of predicting the object's 3D centroid and also the accuracy of predicting orientation. We will discuss approaches for each of these tasks.

Bao *et al.* [BS11] have developed an evaluation method that compares only the estimated and ground truth 3D centroids.Their approach thresholds the error in predicted centroid by a constant, σ. A reasonable value for this threshold has been considered to be the maximum dimension for a particular object category, such as 2.5 *m* for cars. Evaluation at a variety of thresholds can also be performed to better describe the sensitivity of 3D localization. After labeling each hypothesized detection as true or false, precision and recall can be computed in the same fashion as was described for 2D bounding boxes.

A numer of authors including [SFF07, SGS10] have utilized confusion matrices over eight viewpoint-bins (i.e., left, right, forward, backwards and the four intermediate views) to evaluate the accuracy of object pose estimation. A two-phase process is required to form such a confusion matrix. First, we must associate 3D object detections to ground truth regions. This can be done by taking only the objects that were scored as true positive during 3D localization analysis, so that an assignment is available. Each hypothesis-annotation pair places a vote into one of the matrix entries. When the estimated pose matches the true pose, a vote is given to a diagonal entry, while errors result in votes for off-diagonal entry. The resulting confusion matrix allows for visualization of the common errors of pose estimators, such as the off-diagonal bands that arise from nearly symmetric objects.

2.7 Similar Object Datasets

This thesis describes a recently-established evaluation benchmark specifically tailored to the robot recognition problem, named the University of British Columbia Visual Robot Survey (UBC VRS). It has been motivated by the example of object recognition in computer vision, where rapid progress has been made through standardization around the PASCAL VOC challenge [EVW$^+$12] and benchmark tasks for distinguishing large numbers of object categories, such as Caltech 101 [FFFP04] and 256 [GHP07]. Several robotics challenges exist, including the Semantic Robot Vision Challenge (SRVC) [SRV] and Solutions in Perception Challenge [Bra], which compare near real-time systems on robot recognition tasks at a particular venue once per year. These contests capture the full scope of robot recognition, but the requirement to travel to the contest location in order to participate limits their accessibility.

Several datasets based on RGB-D data such as that available from the Microsoft Kinect have recently been released. For example, the Berkeley 3D Object Dataset [JKJ$^+$11] is composed of many indoor scenes contributed by the community through *crowd-sourcing* and annotated by humans. While there are more images and more object types in this dataset than the one we present, each scene is captured from only a single viewpoint, which does not allow exploration of recognition methods involving robot motion. The Multi-View RGB-D Object Dataset by Lai *et al.* [LBRF11] includes a large number of scenes containing a single object on a turn-table, captured with an image-depth sensor from a number of viewpoints, as well as a smaller number of scenes containing multiple objects captured with hand-held trajectories. This dataset allows for rapid iteration and direct comparison between methods, but the single trajectory through each scenes precludes its use in the study of active perception.

In order to learn the viewpoint detection function for an object recognizer, validation data containing multiple viewpoints of numerous instances of each category is required. Many image-only databases containing multiple viewpoints of objects have recently been made available, however, we found that many of them did not fit our purposes. For example, Viksten *et al.* [VFJM09] collected a database with fine-grained viewpoint sampling for each object, but only a single instance of each category is present, as their efforts have been targeted towards grasp planning for industrial applications. The use of the Internet as an on-line forum for users to annotate data has been used to produce very large labeled databases such as LabelMe [RTMF08]. Also, on-line task auction sites are suitable for dataset construction and have been used in ImageNet [DLD$^+$09] and also by [VG09]. These large category datasets have so far not been annotated with image viewpoint. A dataset collected by Savarese *et al.* [SFF07] contains 72 views (8 azimuth angles, 3 heights and 3 scales) of each of 10 instances for 10 common object categories. While containing far fewer object instances than some other resources, the precise viewpoint labels associated with each image make this dataset suitable for evaluation of multi-view techniques, and it will be used to construct our viewpoint detection response functions in Chapter 6.

We primarily evaluate the techniques of this thesis on the UBC VRS dataset, and on the

Ford Campus Dataset of Pandey *et al.* [PME11]. We will describe the VRS data in detail in Chapter 4 and will give more information about the Ford Campus data in Section 8.7. These datasets were selected because, at the time of this thesis, they represented the largest available sources of data where the images contained realistically cluttered visual scenes and where registration information was readily available between different viewpoints. The VRS data contains indoor kitchen objects, and the Ford Campus subject matter is largely parking lots in an urban area. This variety allows us to demonstrate that our techniques are relatively general and to discuss differences between the situations, such as the fact that the appearance of cars have strong correlation with viewpoint, while many kitchen objects are roughly cylindrically symmetrical.

The contribution of our UBC VRS evaluation benchmark is to allow the unique aspects of the robot recognition problem to be explored with statistical significance and repeatability. These aspects include: the use of 3D and visual sensory data; the ability to actively control the robot's path and influence the series of images obtained; and the challenge of cluttered scenes present in real environments.

2.8 Multiple Viewpoint Recognition and Tracking in Three Dimensions

The core of our work is recovering 3D object information from a sequence of 2D visual imagery captured from a variety of viewpoints. We have drawn inspiration from a number of authors who previously performed similar tasks in a variety of domains including inference of human motion from videos and object recognition from multiple images. Specifically, we focus our discussion on methods that leverage the recent advances in object recognition from single images, using per-image object recognizers as an input to later tracking layers, which is commonly known as *tracking-by-detection* [LCCV07]. The most similar approaches to our own are those that use detections in several images to infer 3D object information (e.g., [WRSS10, ESLV10, BS11, WWR$^+$]).

A canonical example of these approaches is [WWR$^+$], which is one of the top performing methods for identifying cars and people in urban driving data as of the writing of this thesis. This approach first applies an object recognizer to all input images. So-called *tracklet* reasoning is then performed, which takes into account that, in video data, subsequent frames represent very similar moments in time. This observation is exploited by merging detection hypotheses which are nearby in image space in neighboring video frames to form tracklets over a short number of frames (e.g., three to five are commonly used values for tracklet length). Tracklets are useful in that they eliminate spurious false detections that occur only in a single frame, without support from neighboring frames. Wojek lifts tracklets into 3D object detections through an inference procedure that accounts for inter-object occlusion, but not for occlusion by the background or by un-modeled objects. Each object is modeled in 3D space and the projections of the objects into each image are related to the tracklet evidence through a mixture-of-experts framework that can incorporate object part information in each view, if available. The final output of this method are tracked 3D cars and people near to the data collection vehicle.

Our work shares many aspects with [WWR$^+$]. Our Chapter 7 is a collaboration between the author of this thesis and Wojek which extends their previous approach to indoor data where sensed range is available. Chapter 8 further extends the 3D model by adding detail for the object parts. Note that, because our data is composed of multiple discrete images that can sometimes be from entirely different locations within the scene, we cannot apply the tracklet pre-processing step. This makes our inference problem slightly more challenging, as we must relate 3D objects to individual detections, which are more likely to be incorrect than entire tracklets.

Several other approaches for urban driving differ slightly from this primary approach. Ess *et al.*[ESLV10] utilized stereo as well as structure-from-motion during the tracking process. The so-called Semantic Structure from Motion (SSFM) method of Bao *et al.* [BS11] has a similar core formulation to the approach just described but, almost uniquely within the literature, the authors attempt to recover the full camera pose information by adding point correspondences along with object detections. They optimize jointly over object and camera locations. Their results represent one of the first indications that semantic object information can be an important input to position estimation for a moving camera. This type of semantic mapping is an interesting direction for future study.

Several innovations to the basic *tracking-by-detection* framework have occurred in the domain of image space tracking of players in sports videos. Okuma *et al.* [OTD$^+$04] track players in sports videos using a particle filter formulation that explicitly combines detection results from an Ada-Boost [FS97] classifier in the tracking likelihood. Lu *et al.* [LOL08] expand upon this formulation by additionally inferring the pose of the players at each point along the track. Most recently, [BRL$^+$09] have demonstrated that the initiation of tracks only from object hypotheses of a recognizer, which in practice are simply local maxima of a detection scoring function, is not an optimal approach. Rather, they reason about the raw score map that underlies the usual detection results. Intuitively, this more faithfully represents the appearance likelihood for an object appearing at a particular location and scale, as it is not affected by image space non-maxima suppression techniques.

Our work is also similar to approaches that recover 3D human pose from video. The approach of Andriluka *et al.* [ARS10] has been particularly inspiring for our approach. The authors applied multi-view reasoning to recover the time-evolving 3D pose of human subjects from video data. In contrast to our analysis of static scenes, typical human motion is a strong cue for such analysis and a Gaussian Process Latent-Variable Model (GPLVM) motion prior proves an effective regularizer for inferring the motions. Similar to our approaches, their method leverages bank-of-detector appearance models to perform weak viewpoint prediction, and they consider this viewpoint variable when reconstructing the direction of the human's body.

Several authors have performed recognition of kitchen objects from a number of viewpoints. Helmer *et al.* [HMM$^+$10] was an early collaboration including the author of this thesis. That work included many elements of the multiple viewpoint reasoning approaches that are included in this thesis, but it did not attempt to infer accurate 3D object locations. Rather, 3D information was represented implicitly, by reasoning about the compatibility of 2D detections in various images. More recently, Susanto *et al.* [SRS12] have considered using multiple Kinect sensors with overlapping fields of view. The positions of the sensors were pre-registered with the Iterative Closest Point (ICP) approach, so that the input data resembles that which we use for the experiments in much of this thesis. However, Sustanto combines features extracted from the depth data and combines these with the visual features, while we model appearance only from visual imagery. Finally, [LBRF12] have also considered Kinect data with kitchen objects as the subject matter. In their method, visual detections from a number of viewpoints are used to label the point cloud that is produced by fusing range information from many viewpoints of a scene. This detailed surface analysis, at the level of pixel-accurate segmentation, is a different output than is produced by our system, but it is a potential avenue for future work.

2.9 Chapter Summary

This chapter has surveyed the previous work in the fields of computer vision, robotics and machine learning that is most similar to our own. We depend on a number of previous object recognition methods and spatial models. Those approaches have been described here in detail to simplify later discussion.

The next chapter will begin the detailed discussion the contributions of this thesis by describing the Curious George robot platform that has been used as a physical testbed for evaluating recognition algorithms and to collect data for off-line experiments.

Chapter 3

The Curious George Visual Object Recognition Platform

3.1 Introduction

In this chapter, we will describe an intelligent system, comprised of a physical robot platform and accompanying software algorithms, that was assembled to support the robotic object recognition research described in this thesis. This platform, named Curious George, has directly or indirectly been the test-bed for the majority of our research. Testing our algorithms on a physical platform has provided insights into the challenges that face such systems, allowed the automated collection of a dataset that will be described in the Chapter 4, and validated that the algorithms described in the remainder of the thesis are practical and successful in the real world.

This chapter will outline the aspects of Curious George that have the most impact towards its success as a visual recognition platform. A suite of sensors measure the robot's surroundings including: a high resolution visual camera, a stereo camera with lower resolution, and several laser range finders. Several control algorithms allow the robot to safely and effectively navigate in its environment and gather visual information suitable for extracting semantic information. These include: a peripheral-foveal visual attention system; a navigation, localization and mapping technique; and top-level heuristic planners that sequence the order of simpler robot behaviours. The chapter will conclude by presenting laboratory-based empirical evidence suitable for evaluating the attention control algorithm and describing the performance of Curious George in an international contest that established its state-of-the-art performance amongst existing research platforms with the goal of object recognition.

The material contained in this chapter is based upon [MFL$^+$08], however it contains significant additional description of work on the Curious George platform that occurred after that publication. The geometry-based attention and SRVC sections are the two most notable additions.

3.1.1 Statement of Collaboration

While the author of this thesis has been the lead researcher responsible for the creation of Curious George, a relatively large number of other students and researchers at the UBC Laboratory for Computational Intelligence (LCI) have collaborated on aspects of the work. Curious George was used as the University of British Columbia's entry into the SRVC in three separate years. In each case, a team of students collaborated on preparing the robot for the contest and the produced system was made public as open-source code. The work described throughout this thesis has leveraged that SRVC code-base.

Per-Erik Forssen and Kevin Lai were highly involved in the initial selection of sensors, development of navigation methods and the creation of a prototype object-finding behaviour. Scott Helmer, Sancho McCann and Ankur Gupta have implemented a variety of recognition algorithms for early testing on the robot. These are distinct from any recognition approach described in the remainder of this thesis, but their presence facilitated robot design, allowed successful performance in the SRVC contests, and have inspired the techniques used in our later work. Marius Muja and Matthew Dockrey assisted with hardware design and electronics of the final physical form of the robot as well as the porting of many of the previously existing algorithms into the Robot Operating System (ROS) software framework. Marius Muja was also instrumental in collaborating on the geometry-based attention mechanism.

3.2 Goals and Design

In the robotic recognition scenario that faces a robot in an ever-changing home environment, recognition, navigation, planning (both for robot motion and the robot's view), and interaction must all occur simultaneously. The robot needs to avoid obstacles to operate

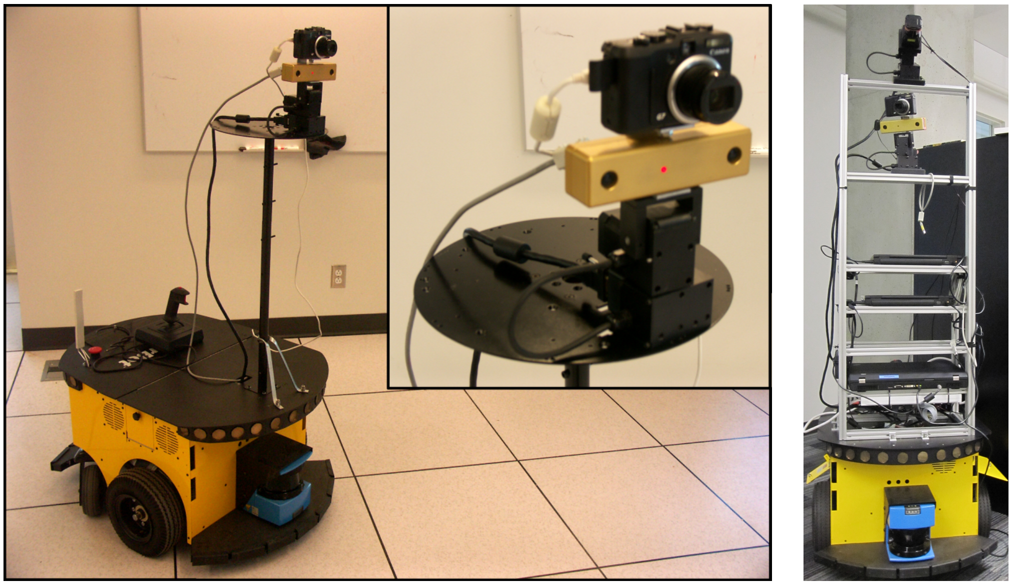


Figure 3.1: Curious George Hardware: Two iterations of our robot platform.

safely. Camera view control is also an essential aspect, as the sensor must capture objects and places of interest. Only once these basic competencies are accomplished can higher-level tasks such as object recognition be completed.

The Curious George project began with the goals of accomplishing these basic robotic abilities, to enable long term study of robot recognition algorithms. We were inspired initially by the work of Ekvall *et al.* [EJK06] and Ranganathan *et al.* [RD07], as both of these approaches produced systems with the basic capabilities to navigate and image simple environments. However, many challenges remained. For example, we seek systems that make more efficient use of robot motion, target the camera intelligently, and integrate the learned visual object representations with other robot behaviours.

Our goal at the outset of the Curious George project was to design behaviours that allow numerous, high-quality views of each of the objects to be collected efficiently. This should largely be completed *before performing object recognition* by quickly identifying promising objects and regions, which we will refer to as *potential objects*. The identification of candidate object locations without evaluating object models everywhere leads to greatly increased computational efficiency. This pre-semantic identification of interesting regions was inspired by the model of human visual attention proposed by Rensink [Ren00], where *proto-objects* are detected subconsciously in the visual periphery, and attention shifts between these to allow more detailed consideration. Note that we distinguish detection of proto-objects from the object discovery task. As described by Southey *et al.* [SL06], object discovery methods attempt to faithfully segment meaningful objects using numerous cues. In comparison, we produce a less precise segmentation with less computation and rely on subsequent recognition to refine the result.

This chapter will continue by describing the system that resulted from our implementation of these goals. We will first describe the robotic hardware and sensors. We will then discuss the attention system that guides the robot's camera to capture high-quality images of proto-objects. Next, we will discuss Curious George's ability to build spatial models of its environment, and to use those models to navigate safely and effectively. The chapter will conclude with discussion of our results in the lab as well as in the international SRVC robotic recognition contest.

3.3 Hardware

Hardware design is an important consideration when constructing a robot that is targeted at operating in a man-made environment. Many extant robot platforms have limited ability to perceive interesting objects due to their height, navigation ability or fixed-direction sensor platforms. For example, objects located on desks or bookshelves in an office are often too high to be seen by a robot's cameras. Our robot platform, Curious George, was designed to have roughly similar dimensions and visual dexterity to a human, so that relevant regions of the environment could be easily viewed and categorised. Our robot is an ActiveMedia PowerBot. A SICK LMS 200 planar range finder is mounted horizontally, roughly 0.1 m above the floor. This laser is referred to as the base laser and is used for 2D navigation and mapping. The robot's cameras are raised by a tower with height approximately 1.5 m. The cameras are mounted on a PTU-D46-17.5 pan-tilt unit from Directed Perception which provides an effective 360° gaze range. See Figure 3.1.

We employ a peripheral-foveal vision system in order to obtain the high resolution required to recognise objects while simultaneously perceiving a large portion of the surrounding region. This choice has again been modeled after the human perceptual system, and was also inspired by design choices made in [KB06]. For peripheral vision, the robot has a Bumblebee colour stereo camera from PointGrey Research, with 1024×768 resolution, and a 60° field-of-view which provides a low resolution survey of the environment. For foveal vision, the robot has a Canon PowerShot G7 still image camera, with 10.0 megapixel resolution, and 6× optical zoom which allows for high resolution imaging of tightly focused regions.

Curious George possesses a second means of collecting 3D range information. A small and light planar laser range finder from Hokuyo has been mounted on a pan-tilt unit and is swept over the scene with a periodic motion pattern. The planar scans are assembled, utilizing the sensor's geometry, to form a densely sampled point cloud over a portion of the robot's environment. One example of such a 3D point cloud is visualized in Figure 3.2.

3.4 Attention System

The attention system identifies potential objects in images from the peripheral vision system. It then focuses on these objects to collect detailed images using the foveal system, so that the detailed images can be further processed for object recognition. Identifying potential objects correctly is a non-trivial problem, due to the presence of confusing backgrounds and the vast appearance and size variations amongst the items that we refer to as objects. Our system makes use of multiple cues to solve this problem. The depth information from the robot's stereo camera and tilting range finder are used to perform simple structural decomposition of the environment. This can filter large completely flat regions likely to be unoccupied floor or table regions. Areas above a threshold of curvature are more likely to be objects. We process visual information directly with a saliency measure to detect regions with distinctive appearance. This section will describe the structural and saliency aspects of the attention approach in detail. We will then describe the subsequent collection of high-quality images from the robot's foveal camera.

3.4.1 Geometry-based Attention

As mentioned, the range data available from stereo vision or the tilting laser sensor is useful to produce a structure-based attention operation. Using the robot's accurate internal calibration, we can transform the measured range values into a global frame where the geometry of the floor plane is known (e.g., a Z-up frame with known floor height). A simple height threshold can then be used to separate floor regions from objects supported by the floor (see Figure 3.3). The resulting noisy binary map is cleaned-up by a series of morphological operations. This helps to remove small disparity regions, which are likely to be erroneous, and also fills in small gaps in objects. The resultant obstacle map is used both to avoid collisions with objects and tables while navigating, and in combination with saliency to determine likely locations of objects that sit on the floor, which may be pieces of furniture, or objects directly resting on the floor plane.

Within each segmented floor-supported region, a second phase of processing is applied to determine if the region is likely to be furniture that, in turn, supports one or more smaller objects of interest. To accomplish this, we apply an approach described by [Rus09]. Horizontal planes are robustly detected within the sensed structure using the Randomized Sampling and Consensus (RANSAC) algorithm [FB81]. This algorithm attempts to find a model that supports a sufficiently large set of inlying points by iterative sampling. In our case, the model equation is that of an infinite 2D plane embedded in the 3D space of our sensory data:

$$\hat{n} \cdot [X - X_0] \quad = \quad 0, \tag{3.1}$$

where $\hat{n}$ is the 3D normal vector to the plane and X_0 are the 3D coordinates of one selected point on the plane. All sensed 3D points satisfying Equation (3.1) are so-called plane inliers, and the target of the plane-finding algorithm is to obtain model parameters with sufficiently many inliers.

The next step in the geometry-based attention operator is analyzing the structures supported by planar surfaces. A simple agglomerative clustering approach, called Euclidean Clustering is used to determine contiguous regions and to segment these from one another. Figure 3.2 provides a visualization of the bounding volumes determined by Euclidean Clustering for one single table viewed by Curious George. For well-spaced collections of objects that all have sufficient height to appear distinct from the supporting plane, this algorithm can return groupings that perfectly match the real physical objects. When objects touch eachother, or are low and difficult to separate from the table plane, the algorithm can either falsely group objects together or falsely split a single object in two. Therefore, further stages of analysis are needed to correctly determine the shape of the objects present, and we refer to the outputs of the geometric attention operator as potential or *proto-objects*.

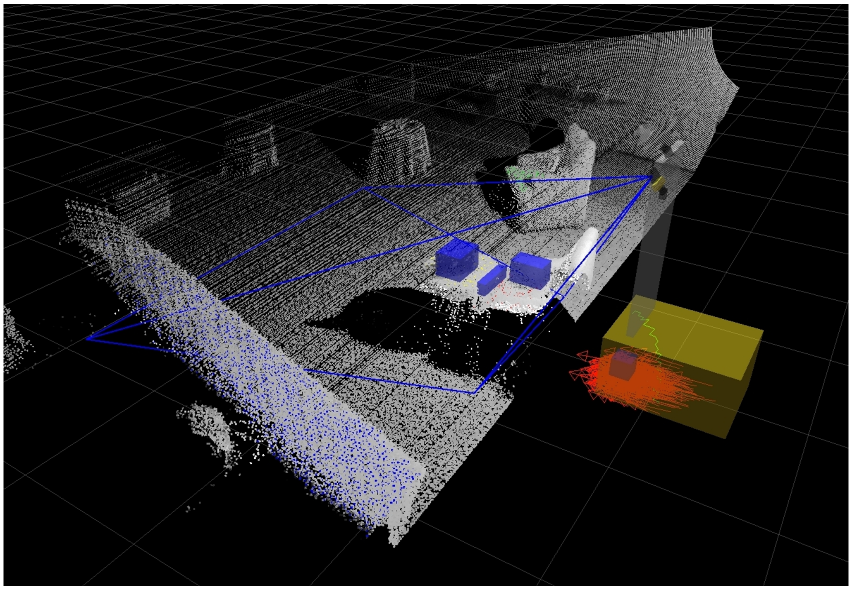


Figure 3.2: Geometry-based Attention: The 3D point cloud data assembled from the tilting laser range finder is displayed along with visualized results of the geometry-based attention operator.

3.4.2 Visual Saliency-based Attention

The second cue used by our attention system to determine likely objects of interest is the visual saliency of regions within the peripheral camera images. We compute the saliency of every point within the image using a modified version of the spectral residual saliency measure defined in [HZ07]. We extend the measure to colour in a manner similar to [WK06]. That is, we compute the spectral residual on three channels: intensity, red minus green, and yellow minus blue. The results are then combined by summing the channels to form a single *saliency map*. Regions of multiple sizes are then detected in the saliency map using the *Maximally Stable Extremal Region* (MSER) detector [MCUP02]. This detector is useful since it does not enforce one single partitioning of the scene. Instead, nested regions can be detected, if they are deemed to be stable. Typically, MSERs are regions that are either darker or brighter than their surroundings, but, since bright in the saliency map corresponds to high saliency, we know that only bright regions are relevant here, and consequently we only need to run half the MSER detector. Bright MSERs are shown in red and green in Figure 3.4. Regions are required to have their smallest saliency value above a threshold

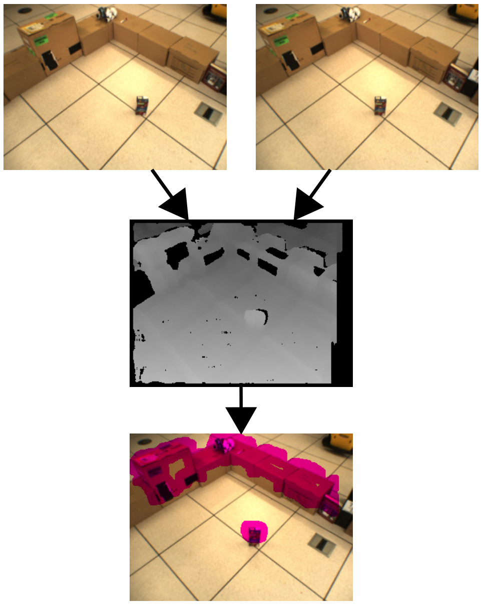


Figure 3.3: Stereo Processing: Top to bottom: Left and right input images, disparity map, and obstacle map superimposed on right input image.

proportional to the average image intensity, which is justified since spectral saliency scales linearly with intensity changes. This gives us automatic adaptation to global illumination and contrast changes. The regions are further required to be more than 20% smaller than the next larger nested region, to remove regions that are nearly identical. To ensure that the salient regions are not part of the floor or support surface, they are also required intersect the obstacle map (see Section 3.4.1) by 20%. Regions which pass these restrictions are shown in green in Figure 3.4.

Compared to [WK06], which can be considered state-of-the-art in saliency detection, the above described detector offers three advantages:

1. The use of spectral saliency and the MSER detector makes the algorithm an order of magnitude faster (0.1 instead of 3.0 seconds per peripheral image in our system).

2. The use of the MSER detector allows us to capture both objects and parts of objects, whenever they constitute stable configurations. This fits well with bottom-up object detection. Since objects typically consist of smaller objects (object parts), we would not want to commit to a specific scale before we have analysed the images further.

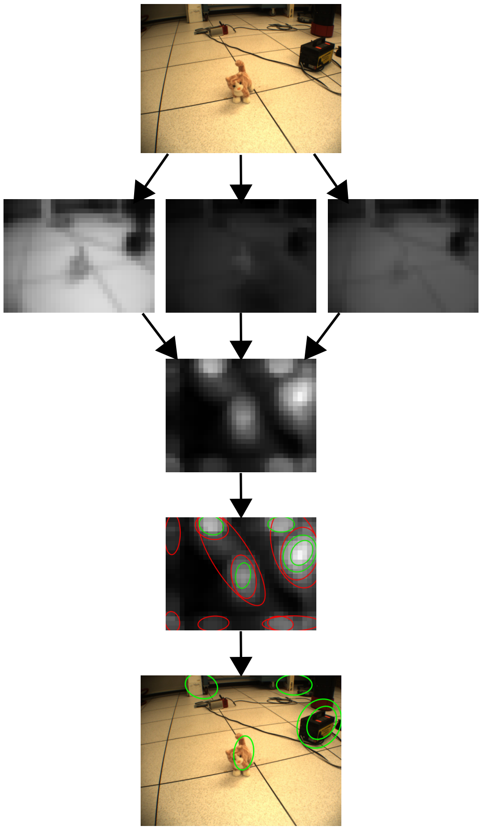


Figure 3.4: Saliency Computation: Top to bottom: Input image, colour opponency channels (int,R-G,Y-B), spectral saliency map, detected MSERs, and MSERs superimposed on input image. Figure best viewed in colour.

The multiple sizes also map naturally to different zoom settings on the still image camera.

3. The use of an average intensity-related threshold allows us to adapt the number of salient regions reported, depending on the image structure. In particular, this thresholding technique will report that there are no salient regions when analysing a highly uniform image such as a wall or floor. This is in contrast to the Walther toolbox [WK06], which, due to its built-in normalisation, only orders salient regions, but does not decide that there is nothing interesting in the scene. Their normalisation approach could also be modified to take into account relation of region saliency to average im-

age intensity, but this would not be straightforward due to non-linear effects in the method.

Note that the potential objects from visual saliency are not necessarily what a human would normally call objects. They are equally likely to be distracting background features such as intersecting lines on the floor, or box corners. The purpose of saliency is merely to restrict the total number of possible gazes to a smaller set that still contains the objects we want to find. This means that it is absolutely essential that the attended potential objects are further analysed in order to reject, or verify their status as objects. We will briefly describe the recognition techniques used on Curious George in Section 3.7. A detailed descriptions of the recognition techniques developed within this thesis can be found in Chapters 5 – 8.

3.4.3 Gaze control

In order to actually centre a potential object in the still image camera, we employ the saccadic gaze control algorithm described in [For07]. This algorithm learns to centre a stereo correspondence in the stereo camera. To instead centre an object in the still image camera, we centre the stereo correspondence on the *epipoles* (the projections of camera's optical centre) of the still image camera in the stereo camera.

In order to select an appropriate zoom level, we have calibrated the scale change between the stereo camera and the still image camera for a fixed number of zoom settings. This allows us to simulate the effect of the zoom, by applying the scale change to a detected MSER. The tightest zoom at which the MSER fits entirely inside the image is chosen.

3.5 Spatial Representation

An embodied recognition system must be able to move through its environment in order to obtain perceptual information. Performing this motion safely, planning collision-free paths to new locations, and reasoning about the expected value of perceptual information that can be obtained from each location requires a system to have 3D spatial-awareness. Curious George achieves this awareness through a well-calibrated model of its on-board sensors, an accurate map model of its surrounding environment that is constructed online from sensory data, and a localization system that tracks the robot within the environment map. We will describe these components in the remainder of this section.

3.5.1 System Calibration

Perceptual information is used to build all models of the environment surrounding a robot, and the first step in relating perceptions over time and space is an internal model that describes how various sensors and devices on the robot are spatially related. This internal model, which we will call system calibration, has been partly collected from the physical specification and design of Curious George, and the remainder of unknowns are determined accurately through a calibration procedure.

We have calibrated each of the robot's cameras independently using the technique described by [TL87]. This procedure inolves capturing numerous images of a planar target from a variety of viewpoints. The calibration algorithm locates points in the images of the target whose real physical locations are known precisely. An optimization procedure determines a model for the camera which best fits the observed correspondences. In practice, this procedure produces camera calibration accurate to less than a pixel of reprojection error, so long as sufficiently many calibration images are collected, the calibration target has been carefully constructed, and the correspondence finding process is able to correctly locate all image points with no false matches.

The pose of the laser relative to the cameras is estimated with the technique of [UH05]. This information is represented as 3D rigid-body transformations between frames representing the position and orientation of each sensor, composed with dynamic frames for moving parts such as the pan and tilt units that actuate the sensors. The laser to camera calibration also uses a planar target, but in this case it is observed simultaneously by both sensors. The optimization is over the relative pose between sensors, and is somewhat less accurate in final results due to the need to match information between two sensors with different noise characteristics. It should be remembered that mapping of laser information into images is therefore less trustworthy than geometric operations based only upon the cameras.

Using the combination of our system specification and calibration information, Curious George is able to relate the information from each sensor to its base frame, and to produce coherent spatial estimates of quantities in the world relative to a still robot. However, the robot moves through the world, so it is also necessary to register between the poses of the robot at different times. This is solved by mapping and localization, which will be described in the next section.

3.5.2 Mapping and Localization

Our system performs mapping with FastSLAM, a Rao-Blackwellized Particle Filter implementation [MTKW03], which builds a probabilistic occupancy grid map [ME85] based on readings from the laser range finder readings from the robot's planar base laser and the robot's odometry, and simultaneously tracks the robot's position within the map. An occupancy grid is well suited to guide navigation and planning tasks for a mobile robot moving on a flat surface since it mirrors the inherently 2D nature of this environment. The robot produces a map as it moves through an environment for the first time. For accurate maps, several so-called loop-closures, or repeated observations of the same location, are required. Several example maps produced by Curious George are visualized in Figure 3.5.

Upon building a complete map, the robot can optionally cease the updating of the map and transition into a localization-only procedure within a static representation of the environment. Curious George uses a sampling-based localization package based on the work of [FBT99] to estimate its position within a static map.

3.6 Planning and Control

We have described the sensors that Curious George uses to observe its world and the algorithms used to spatially relate sequences of these observations. We next describe the set of planning and control procedures that allow the robot to autonomously explore an environment and to locate instances of a variety of object categories. These behaviours must allow the robot to cover its environment. That is, the time-evolving geometric map should eventually be a complete representation of the 2D traversable space within the environment. This requires moving the robot's base to sufficiently many locations in the world so that the planar range finder observes each wall and surface in 2D. The 3D world should also be covered by the visual sensor, so that all object instances are observed. Finally, in order to allow high confidence in final decisions, each object should be imaged from a variety of viewpoints. We describe a set of planners to achieve these goals in the remainder of this section.

3.6.1 Low-Level Safety and Navigation

Safely and effectively moving through an environment to a target point within a map is a fundamental behaviour for mobile platforms, and is the goal of Curious George's low-level

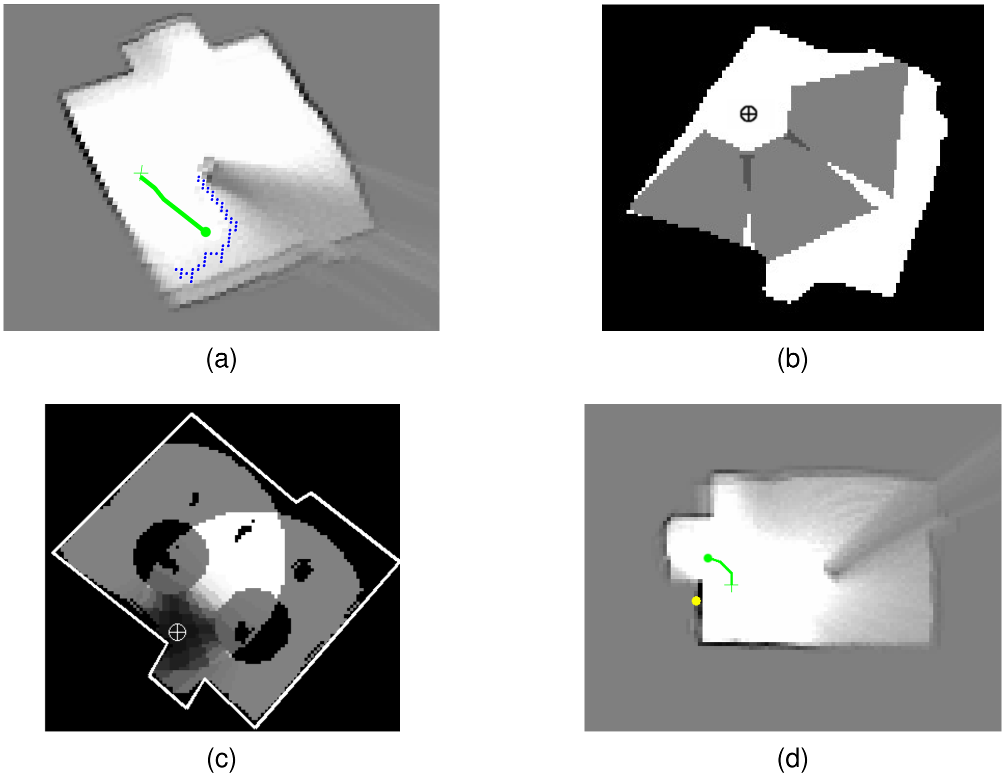


Figure 3.5: Sample Paths: (a) Paths towards the frontier of unexplored space (indicated by blue dots) allow for exploration of the entire environment. (b) Three potential camera-fields-of-view are considered to achieve coverage by the visual sensor. (c) The object permanence cost function considers the value of capturing each object from a new viewpoint. (d) A path to another clear view of an object (indicated by a yellow dot) is chosen. Legend for images (a) and (d): + start of path. • end of path.

navigation routines. In early versions of the robot platform, we implemented A^*-search through the occupancy grid map to produce a path from the robot's current position to its goal, which avoids obstacles. The robot then attempted to actually follow this path and reacted to moving objects and map changes using the Vector-Field Histogram local planner described by Borenstein et al. [BK91]. In later iterations, we found that a robust and Open-Source implementation of similar behaviour was made available within the ROS architecture. Curious George currently uses the so called nav-stack from ROS to achieve safe navigation to goal points.

3.6.2 Exploration Planning

We employ the frontier based exploration technique described by Yamauchi et al. [YSA98] to quickly cover the environment with the laser scanner and produce an initial map. As is illustrated in Figure 3.5(a), a frontier is defined as the border between explored and unexplored space. For our system, these frontiers will be the locations just beyond the range of the laser scans, and in the laser shadows created behind objects or around corners. The frontier planning technique identifies candidate locations where laser scans would be most likely to uncover new regions to explore. First, one of these promising locations is chosen, then the robot moves to this location using the low-level navigation routines through the partial map, and the map is updated. This process is iterated, until all regions have been explored.

3.6.3 Visual Coverage Planning

Each time a region of the environment is observed with the peripheral camera, the attention system has the opportunity to detect potential objects within that area. In order to maximise these opportunities, the camera should be pointed in directions that cover as much new territory as possible. We use an iterated greedy search based on visible area weighted by the number of previous observations to select favourable directions. This approach causes the camera to cover the environment roughly uniformly and give an equal chance of detecting potential objects in any location. One snapshot of the cost function employed by this planner is depicted in Figure 3.5(b).

3.6.4 Object Permanence

We refer to object permanence as a visual-spatial memory preserved by the system after objects or *proto-objects* have initially been identified. The task of the object permanence planner is to attempt to obtain additional observations of these promising regions for puposes of verifying or becoming more confident about their identities. As mentioned in the descriptions of our visual attention operators, the *proto-objects* located by attention are often incorrectly segmented. The set of *proto-objects* also includes distracting non-object items such as texture on support surfaces, which repeated observations may be able to filter.

Even once the system establishes an initialy hypothesis about the category label of an object, further verification is often wise. A major reason for this is that the set of available object poses in visual training data is often incomplete. One tends to get the characteristic views [Pal99] (e.g., a shoe is normally photographed from the side, and hardly ever from the front), rather than a uniform sampling of views. A second reason is that objects are often more recognizable from some viewpoints than from others. A cannonical example is a bicycle, for which side views yield significantly more visual features than front or rear views. In order to perform successful recognition in the face of limited training data and biased 3D object properties, we attempt to collect numerous views of each potential object by repeatedly looking back to the same locations as the robot moves. This requires awareness of an object's location even when it is not in the visual field, which we refer to as *object permanence*.

The behaviour of looking back from many views increases the likelihood that one of the collected images is taken from a similar view to that of the training data. However, if the sampling over viewpoints is performed randomly, significant duplication may occur. Curious George's *object permanence* planner attempts to obtain unique viewpoints by allowing the previous views of an object to vote for nearby angles into a histogram with values in the range $[0, 2\pi]$. Histogram bins with low scores are selected. That is, views from a completely new direction are favoured over those from similar angles. We employ greedy search over histogram values and iterate the procedure to obtain roughly uniform coverage of viewing angles. Once a direction is selected, the hierarchical planning method moves the robot to the desired viewing position and a foveal image is collected. Figure 3.5(d) shows an example of a path produced during this behaviour. Please note that this simple object permanence planner will be extended in Chapter 6 with a more detailed planning approach that accounts for learned object properties.

3.7 Visual Appearance Modeling

We have included a variety of visual recognition routines on the Curious George platform. These routines are responsible for analyzing the images collected and forming hypotheses about the category labels for the objects present in each. The earliest set of visual recognition routines were developed by collaborators of the author of this thesis. So, they are not described in detail here. These include a specific object recognition method based on

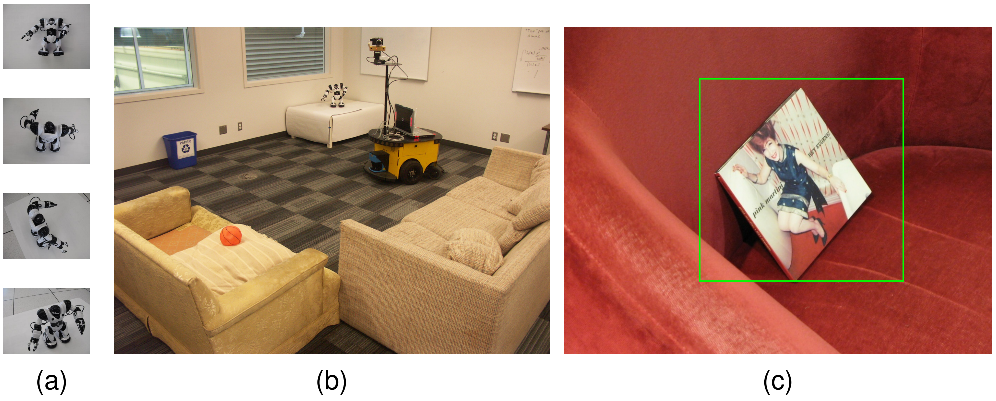


Figure 3.6: Spatial-semantic Maps: Combining the spatial awareness provided by SLAM with object recognition, meaningful object labels can be assigned to locations in the map. (a) Training data for object "robosapien". (b) Overview photo of the room the robot is exploring. (c) The map with three objects, and the locations from which they were observed.

matching SIFT features with geometric verification over sets of potential matches and several object category recognizers including the so-called spatial pyramid matching approach of Lazebnik *et al.* [LSP06]. We will describe the visual learning and recognition algorithms developed within this thesis in subsequent chapters.

The autonomous exploration and planning behaviours of Curious George are largely agnostic to the specific recognition method, however several basic criteria are required in order to interface with the geometric reasoning and planning modules. Specifically, object hypotheses must contain localization information including scale, as this is essential for estimating the 3D position of the object in the world, which is leveraged for object permanence planning and active vision.

3.8 Experimental Results

We have evaluated the performance of the Curious George recognition platform both with experiments in our laboratory as well as by entering the platform in an international contest for visual recognition. This section will begin by describing the laboratory experiments, where we describe the goal as semantic mapping, or correctly placing semantically meaningful objects into a spatial representation of the world. We will then describe the Semantic

Robot Vision Challenge and the results of Curious George on that task.

3.8.1 Semantic Mapping

The combination of techniques described in the previous sections endow a mobile agent with the ability to autonomously explore its environment and to recognise the objects it discovers. In simple environments, this behaviour can be extended to spatial-semantic mapping by back-projecting the recognised objects into the robot's map representation of the world. Later in this thesis, we will consider more sophisticated algorithms that are capable of locating objects in 3D, even in cluttered environments. The probabilistic occupancy grid constructed from laser range scans fed through the FastSLAM algorithm can be augmented with the locations of visual objects. For example, Figures 3.6(b) and 3.6(c) illustrate the locations of objects matching the labels “robosapien”, “basketball”, and “recycling bin”. The object recognition subsystem was provided with between 2 and 4 example views of each object, see Figure 3.6(a) for an example. Each object shown was identified by the attention system and observed from various locations, giving several pieces of information about its position, and allowing for collection of numerous views for recognition or future matching. We envision that the types of maps illustrated here could be easily used in a human-robot interaction system where the human operator would be able to relay commands to the robot in semantically meaningful terms.

3.8.2 Comparison of Attention Approaches

To validate the effectiveness of the saliency and structure based attention systems described in Section 3.4, we compared its performance against two other methods for selecting foveal views which will be described shortly. In order to ensure a fair comparison, the remaining components of our system were held constant. This suggests the following decomposition of our system into three parts:

1. (Identical for each method) Robot motion to a location that allows coverage of the environment and collection of a peripheral image of a large region at low resolution.

2. (Three different attention methods compared) Selection of a number of sub-regions and collection of foveal images.

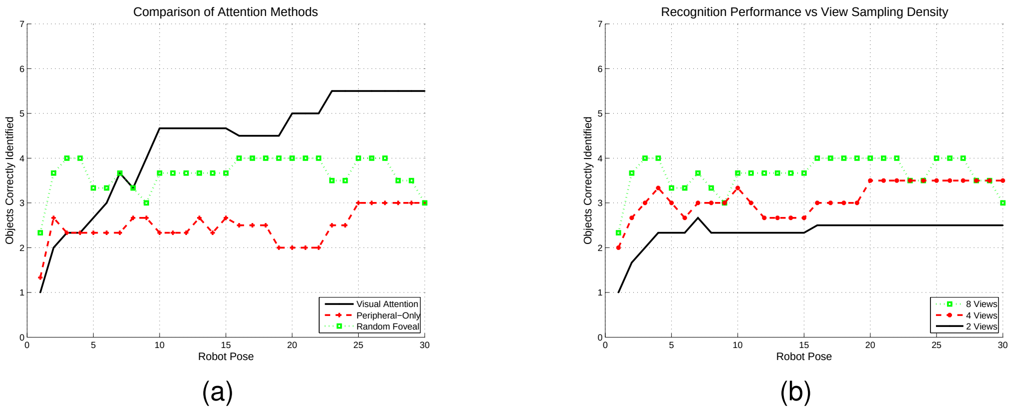


Figure 3.7: Attention Comparison: Methods were compared in a number of ways. (a) Shows recognition results for the three classes of approaches. (b) Demonstrates that for the random view selection approach, recognition performance increases with the parameter *n*, which is the number of foveal views sampled at each robot pose. Each result is averaged over 3 separate runs of the robot.

3. (Identical for each method) Classification of the collected foveal images by the object recognition system.

The three attention methods evaluated were the visual saliency and structure approach described in Section 3.4 and two comparative methods:

1. *Peripheral view only*. This method took only one low-resolution peripheral image at each robot pose, simulating the lack of a foveal vision system. The image covered the entire peripheral region at a wide zoom setting. Recognition results from this approach should be viewed as a baseline for any more selective attention system.

2. *Random view selection*. This method sampled from sub-regions of the peripheral view by randomly selecting *n* pan-tilt and zoom values from the view-cone visible in the peripheral camera, where *n* is a tunable parameter. The space of possible images collected by this method is the same space in which the guided attention system searches. As such, there is some likelihood that these samples are identical to the interesting views obtained by the guided attention system, or are even better views

(by chance). So, the recognition results from this approach can be used to evaluate whether or not our guided system is better than chance at selecting interesting views.

To additionally enforce fairness of comparison, we ran each of the three attention methods from identical robot locations. That is, once the robot had moved to a point, each one of the three attention methods was executed in sequence. The pan-tilt unit and zoom settings were reset to their defaults between each method, and the robot base was kept stationary during the process.

Figure 3.7 displays the evolution of recognition performance over time for each of the 3 approaches. To demonstrate the utility of using attention to guide the robot's camera, we allow the random approach to sample many more views than our guided approach. Specifically, Figure 3.7(a) describes a trial where the random view sampler uses 8 images at each location. The attention method varries the number of images it takes based on the saliency of the visual content at each location, but in this trial it averaged 3 images per robot pose. Even with many more overall images taken by the random strategy, results show that the planner based on our visual attention system is able to recognize more objects correctly. The *peripheral-only* method, which only has access to a small number of low-resolution images (1 per robot pose) performs worse than both of the methods that obtain high resolution images from the foveal system.

To confirm that our comparison was fair, we performed several additional runs where we varied the parameter *n* for the random view selection strategy, to observe its effect on recognition performance. Figure 3.7(b) demonstrates that for all of 2,4, and 8 views per pose, the collection of more data generally increases performance. Again, even with more than twice as many foveal images, the random approach does not perform as well as our guided attention measure. This is strong evidence that the visual saliency method is indeed guiding the robot to obtain promising views of objects, and that it is performing well in realistic scenarios.

3.8.3 The Semantic Robot Vision Challenge Contest

Our second approach to verifying our methods experimentally was to enter the Curious George platform in several iterations of the Semantic Robot Vision Challenge, which we will describe in detail shortly. In both the 2007 and 2008 SRVC contests, Curious George obtained the highest score on the robot hardware version of the contest and was awarded first place. In 2009, due to a software malfunction, the physical robot platform was unable to complete the navigation portion of the contest, however the accompanying learning, filtering and recognition system placed first in the software-only version. This section will provide a brief overview of the SRVC contest format and will provide discussion of the results of each Curious George's three contest performances.

3.9 SRVC Description

The goal of the SRVC contest was to evaluate the ability of the scientific community to produce a complete physical and computational system capable of recognizing a wide variety of objects in the real world. The contest organizers specified the following priorities: a limited amount of manual labeling is required for systems to learn models of new objects; robots are completely autonomous in their movements and decision-making; the visual survey of environment should happen relatively rapidly; and both generic object categories and also specific object instances should be recognized. Note that an important aspect of the SRVC was to enforce sharing of source code by each participating team. The source code related to this chapter is included online along with that of all other contestants[1].

The SRVC contest took place in three phases. First, systems were given time to learn object models. To eliminate the reliance on manually collected training data, the list of twenty target objects for each year's contest was not released ahead of the contest. Rather, on the day of the contest, teams were given the list as a text file. This file had to be the starting point for a learning procedure where the robot produced appearance models solely from this text file plus any internal resources that had been prepared and an Internet connection. In practice, all teams used web-based image search engines such as *Google Image Search*, as well as on-line databases of labeled objects such as ImageNet [DLD$^+$09]. These sources provided training data for each of the target objects. This first training phase of the contest took place over a period of time that ranged from four to twelve hours in different years. At the conclusion of phase one, all download and processing of the training images must have completed, yielding appearance models suitable for use in subsequent phases.

The second phase in the contest was robot exploration of a real physical environment. The contest organizers placed one or more instances of each object along with distracting objects and furniture in a section of a conference room with clearly delineated boundaries.

[1] http://www.semantic-robot-vision-challenge.org/teams.html

Contestant robots were required to autonomously explore this environment and collect as much perceptual information as possible for the purposes of locating the objects. Any robot that contacted the furniture or objects was technically disqualified, although small exceptions were made in practice. Only very minor human interaction with the robots for the purposes of safety was allowed. Note that phase two of the SRVC encourages the type of coverage and object permanence planning that we have described previously in this chapter. Any object in the environment that is not quickly captured by the robot's sensors through autonomous planning cannot be recognized by the platform, even if the learned appearance model for that object is flawless. This was a common failure mode of teams entering the contest.

The third SRVC phase was automated object recognition based on the perceptual data collected. Each robot's answers were delivered to the contest organizers, in real-time for two out of three years, in the form of images labeled with a bounding box and object name. The image, bounding box and label were all produced automatically from the software system running on each robot. Scoring of these results was done with a procedure similar that defined by the PASCAL VOC contest [EVW$^+$12], which we have described in Section 2.6. The SRVC scoring slightly modified the basic procedure to weight correct detections by localization accuracy. Perfect overlap with the human's annotation was worth full points while increasing amounts of mis-match lead to lower scores, and any mislabeled object received zero or negative points, depending on the year.

In each year, the contest was split into two independent leagues: the robot league and the software league. The robot league, where participants required a physical platform to explore the environment, involves all three phases described above. The software league is a modified version where the organizers provided a robot to explore the environment and collect a standardized perceptual experience, effectively removing phase two, along with the need for contestants to provide their own physical robot platform. Teams in the software league only performed web-based appearance learning and then detected objects within the standardized image set to obtain a score.

3.9.1 Summary Results

Over the three years in which the SRVC occurred, Curious George placed first in the robot league of the contest twice and first in the software league once. Table 3.1 summarizes the performance of each of these systems. In the 2009 contest, Curious George received a zero score for the official robot league contest because the autonomous control system crashed due to a power malfunction. Curious George was restarted manually, which disqualified it from official scoring. However, our unofficial analysis based on the performance after the manual restart showed that the robot's score would have been similar to the score we achieved in the software-only league, and approximately triple the score of the official robot-league winners for that year.

Some trends were apparent in the results of Curious George and all other participants in the SRVC contest. The scores for recognizing specific instances were always significantly greater than those for recognizing generic object categories. The temporal trend for all participants in the contest was for stronger performance in each subsequent year, although details of the objects chosen and contest setup led to significant year-to-year variability.

Year	**Category Performance**	**Instance Performance**
2007 - robot	1 correct and 1 partial out of 9	5 correct out of 10
2008 - robot	0 correct out of 10	5 correct and 1 partial out of 10
2009 - software	3 correct out of 9	10 correct out of 11

Table 3.1: SRVC Result Summary: The scores for the Curious George platform for the first place finishes in various contest leagues and years.

The remainder of this section will briefly describe the objects selected and in the contest environment prepared by the contest organizers in each of the three years of the SRVC contest, and will describe the results of the Curious George platform in each year.

3.9.2 SRVC 2007 at AAAI

The 2007 SRVC contest was held in Vancouver, Canada at the Conference of the Association for the Advancement of Artificial Intelligence (AAAI). In this first iteration, the contest environment was relatively simple. Objects were placed on regularly shaped tables covered by white tablecloths as well as on floors and chairs. The majority of objects were well-spaced. The contest area was relatively uncluttered, so it was possible for nearly all of the objects to be observed from a single position in the environment. This meant that only minimal robot motion would have been required. However, this fact did not prevent Curious George and other platforms from traveling significant distances during the contest.

Year	Type	Name	Result
2007	category	scientific calculator	no attempt
		fork	incorrect
		electric iron	incorrect
		banana	incorrect
		green apple	incorrect
		red bell pepper	correct
		rolling suitcase	incorrect
		red plastic cup	partial
		upright vacuum cleaner	incorrect
	instance	Ritter Sport Marzipan	no attempt
		book “Harry Potter and the...”	no attempt
		DVD “Shrek”	no attempt
		DVD “Gladiator”	correct
		CD “Hey Eugene” by Pink Martini	correct
		Lindt Madagascar	incorrect
		Twix candy bar	incorrect
		Tide detergent	correct
		Pepsi bottle	correct
		yogurt Kettle Chips	correct

Table 3.2: SRVC 2007 Results: Detailed results of the robot-league performance of Curious George for the 2007 SRVC.

Table 3.2 describes the detailed results obtained by Curious George and Figure 3.8 provides example result visualizations.

3.9.3 SRVC 2008 at CVPR

The difficulty of the visual task was increased significantly for the second version of the SRVC contest, which was held in Anchorage, Alaska, in 2008 at the Conference for Computer Vision and Pattern Recognition (CVPR). A much wider variety of furniture was used to support objects, which significantly limited the visibility-range of objects and made robot motion a necessity. The furniture included included tables of various sizes and height as well as chairs placed independently, or in realistic proximity to tables (i.e., objects on chairs were nearly underneath tables). Also, the spacing between furniture and other obstacles was reduced, so it was more difficult for robots to navigate through the environment and fewer objects were clearly visible from any single location. Figure 3.9 provides several example images from the 2008 SRVC environment, and Table 3.3 summarizes Curious George's results from that year.

The selection of objects was changed, although several of the same objects were retained. Notably, one object category, *ulu*, was unique to the Alaskan setting. Only a small number of examples were available in the web-based training data for this category, and performance was poor overall. This highlighted the limitations in using generic image sources to learn about object appearances rather than operating with specific knowledge of the location.

For SRVC 2008, teams were encouraged to report results in real-time through a bonus point system. A correct answer delivered by a robot in real-time was worth one additional point.

Year	**Type**	**Name**	**Result**
2008	category	apple	incorrect
		saucepan	incorrect
		remote control	incorrect
		digital camera	incorrect
		upright vacuum cleaner	incorrect
		banana	incorrect
		eyeglasses (not present at contest time)	incorrect
		fax machine	incorrect
		ulu	incorrect
		frying pan	incorrect
	instance	CD "Retrospective" by Django Reinhardt	correct
		book "Paris to the Moon" by Adam Gopnik	incorrect
		Spam	correct
		Ritz crackers	correct
		book "Big Book of Concepts"	correct
		DVD "I, Robot"	partial
		DVD "300"	incorrect
		game "Crysis"	incorrect
		Doritos Blazin' Buffalo Ranch	incorrect
		Kiwi Strawberry Snapple	correct

Table 3.3: SRVC 2008 Results: Detailed results of the robot-league performance of Curious George for the 2008 SRVC.

3.9.4 SRVC 2009 at ISVC

The final SRVC contest was held in Las Vegas, Nevada, in 2009, at the International Symposium for Visual Computing (ISVC). The robot arena was the largest of all in this year, and for the first time objects were placed on the boundary of the environment, which provided a challenge for platforms to distinguish objects on the boundary from the crowd of spectators just beyond. Figure 3.10 shows several examples images from the 2009 SRVC environment. The object selection was changed once more, and several of the object categories were purposely selected to be mutually similar in appearance (e.g., orange, pumpkin, soccer ball). Similar to 2008, objects were placed such that they were hidden from all but a small range of positions, so robots needed to cover the space thoroughly. The most varied set of supporting furniture was chosen in 2009 with multi-layered wood steps and a table with textured shelves and detailing being added to a selection of tables and chairs that was similar to those used in the previous two years. For the first time, in 2009, a subset of the object categories was released prior to the contest date. This was meant to give teams the advantage of being able to carefully train appearance models before arriving at the contest. Also, real-time result reporting was a requirement in 2009, rather than a recommendation. Table 3.4 displays the results obtained by the software portion of the Curious George system in 2009, since the physical platform was not able to officially compete in that contest iteration.

3.10 Chapter Summary

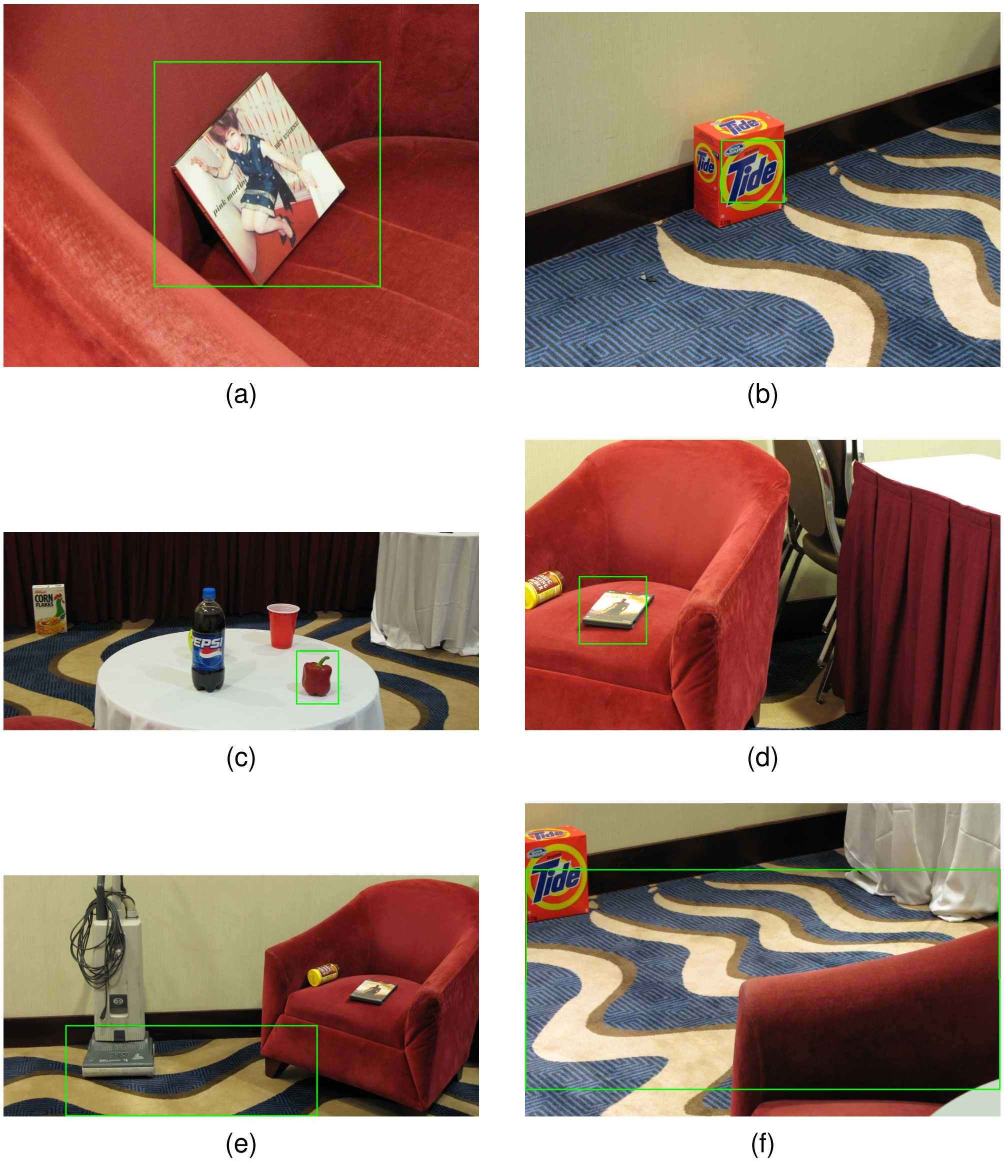
This chapter has described the Curious George robotic system, developed as part of the author's doctoral studies. Curious George is able to autonomously navigate through environments and find objects. A set of planners driven by saliency and several heuristics allow the robot to accomplish this task relatively efficiently. We have shown results both in a laboratory setting that we constructed for the purposes of illustrating the robot's abilities as well as in an international contest organized by members of the robotic recognition community. In both cases, Curious George successfully navigated through the environment with a large degree of autonomy. The planners and recognition approaches succeeded in directing the robot's sensors and correctly locating many of the objects in the environment.

The primary limiting factor in all of the tasks described in this chapter has been the final step of correctly labeling an object once it has appeared in one or more visual images.

Year	Type	Name	Result
2009	category	pumpkin	incorrect
		orange	correct
		red ping pong paddle	incorrect
		white soccer ball	incorrect
		laptop	incorrect
		dinosaur	incorrect
		bottle	correct
		toy car	incorrect
		frying pan	correct
	instance	book “I am a Strange Loop” by Douglas ...	correct
		book “Fugitive from the Cubicle Police”	incorrect
		book “Photoshop in a Nutshell”	correct
		CD “And Winter Came” by Enya	correct
		CD “The Essential Collection” by Karl ...	correct
		DVD “Hitchhiker’s Guide to the Galaxy”	correct
		game “Call of Duty	correct
		toy Domo	correct
		Lay’s Classic Potato Chips	correct
		Peperidge Farms Goldfish Baked Snack ...	correct
		Peperidge Farm Milano Distinctive ...	correct

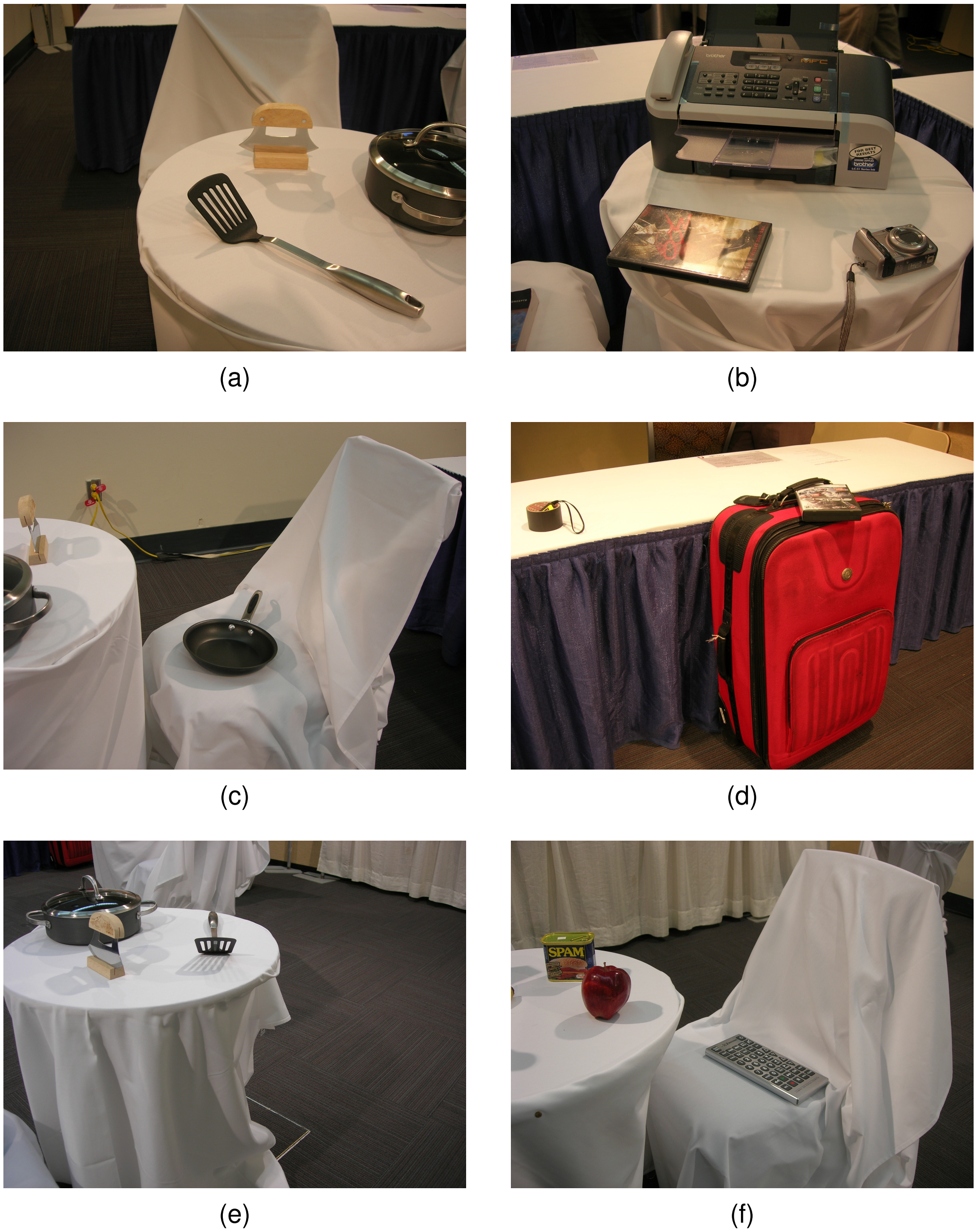
Table 3.4: SRVC 2009 Results: Detailed results of the software-league performance of Curious George for the 2009 SRVC.

Failures in this step occur because object recognizers have a large number of false positives, leading to objects being hypothesized at incorrect locations. Also, objects are often incorrectly localized in 3D because the image space 2D localization of the recognition methods is imperfect. Finally, 3D effects such as occlusion and non-informative appearance from the observed viewpoint often lead to errors in final object recognition results. This observation has motivated the majority of the remainder of the work in this thesis to focus on the task of more robustly recognizing objects from a particular set of images rather than on refining our attention approaches for obtaining good images or on controlling the robot’s base within the environment.



(a) (b) (c) (d) (e) (f)

Figure 3.8: SRVC 2007 Images: Recognition results recorded during the official run of the 2007 SRV Contest. (a-d) High quality views obtained by the focus of attention system allow for correct recognitions. (e-f) The system's best guesses at objects for which no good views were obtained. These are clearly incorrect.



(a)

(b)

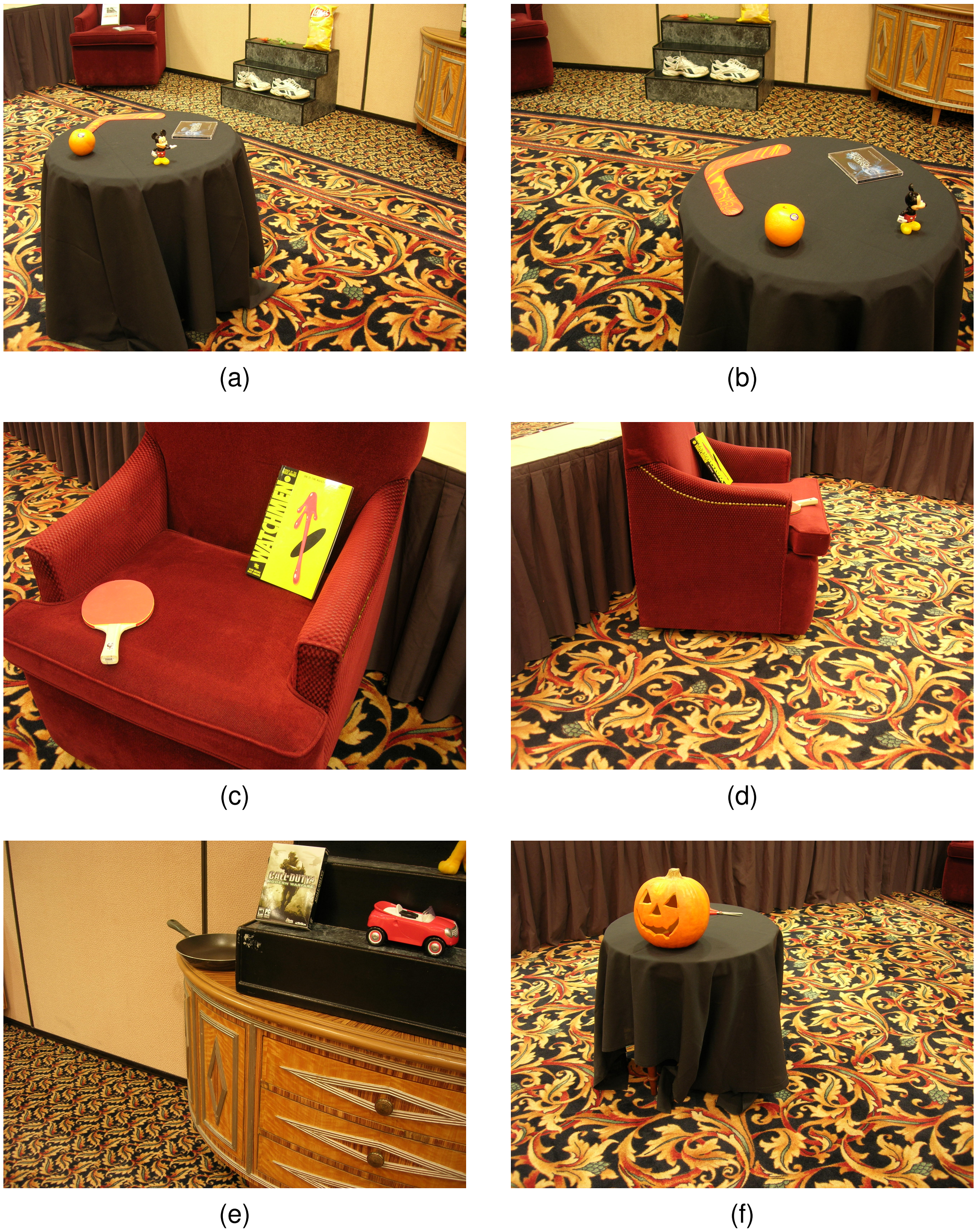
(c)

(d)

(e)

(f)

Figure 3.9: SRVC 2008 Images: Images of the 2008 SRV Contest environment, located in Anchorage, Alaska.



(a) (b) (c) (d) (e) (f)

Figure 3.10: SRVC 2009 Images: Images of the 2009 SRV Contest environment, located in Las Vegas, Nevada.

Chapter 4

The UBC Visual Robot Survey Benchmark Dataset

4.1 Introduction

Repeatable experimental procedures and benchmark tasks have been instrumental to recent progress on a number of tasks in automated perception. Upon completing of the SRVC challenge, we sought a repeatable benchmark that would allow testing of a robot's ability to recognize objects using the full set of cues that were available to Curious George: visual images, knowledge of the motion that had occurred between images, active control of this motion and 3D range sensing. Standard visual recognition benchmarks typically consisted of datasets of images with labeled objects. These either lacked auxiliary geometric cues or, as for [Min09], the geometry linking the images was imprecise. On the contrary, robotic datasets often contained accurate geometric information regarding the robot's trajectory, but the visual content typically contained few objects and these were not annotated.

This chapter describes a new benchmark task that we have produced, known as the UBC Visual Robot Survey (VRS), which captures all of the uniquely robotic aspects of object recognition. We used a robot to exhaustively sample sensory information from a number of environments and record the robot's trajectory, its visual images and the sensed 3D information. During training and testing of robot recognition algorithms, the recorded data can be provided to recognition algorithms by a simulator that mimics a robot's sensing and response to control input by selectively replaying perceptual elements from the recorded data. We refer to this procedure as *simulate-from-real-data*. Except for small limitations due to sampling discretization, this allows repetition of the same perceptual and control feedback experience that a robot would experience.

In order to ensure that our benchmark adequately mimics real-world conditions, we replicated cluttered kitchen-like indoor scenes composed of many mutually occluding objects. A small number of instances from many naturally-occurring object categories were present and annotated, but we emphasized three categories by purchasing many instances for each of: mugs, bottles and bowls. The VRS contains on the order of one hundred unique instances for each of these three highlighted categories. This allows for the division of the data into training and test sets, while maintaining sufficient statistical support for generalized performance. We collected information from thirty different scenes, each with a unique background, unique set of object instances and unique layout of those objects. Overall, the UBC VRS dataset is a reasonable approximation of the visual recognition task that would face a robot operating in a kitchen. Performance on our benchmark is likely to predict performance within real kitchens. This is supported by evaluation of our own methods on both the UBC VRS and real kitchens later in this thesis (see Figure 7.5).

4.1.1 Statement of Collaboration

This chapter is an extended version of the material previously published in [ML12]. The author of this thesis was the originator of the UBC VRS concept, programmed the majority of the tools, collected most of the data and produced all manual annotations. However, numerous researchers provided essential discussion and assistance at various stages. The author worked with Scott Helmer and Marius Muja during the writing of [HMM$^+$10]. That paper used an early version of the dataset where 3D range information was not available and the data was collected with a hand-held camera, rather than by a robot. The cube registration target that we will soon describe was developed during that work and a number of the software tools were developed such as the registration procedure to triangulate 3D points from 2D correspondences. During the collection of the robotic data, Hana Yoo assisted with some of the early Graphical User Interface development during her stay at the Laboratory for Computational Intelligence as a summer volunteer.

Christian Wojek and Bernt Schiele, co-authors of [MWSL11], had crucial input on the necessary quality of 2D annotations for a modern vision dataset, which led to a complete

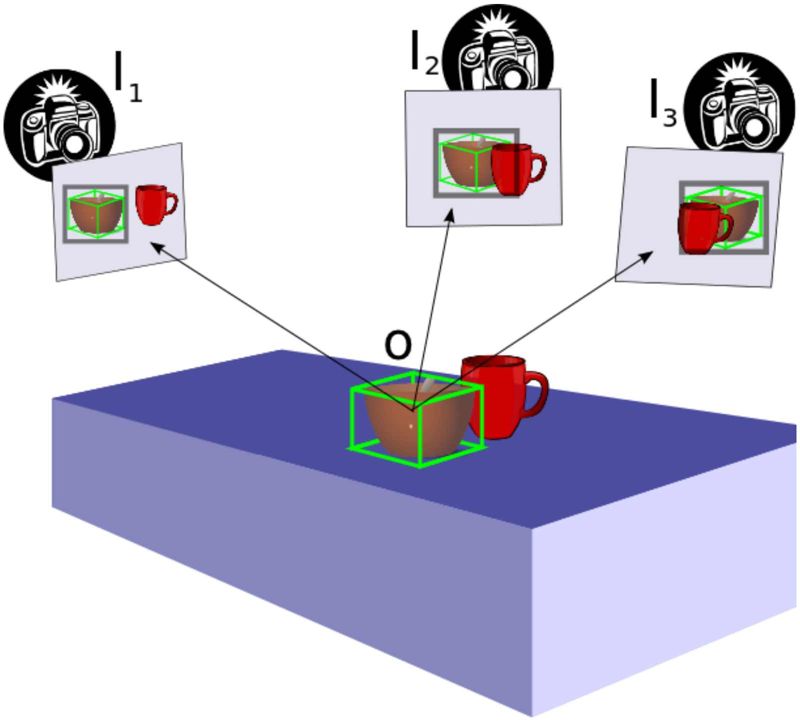


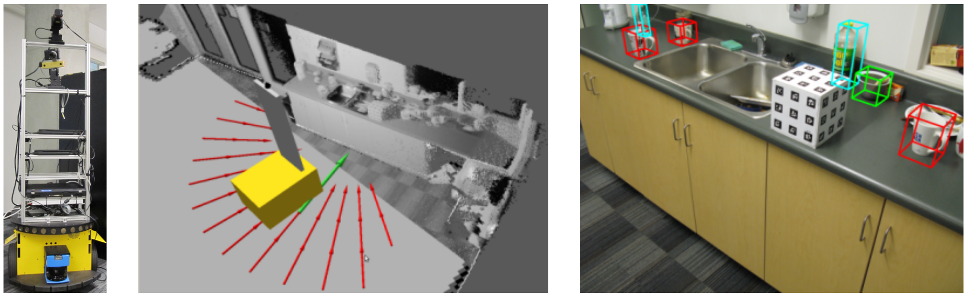
Figure 4.1: UBC VRS Overview: Our robot collects a number of images of a scene. Geometric registration allows 3D object information to be projected into each image. Accurate 2D bounding box annotations are also provided.

re-annotation and the highly reliable bounding boxes now present in the dataset.

Finally, the author also kindly thanks Mark Fiala and the National Research Council of Canada (NRC) for making the AR Tag [Fia05] library available for research use, first through release on the NRC website and later through licensing with the purchase of Dr. Fiala's text [CF08]. Accurate registration between the numerous images contained in our dataset would have required extensive manual effort or a costly motion-capture setup, but both of these were avoided by building our calibration target from AR Tag markers which can be automatically localized with sub-pixel accuracy and nearly zero false positives.

4.2 UBC Visual Robot Survey Dataset

This section describes the method use to collect, register and annotate the data that forms the UBC VRS. Briefly, we recorded the actual sensory experiences of the Curious George robot



(a) (b) (c)

Sub-set	Scenes	Views	Instances	Boxes	Mugs	Bottles	Bowls
Training	19	453	184	4026	56/1405	42/1119	17/378
Validation	11	295	116	2701	33/711	33/839	17/346
Test	30	334	303	3466	85/935	57/589	64/681
UBC VRS	60	1082	603	10193	174/3051	132/2547	98/1405

(d)

Figure 4.2: UBC VRS Details: (a) The Curious George robot platform used for data collection. (b) A sample point cloud, and poses from the survey path followed by the robot. (c) A sample image with 3D wire-frames projected to display user-annotated ground truth volumes. (d) Summary statistics of the annotations available for the UBC VRS database. The final 3 columns represent the (unique instances / number of bounding boxes) that are present for the selected category. Note that the dataset contains numerous additional categories that could not be shown specifically, but are included in the overall totals.

while it was commanded on a trajectory that covered many of the reachable poses within a number of environments. The visited poses are registered to a consistent coordinate frame using a cube-shaped registration target that has visually identifiable patterns on each face. A human manually annotated the locations of all object instances from several categories, both in the 3D coordinate frame and within each collected image. Figure 4.1 illustrates the final product of this procedure, which is robot sensor data from a set of viewpoints of each scene, along with geometric knowledge relating all data to a common coordinate frame and object annotations in both 3D and 2D.

4.2.1 Robotic Data Collection

The sensor data that comprises the UBC VRS dataset was collected with the Curious George robot that was described in Chapter 3 and is shown in Figure 4.2(a). During data collection, the robot moved through a dense set of poses covering the space of possible visual experiences. We achieved this by planning a path consisting of three concentric circles. Along each circle, stop-points were located at an angular spacing of at most ten degrees, and at a finer resolution for selected scenes. When the robot reached each stop point, it turned to face the center of the scene and it collected a single reading from each of its sensors. Figure 4.2(b) shows a sample path in one environment.

In the ideal case, this data collection method ensures that, for every robot pose, T, in the environment, there exists a real sensor reading in our dataset that is less than 5 degrees in horizontal angle (i.e., azimuth) from T. However, constraints of our robot and the environments prevented a complete sampling. Factors such as building layout, uneven floors, and furniture obstacles caused the robot's navigation routines to skip some of the requested stop-points. Data from these skipped viewpoints is not available to recognition methods, which is also the case for real robotic systems exploring an environment. Recognition methods must therefore be robust to this realistic property of the dataset. Figure 4.2(d) displays the final number of images and scenes that were collected.

As has been stated in Chapter 3, the Curious George robot has a variety of sensors suitable for object recognition. Images from the robot's high-resolution digital camera capable of 10 mega-pixel imaging were down-sampled to 1600 x 1200 pixel resolution, to balance overall data size with sufficient resolution to capture objects in detail. A planar laser range finder was tilted with a continuous periodic command to capture an entire 3D sweep of the scene from each viewpoint. The set of scans was then assembled to form a point cloud comprised of roughly 500,000 individual points. Each point is represented with a 3D position as well as an intensity value measured by the laser. The relative positions of the robot's sensors were calibrated as often as possible with the method previously described in Section 3.5.1. However, a moderate degree of calibration error remains a factor, as is the case for many commodity robotic platforms.

4.2.2 Geometric Registration

When a physical robot platform explores an environment, it typically has access to several forms of sensor feedback that can be used to determine its position. Also, it can actively control its position by issuing movement commands. In order to replicate this situation as closely as possible when performing recognition from our pre-recorded data, all information

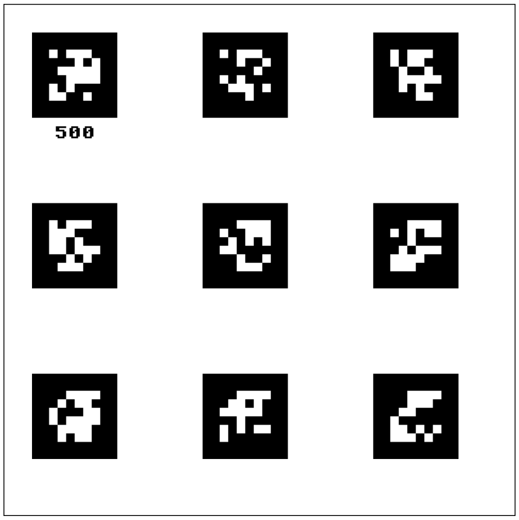


Figure 4.3: A Single Face of the Registration Target: A cube-shaped target, composed of 6 such faces, is used to register images contained in the UBC VRS dataset.

in the database was registered to a common coordinate frame. First, the set of camera poses was registered using automatically detected fiducial marker points that correspond to known 3D target geometry. These correspondences allow us to solve for the camera poses in a global frame. Then, pre-calibrated transformations that relate the camera to other sensors and the robot's base were applied to globally register all information types. Using this common registration, robot control can later be simulated, in combination with the real sensor data. This section will describe the process for registering the camera poses in detail.

The cube-shaped target displayed in our example images (e.g., Figures 4.3 and 4.4) is comprised of ARTag visual fiducial markers [Fia05]. Each square AR Tag maker encodes identity information in the form of an integer value represented by its internal white–black binary pattern. Detection of markers in realistic, challenging images (i.e., those with partial tag occlusion, many similar tags, and small scales) is achieved with robust image processing, and by using redundant encoding that makes checksums and error correction possible. The ARTag library provides a tag decoder that achieves virtually zero false positives and simultaneously localizes the corners of the fiducial patches in the image with sub-pixel accuracy.

As shown in Figure 4.3, we have constructed our registration cube such that each face contains 9 ARTag markers. We have selected a different set of integer identifiers for the squares on each cube face, so that by decoding the tags, the viewing pose of the camera with respect to the cube is uniquely identified. For example, the top-left square shown in Figure 4.3 has the identity 500, and the subsequent squares shown in the figure have identities ranging from 503 to 525. Consecutive identities are not chosen because these are slightly less easily distinguished by the ARTag decoding algorithm. We have manufactured the cube target from six printed faces, taking care to achieve precise 3D geometry such as right angles between faces and identical proportions and sizes for each face.

Each detected image location (i.e., tag corner) provides a 2D to 3D constraint on the extrinsic camera parameters (pose) using the typical pinhole camera projective equation

$$\alpha \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = K[R|t] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}, \tag{4.1}$$

where: x and y are the image coordinates of the detected corner pixel; K is the known intrinsic camera calibration containing the focal length, offsets and skew; R and t are the unknown rotation and translation of the camera at a particular pose, which we seek; X, Y, and Z are the 3D coordinates of the corner point using the known layout of the target; and α represents projective scale. Our target provides between 36 (the 4 corners of 9 squares on a single cube face) and 108 (all of the corners on 3 cube faces) visible corners, depending on the viewpoint. This yields a highly over-determined system. We estimate the solution using an approach similar to camera calibration methods such as [TL87], which involves making an initial guess using homography constraints that exploit the known planarity of the target's faces, and then by refining the estimate using the Levenberg Marquardt algorithm to minimize re-projection error.

We have validated this registration method on a number of test images by projecting known 3D points (e.g., a cube corner, or another point we have physically measured in 3D) into each of the images and manually observing the error in re-projection. The registration is typically accurate to within a pixel with the maximum error on the order of several pixels. Figure 4.4 illustrates the registration accuracy in a set of example images. Registration

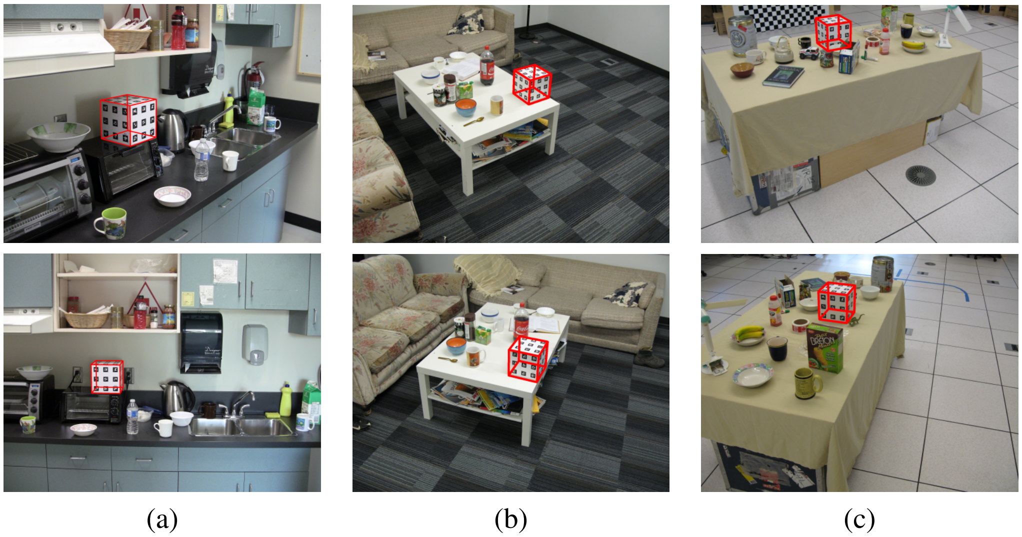


Figure 4.4: Sample Registered Scenes: Each column holds two views of the same scene. Our system uses the estimated camera positions relative to a global frame along with previous intrinsic calibration estimates to render a wire-frame of the extents of the calibration target (shown in red where colour is available) into each view. Accurate alignment of the wire-frame to image content in both views indicates accurate registration.

information is stored with the raw sensory data and both are used during annotation and simulation of robot motion for testing.

4.2.3 Object Annotation

In order to evaluate the performance of recognition algorithms, a human has annotated the ground truth object information in each scene and image in the dataset. We seek to describe objects both in 3D, in the common registration coordinate frame, as well as in 2D, in each image. Annotating this information is a time-consuming process, so we have leveraged the registration information described above to ease the manual burden. We provide the annotator with a software tool that allows selection of the 2D image points corresponding to a feature that is visible in each of a number of views, such as the object's centroid. The software triangulates a 3D point in the global frame that best explains all of the selected image points, and an object is instantiated in that location. Finally, a set of controls are provided to the user to fine tune the object's orientation and scale in 3D, and another set are available to refine an automatically initialized 2D bounding box for each object. Please see Figure 4.5 for visualizations of both the final 2D and 3D ground truth that we produce. We will continue by providing more detail on our annotation procedure.

As mentioned previously, each image in our dataset has been accurately registered into a common coordinate frame. This allows projection of 3D information into each image, and it also permits triangulating a set of image points that correspond to a single 3D point. The first step in our annotation process is for a human to mark a central and identifiable feature on an object in 3 or more images. We then solve for a 3D point that falls closest to the rays through each marked pixel. As described in Hartley *et al.* [HZ00], this can be represented as the solution to an equation of the form $AX = 0$. A is formed as

$$A \equiv \begin{bmatrix} x^1 P_3{}^1 - P_1^1 \\ y^1 P_3^1 - P_2^1 \\ \ldots \\ x^n P_3{}^n - P_1^n \\ y^n P_3^n - P_2^n \end{bmatrix}, \tag{4.2}$$

where each P^i for $i \in \{1, ..., n\}$, is the three row by four column projection matrix combining extrinsic and intrinsic parameters: $P \equiv K\,[R|t]$. Subscript notation indicates selecting a particular 1-indexed row of the matrix. We use superscripts to enumerate the image points and projection matrices for each of the n marked correspondences. The result of triangulation is an estimated 3D center point for the object. Our annotation tool instantiates a 3D object region composed of a 3D centroid initialized to the triangulated point, a 3D scale initialized to be the mean size of the object category, and an azimuth angle (i.e., rotation around the *up* or Z axis) initialized to zero. The annotator is then able to refine each dimension so that the projections of the oriented 3D bounding volume match the object's extents in each image. We have found that, if the image points are specified accurately at the outset, there is little extra effort required beyond specifying the true object orientation and making small refinements to the scale. Upon approving of all properties of the 3D annotation, the annotator saves the object volume and this is recorded along with the sensor data and registration information to be available for use of the data in training or test.

Image space 2D annotations are also available for all of objects in all images in the dataset. The 2D annotations share the format used by PASCAL VOC and other recognition challenges. That is, they are made up of a bounding box, with extents tight to the object content, a category label, and additional meta-information such as that the instance may be *difficult*, in that it is an uncommon representative of the class (e.g., a toy coffee mug in the shape of a cartoon character is a difficult mug), or that the instance is *occluded* in the image. For occluded objects, the annotator estimates the ratio of the object's area that is visible, and this is recorded in the dataset for later use during evaluation.

Creating 2D bounding boxes for more than ten thousand views of objects is also time consuming. To assist the annotator, the previously created 3D annotations are leveraged to expedite the process of creating 2D annotations. Volumes are projected into every image in which they are visible, and a bounding box that encompasses the 3D corners is created. The annotator's task is then simply to refine the precise image locations and meta-information values, rather than having to create each bounding box. This saves significant effort and reduces the probability that an image region will be missed due to human error.

At this stage, the annotator also often makes small adjustments to the 2D bounding box to ensure that it is *pixel-tight* to the underlying object content within the image. This hand-adjustment is needed because we project imprecise shape models (i.e., a box-shaped 3D volume, rather than the object's true 3D shape), and to account for any small errors introduced by 3D to 2D projection. Once again, when the annotator is satisfied with the quality of the data, the 2D box and meta-information are saved to the database.

The code and tools of our labeling pipeline can be re-used for any series of moderately well-registered images, such as video sequences from well-calibrated vehicles possessing accurate inertial positioning and a camera, or sets of highly overlapping photographs where structure-from-motion can be used for registration. We have made our software tools available and open-source to the community on-line along with the dataset[1].

4.2.4 Scene Details

We have described a procedure for collecting robot sensor readings of an environment with a robot platform, geometrically registering the sensor positions, manually annotating the object locations, and storing all of the information for future use. This procedure is only meaningful if the environments covered by the robot are sufficiently realistic and interesting

[1] The data and code are both available at http://www.cs.ubc.ca/labs/lci/vrs/index.html

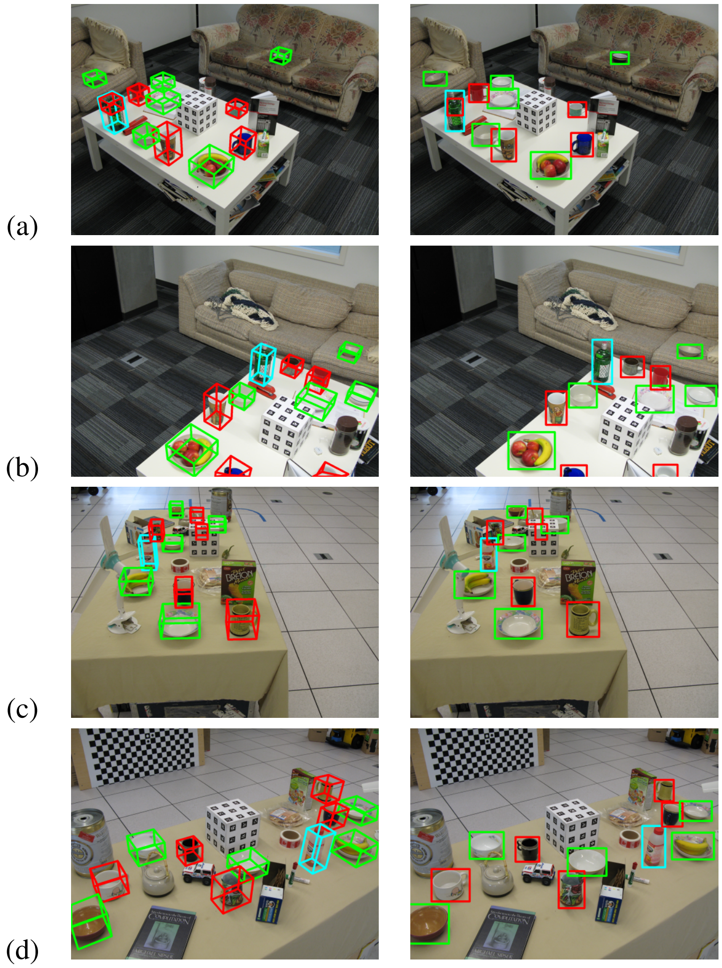


Figure 4.5: Sample Annotated Scenes: The left column shows 3D annotations projected onto the image (another verification of accurate registration) and the right column represents annotated 2D bounding boxes, which are manually verified and adjusted in each image. The first pair of rows, (a) and (b), are two views of the same scene, and the second pair of rows (c) and (d) are a second scene. The colours (where available) represent the object category with: bowls in green, mugs in red and bottles in blue.

to evaluate robotic object recognition. As the focus of our research is to produce capable home robots, the ideal data would originate from real physical home locations. Unfortunately, the Curious George platform is large and cannot easily be transported between a large number of homes. Therefore, we have made our best effort to replicate home-like scenarios within our lab setting. We have obtained a wide variety of real object instances from categories that occur in homes, laid out these instances to simulate the density and configuration that would be expected, and varied the supporting furniture and room configurations as appropriate. In order to minimize dataset bias, we ensured that exact object instances were not re-used between the training and test portions of the dataset. This does not preclude multiple similar instances of the same category from occurring in both subsets. This occurs within our data mainly for categories with little intra-category variation such as *bowls*, and when a large portion of the instances of a category are obtained from a single source in the real world, such as Ikea for kitchenware in North America.

As many modern recognition approaches require the use of labeled data to construct appearance models or to set parameters, we have split the scenes into training, validation and test sets. We have followed the same practices for selecting realistic objects across the three sets, and each consists of real physical scenes. However, the scene backgrounds available to us did vary somewhat in their similarity to real homes, and we have placed the most realistic environments largely into the test set. This makes the problem of classifying objects in the test set as hard as possible and ensures that final results of the methods can most accurately represent expected performance in the real world. Some training set scenes have backgrounds that could occur in homes, but in other cases, a plain table cloth has been used with little additional clutter and few distracting objects.

4.3 Evaluation Protocol

This section describes a set of protocols that leverage the information provided in the dataset to simulate the tasks that face robot recognition platforms and to evaluate the performance of algorithms on these tasks. The flow of data for this general process is illustrated in Figure 4.6. Specific robot recognition scenarios can be replicated by varying the nature of the data that is exposed to the object recognizer. In each case, data is provided through a simulation tool with an interface similar to those present on typical robot platforms, following the *simulate-from-real-data* protocol. The performance of recognizers is evaluated by comparison against the annotation information produced by our human labelers, using the well-accepted techniques that were described in Section 2.6.

The UBC VRS data and simulator allow for study of each of the robotic recognition scenarios that were previously described in Figure 1.3. Here we will briefly describe how

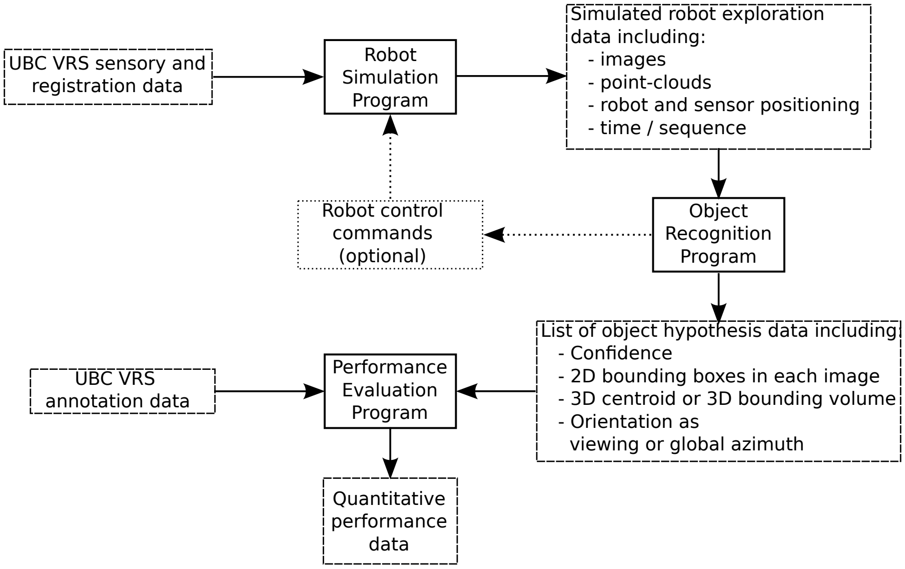

UBC VRS sensory and registration data
Robot Simulation Program
Simulated robot exploration data including:
- images
- point-clouds
- robot and sensor positioning
- time / sequence
Robot control commands (optional)
Object Recognition Program
List of object hypothesis data including
- Confidence
- 2D bounding boxes in each image
- 3D centroid or 3D bounding volume
- Orientation as viewing or global azimuth
UBC VRS annotation data
Performance Evaluation Program
Quantitative performance data

Figure 4.6: The *simulate-with-real-data* Protocol: A flow chart indicating the data and software programs involved in assessing the ability of a proposed object recognition algorithm to complete the task of localizing objects in the UBC VRS benchmark.

each of these tasks can be accomplished with our data:

- **Single View Recognition –** can be achieved by simply providing an index file into each of the raw images that make up our dataset. In this case, a robot is not truly being simulated, so no special software is required. Although this problem instance does not capture the robotic aspects of the recognition task, it can be a useful measure of the strength of an image-based recognizer and we will use such results throughout this thesis as baselines for more integrated methods.

- **Passive Multiple View Recognition –** requires our software simulator to query coherent sets of data from the same physical scene and to link the data from each location with the common registration or path information that records how the robot moved during data collection. To model a passive approach (i.e., one that does not send control inputs), the sub-set of robot poses for each scene is chosen by the simulator from the full set of possibilities that were visited during data collection. Several modes are available for this selection including: selection of sequential poses along the robot's real path; random selection of poses that disregards ordering; or selection of poses that achieve a desired angular spacing. The data from all poses can be provided to the recognition method in a single shot for batch inference, or one pose at a time can be processed to support filtering-type methods.

- **Active Multiple View Recognition –** involves a recognition method providing control feedback to the simulator based on the data that has been examined so far. In this context, the simulator only yields data from one view at a time. It then models a robot motion that corresponds to the given control input as closely as possible. Our software can only sample from poses that actually occurred during data collection, so there is a discretization of the requested control inputs (i.e., the simulator returns the view that best matches the given control). This is the most realistic representation possible of the real robot exploration experience, given the data that we have collected.

A set of standard tools for results evaluation is available as part of the dataset software. These tools implement the community-accepted performance evaluation metrics that were discussed in Section 2.6, based on the annotation data within our dataset. Several subsequent chapters will rely on this analysis.

4.4 Discussion

While careful collection and annotation of a dataset with sufficient scale for meaningful evaluation is a large effort, the resulting repeatable evaluation will hopefully be of value to the robotics community. Beyond data, a key contribution of our method is the labeling and evaluation pipeline. The tools related to these can extend to a variety of additional data sources. For example, we have already succeeded in using the same tools to annotate and evaluate our approaches using Kinect data that was registered without the use of our fiducial marker (i.e., using the software of Endres *et al.* [EHE$^+$12]) as well as outdoor data collected by an automobile with a highly accurate inertial measurement unit [PME11]. A number of authors have already obtained the UBC VRS and are currently beginning to develop new solutions. Ideally, this will lead to additional performance improvements being published in coming years by a variety of authors.

4.5 Chapter Summary

This chapter has described the collection of the UBC VRS dataset as well as the associated tools and protocols that allow repeatable evaluation of robot recognition methods using our data. We outlined the *simulate-from-real-data* framework, which allows realistic sensory characteristics, while still providing the ability to evaluate effects of information from multiple viewpoints and active control on recognition performance. This chapter has not described any solutions for recognizing the objects that are present in our data. This will occur primarily in Chapter 7, where the UBC VRS dataset is used as the primary quantitative evaluation. Several authors have also previously used the UBC VRS or early sub-sets for evaluation of published recognition approaches. At the time that this thesis was submitted [HMM$^+$10, ML11, MWSL11] have used our VRS dataset for evaluation, and several additional methods are likely to be released in the near future. The remainder of this thesis will describe the algorithmic and technical contributions, which includes a number of approaches for multiple viewpoint object recognition and active control of a robot platform that is performing recognition.

Chapter 5

A Multiple Viewpoint 3D Object Model

5.1 Introduction

The previous chapters of this thesis described the robotic object recognition problem, our physical platform and the data collection efforts that support the research in this thesis. The remainder of this thesis will describe our algorithmic contributions, including a viewpoint planner and robotic recognition techniques. There will be little further discussion of hardware platforms, specific sensors, visual attention, or robot navigation. The experiments will be conducted in a repeatable fashion on publicly available data (e.g., the UBC VRS set or others), rather than with trials of live robots. Motivated by embodied recognition, we continue to consider the active and passive multiple viewpoint recognition problems that were described in the previous chapter, and we choose datasets that are close to those facing real platforms.

Our methods combine 3D reasoning with visual appearance modeling for object categories. A common theme is that the 3D positions of objects are inferred based largely upon 2D cues in sensor data. The missing depth dimension can be difficult to recover even with available range sensing, since realistically cluttered environments contain structure at multiple depths within most image regions. Our methods locate corresponding *semantic content* across multiple images to robustly recover this depth information. However, occlusion and clutter also complicate the process of identifying objects within a single image and determining correspondences between images. We address all of these issues with a robust probabilistic relation between objects, visual evidence and 3D physical constraints.

We express the relation as a Bayesian probabilistic model, following recent trends in automated perception. Each source of information from each available viewpoint is explained by an independent generative model, potentially containing parameters and latent variables. This allows powerful inference tools to be employed and enables us to learn model parameters from training data. The expected visual appearance of each object category is related to object instances in three dimensions using the *tracking-by-detection* framework. This means that we explain the intermediate outputs of an appearance model and do not examine the pixels of the image directly. Where available, sensed range information is used to reason about 3D locations and object occlusions.

A common model formulation has been used for several distinct contributions, as our ideas and implementation evolved over the second half of the author's PhD studies. This fact is reflected by the structure of the next several chapters. In this chapter, we introduce the common underlying material by describing the form of our probabilistic model and by outlining several options for each component. However, we will not completely instantiate the model here or present results for any specific task. The following three chapters will then make the form of each model component explicit, as it has been used for specific tasks and has evolved in our implementation over time. These chapters will contain experimental evaluations to demonstrate the usefulness of each approach. Chapter 6 will describe the use of viewpoint-aware object models to plan paths so that the robot's action assists in distinguishing an object correctly. Chapter 7 will describe a model suitable for overcoming occlusion caused by cluttered scenes by modeling the visibility of object parts. Finally, Chapter 8 will introduce the scene understanding problem and our solution, which is based on even more detailed 3D object-part models.

5.2 Model Components

We express the robotic object recognition task probabilistically, as $P(\mathbf{X}|\mathbf{E})$: the probability of a state of the world, $\mathbf{X}$, given the observed evidence, $\mathbf{E}$. We seek to recover the world state as a set of objects, each with a semantic category label and a spatial description (e.g., 3D shape and pose). The evidence available to modern robots often includes visual imagery from on-board cameras, sensed range information (i.e., from a tilting laser range finder or depth camera), and geometric knowledge such as calibration of system components or localization information about the robot's position. We continue by formally defining both objects and evidence.

5.2.1 Objects

We represent a set of objects symbolically as: $\mathbf{X} = \{O_1, ..., O_i, ...\}$. For each object, O_i, the semantic category label is represented as O_i^c. We will often employ an *oriented bounding volume* model for three dimensional spatial information, or pose, of an object, which we represent as the centre position, $O_i^{cent} = \left[O_i^x, O_i^y, O_i^z\right]^T$, scale (composed of length, width and height), $O_i^S = \left[O_i^l, O_i^w, O_i^h\right]^T$, and orientation (composed of roll, pitch and yaw), $O_i^{ori} = \left[O_i^{roll}, O_i^{pitch}, O_i^{yaw}\right]^T$.

Note that spatial information is not indexed by time. Throughout this thesis, we make the static world assumption, which implies that we solve localization and not tracking. While many of the methods we describe can be extended to moving objects, that has been left for future work.

5.2.2 Evidence

The evidence available to a robot can be modeled as a stream of observations gathered over time and space, as the platform moves: $\mathbf{E} = \{Z_1, ..., Z_t, ...\}$. We study the case where a visual image, I_t, and optionally a synchronized range image or point cloud, C_t, are available, at a number of discrete instants in time t. The position of the platform at each time is also assumed known throughout our work. This registration information, which can be obtained from structure-from-motion or mapping methods, is expressed as a coordinate transformation matrix, ${}^W_V T_t$, which expresses the pose of the vehicle in the world frame. Figure 5.1 illustrates the geometry available to our system. The observed data from each view is $Z_t = \{I_t, C_t, {}^W_V T_t\}$. The sequence of observations from the beginning of time until the present is represented as: $Z^t = \{Z_1, ..., Z_t\}$.

Evidence Geometry

We seek three dimensional object information relative to a stationary, global frame of reference, which we will refer to as the world frame. For some tasks, the origin of the world frame will coincide with a physical element on the robot at a particular time, such as its

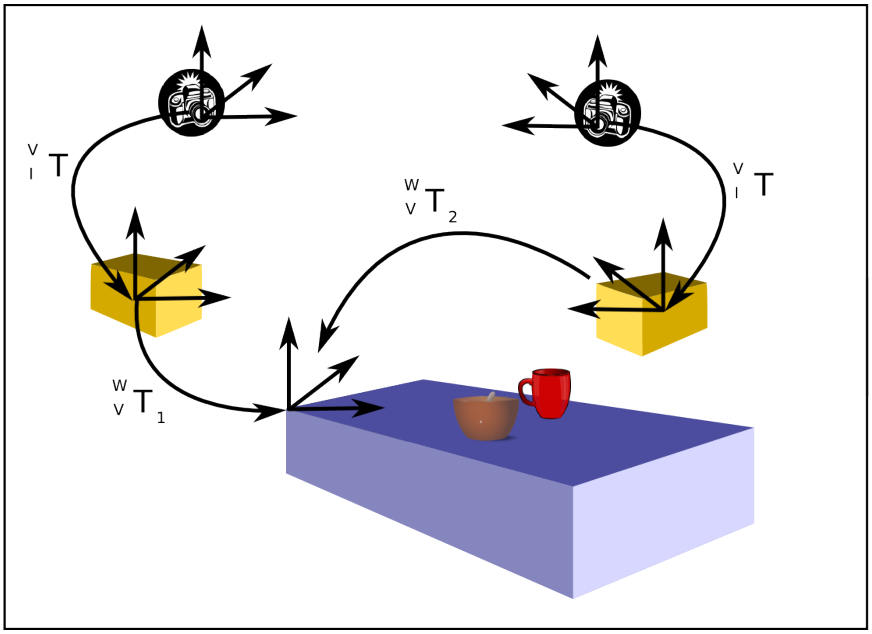


Figure 5.1: Multiple Viewpoint Geometry: Selected coordinate transformations available to our object recognition system. The figure shows mapping information that relates the robot's position to the world frame at two distinct times: ${}^{W}_{V}T_1$ and ${}^{W}_{V}T_2$. Also, the calibration information relating the camera's position to the base of the robot (vehicle) is shown as ${}^{V}_{I}T$. By composing transformations, sensory data can be related to object locations.

pose at the beginning of the trajectory considered. Elsewhere, we will place the origin on a landmark in the environment, such as a corner of the cube registration target described in Chapter 4. In all cases, we will clearly describe how the world frame has been defined.

Since the platform is moving, the relationship between the sensory information and the fixed world frame changes over time. Pre-calibration of the sensor positions gives access to rigid-body transformations between the base frame of the vehicle and each sensor: ${}^{V}_{I}T$ for the image from a camera and ${}^{V}_{C}T$ for the point cloud from a depth sensor if available. Note the lack of a time subscript indicates that the calibration information is static and not related to the platform's motion. We consider calibration information a part of the system parameters, and therefore do not include it within the list of observed evidence at each time.

Sensed point clouds are denoted as C_t. The 3D information contained within each cloud is initially expressed relative to the range sensor's local coordinate frame. Mapping and calibration information allows each point cloud to be expressed within the world frame, and denoted ${}^{W}C_t$, by composing transformations as

$$\begin{aligned} {}^{W}C_t &= {}^{W}_{C}T_t C_t \qquad (5.1) \\ &= {}^{W}_{V}T_t {}^{V}_{C}T C_t. \qquad (5.2) \end{aligned}$$

Cameras are pre-calibrated, so the the intrinsic parameters, K, are known. In this case, the relationship between sensed images and the world is through projection into the image as

$$\begin{aligned} \alpha \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} &= {}^{I}_{W}P_t \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \qquad (5.3) \\ \alpha \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} &= K[{}^{I}_{W}R|{}^{I}_{W}t] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}, \qquad (5.4) \end{aligned}$$

where P represents a projection matrix and x and y are pixel coordinates within an image. A three dimensional point is represented as X, Y and Z. ${}^{I}_{W}R$ represents the rotation sub-matrix (top three rows and top three columns of ${}^{I}_{W}T_t$) and ${}^{I}_{W}t$ represents the translation vector (rightmost row, top three columns of ${}^{I}_{W}T_t$). ${}^{I}_{W}T_t$ is obtained by reversing the directions and composing mapping and calibration information as

$$\begin{aligned} {}^{I}_{W}T_t &= {}^{W}_{I}T_t^{-1} \qquad (5.5) \\ &= ({}^{W}_{V}T_t {}^{V}_{I}T)^{-1} \qquad (5.6) \\ &= {}^{V}_{I}T^{-1} {}^{W}_{V}T_t^{-1}. \qquad (5.7) \end{aligned}$$

We have now defined all of the symbols needed to express the quantities of interest in the multiple viewpoint robotic recognition problem. The next section will continue by expressing the probabilistic model that relates objects to evidence.

5.3 Probabilistic Model

The likelihood of an object given the available data is expressed using Bayes rule and by making the naive Bayes assumption to achieve independence between viewpoints. That is

$$
\begin{aligned}
p(\mathbf{X}|\mathbf{E}) &\propto p(\mathbf{X})p(\mathbf{E}|\mathbf{X}) && (5.8)\\
&\approx \prod_t p(\mathbf{X})p(Z_t|\mathbf{X}) && (5.9)\\
&= p(\mathbf{X})\prod_t p(I_t, C_t, {}^W_V T_t|\mathbf{X}) && (5.10)\\
&= \underbrace{p(\mathbf{X})}_{object\ prior} \prod_t \underbrace{p(I_t|C_t, {}^W_V T_t, \mathbf{X})}_{appearance} \underbrace{p(C_t|{}^W_V T_t, \mathbf{X})}_{geometry} \underbrace{p({}^W_V T_t|\mathbf{X})}_{registration}. && (5.11)
\end{aligned}
$$

The above equations demonstrate how the likelihood of a set of objects can be explained with four sub-models. We will continue by explaining each in detail.

5.3.1 Object Prior

The object prior, $p(\mathbf{X})$, represents our knowledge of the properties and layout of objects, independent of any source of sensor data. Subsequent chapters will describe how this term encodes penalties on objects occupying the same physical space, the expectation that objects lie on a supporting plane within the environment and have appropriate scale, and our expectation on the relative frequencies of various objects occurring.

5.3.2 Appearance Likelihood

The appearance likelihood term, $p(I_t|C_t, {}^W_V T_t, \mathbf{X})$, explains the visual image evidence given the objects present, the sensed geometry and the camera's pose when it obtained the image. Throughout our work, we relate objects to image evidence using the proxy of trained object appearance models, inspired by the so-called *tracking-by-detection* framework that was described previously. Rather than including raw pixel values in our probabilistic model, we initially apply an object detector (specific choices to be explained in subsequent chapters),

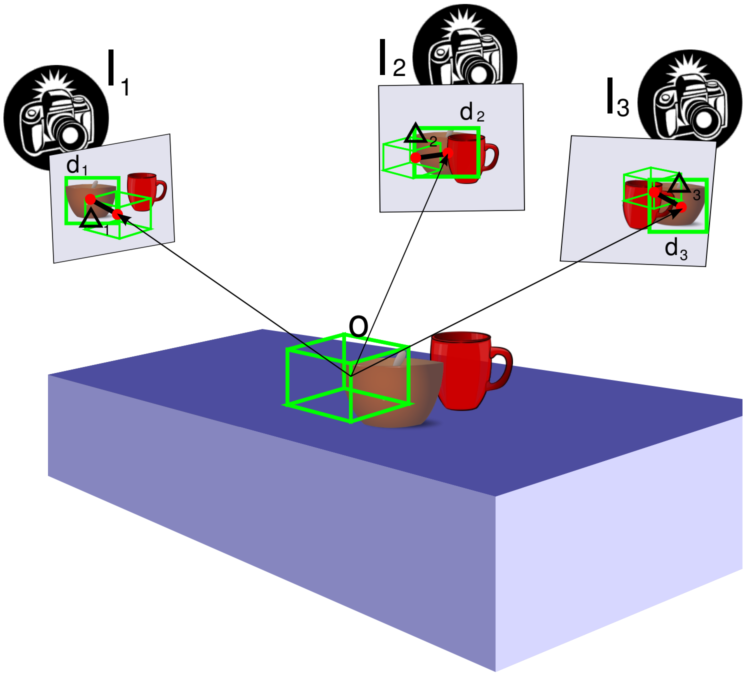


Figure 5.2: Object-Model Cartoon: Using the *tracking-by-detection* framework, each inferred 3D object is projected into many views and related to image space detections, d_i, that are generated by an appearance model. Alignment errors Δ_i reflect the difference between the expected image position and scale and that generated by the 2D recognizer. The likelihood of each object hypothesis is expressed based on the 3D geometry, the score of matching detections modulated by occlusion and the error in projection.

to produce a set of detections in the image $D(I_t) = \{d_1^t, ..., d_j^t, ...\}$, where d_j^t is the j^{th} 2D object hypothesis output by the recognizer in the image at time t. A single detection, d_j^t carries a variety of information, depending on the outputs of the detection model. In all cases, it includes a floating point confidence score and a 2D *bounding box* around the object, which we represent as a centre-point and a scale.

The appearance likelihood, therefore, explains the output of an object detector as

$$p(I_t | C_t, {}^W_V T_t, \mathbf{X}) \quad \approx \quad p(D(I_t) | C_t, {}^W_V T_t, \mathbf{X}). \tag{5.12}$$

Numerous factors affect the appearance likelihood including the properties of the object recognizer that produces the set of detections, D, physical properties such as occlusion that can be treated as latent variables, and geometric properties of the object category such as whether or not it is symmetric. We will continue by briefly discussing a number of these factors within this chapter, in particular highlighting those that will be featured in more detail in subsequent chapters.

Object Recognizers, Training and Validation

For the purposes of this thesis, we treat an image space object recognizer as a gray-box function that takes a visual image as input, and produces a set of scored bounding box detections as output. A perfect recognizer would produce detections with full confidence only on the true locations of objects within an image and no detections elsewhere. In this case, the task for robotic recognition would simply be to recover the three dimensional object locations that explained these image space detections geometrically. However, currently no visual object recognizer achieves perfect performance on the visual images that we consider. Our appearance likelihood model allows us to gracefully handle errors in visual recognition and to accumulate evidence across viewpoints as a component of Equation (5.11).

Current state-of-the-art methods are based upon learning visual appearance models from numerous labeled training examples. Throughout the remainder of this thesis, we will use models that have been carefully trained to perform as well as possible on the object categories and imaging conditions present in our evaluation environment. The appearance likelihood function, which relates detections to objects, is obtained after the model training process, through a validation procedure using a held-out set of labeled data. That is, the gray-box recognizer is first trained so that it performs as well as possible without knowledge of 3D properties, and then its performance is described probabilistically with a model of the form given in Equation (5.12). One example can be found in Section 6.2.1 which describes an approach to estimate the detector's non-uniform response to the different viewpoints of an object.

Data Association

Objects in three dimensions and detections in image space are both discrete quantities, and each represents a different type of spatial information. We perform data association in order to reason about them jointly. Specifically, each object is projected into each available view, and a matching function attempts to locate a corresponding detection, using proximity. We express data association symbolically as $A(O_i, {}^{W}_{V}T_t, D(I_t))_t = d^t_j$, which tells us that the three dimensional object, O_i, has generated detection d^t_j, a detection in image t.

The association function, $A(.)_t$, is a matching between objects in 3D and detections in 2D image space. It is not trivial to find the correct matching in the general case. In cluttered scenes, objects can mutually overlap and occlude one another in views. Some true target objects may lack image space evidence: a false negative error. In other cases, detections will occur in locations where no object exists: a false positive. Multiple detections with slightly incorrect geometry may be present in an area with only a single true object: a multiple detection. Therefore, numerous potential matchings are worthy of consideration by a search procedure. However, the data association step often occurs as the inner loop of a learning or inference procedure. So, if association is an expensive operation, the entire object recognition method is likely to be inefficient.

A simple heuristic matching function can perform well in practice, even without guarantees of optimality. Consider one of the simplest reasonable possible choices, which we call *Greedy Matching*, that is described in Algorithm 1. The output of *Greedy Matching* depends on the ordering of objects within O, and it does not guarantee the global optimality of the assignments. However, it is efficient to compute. It requires only a single pass through the objects for each image, and the $NearestUnclaimedNeighbour$ sub-step can be implemented with a k-d tree index on detections to provide $O(log(n))$ over n detections in the image, if desired. The majority of our approaches and results have been produced using *Greedy Matching* or a slight variation, since this has provided suitable performance on real-world tasks. Consideration of other association functions is an avenue for future work.

Object Viewpoint

Some object recognizers perform pose estimation. The appearance of objects varies as they are viewed from different angles, and a pose estimator attempts to recover the unknown viewpoint based on the observed appearance. In principle, pose-estimation can be performed in three dimensions, but in all cases that follow, our input pose estimators simply estimate the azimuth viewing angle, which represents rotation about the object's canonical up direction. Our method utilizes inferred pose information from the recognizer during 3D object inference, during both data association and likelihood computation. During data

Algorithm 1: Data Association Algorithm: The *Greedy Matching* data association function

```
Data: $O$, a set of objects in three dimensions, $D(I_t)$, a set of image space detections,
      and ${}^W_V T_t$, the mapping information at time $t$
Result: An association $A$ from each object-index $i \in \{1, ...n\}$ to detection-indices $j$
        of the assigned match, or $\emptyset$ if no detection matches
Set all $d_j \in D$ unclaimed ;
for $i \leftarrow 1$ to $n$ do
    $A(i) \leftarrow \emptyset$;
    projObj$_{it}$ $\leftarrow$ project$(O_i, {}^W_V T_t)$;
    $j \leftarrow NearestUnclaimedNeighbour(D,$projObj$_{it})$;
    if $d_j$ agrees with the position and scale of projObj$_{it}$ within a threshold then
        $A(i) \leftarrow j$;
        Set $d_j$ as claimed;
```

association, the matching function can discard some very unlikely pose assignments. For example, if the three dimensional object's pose indicates a side-view, we may not allow detections that indicate a front-facing object to be assigned. As with all of our model components, this step must be robust to errors in the pose estimator. We have considered a number of alternative approaches in each instance and choose the one that gives the best empirical performance.

Chapter 6 will describe how viewpoint-aware object recognizers can allow informative viewpoint planning for a platform while it performs recognition, and Chapter 8 will show that 3D object pose can be inferred from 2D viewpoint estimates and that this information can increase the detail with which our method can understand the objects within a scene.

Occlusion

Occlusion changes the visual appearance of the object in an image by obscuring many of the features that we use to describe an object. Object recognizers can overcome occlusion to a certain extent through robust modeling. In particular, if occluded instances are provided during training, there is some potential for the recognizer to have learned the typical patterns of appearance under occlusion, and to correctly identify both occluded and un-occluded instances. However, in practice, occlusion is one of the most prominent failure modes for the object recognizers that perform well at the time of this thesis (as is empirically verified by [HCD12]). When 3D objects appear occluded in images, there will often be no detection provided by the recognizer, or the detection will have low confidence, which typically indicates a low likelihood of an object actually being present.

A focus of Chapter 7 is to examine the effects of occlusion on the appearance of objects within images, to learn a model for the performance of a particular recognizer under occlusion, and to exploit the three dimensional nature of our object models along with sensed range data to improve our ability to perform the multiple viewpoint recognition task under heavy occlusion.

Object Parts

Recent visual recognition approaches have often included a hierarchical model for the appearance of objects. A collection of parts describe distinguishable components of the object and these are spatially-related to other parts or to the object as a whole (e.g., [FGMR10] and [TFL$^+$09]). Starting with Chapter 7, our work is able to leverage the part information from an image space detector. We reason about part-detections alongside the whole-object detections. Also, we have extended this part reasoning to our 3D object models, reasoning about an object as a collection of parts that are have spatial relations to the whole object. We will discuss our three dimensional parts models in Chapter 8.

5.3.3 Geometry Likelihood

The third term in Equation (5.11), $p(C_t|{}^W_V T_t, \mathbf{X})$, relates sensed range evidence, C_t, to the set of objects in the world and the position of the vehicle. As mentioned previously, we do not consider detailed shape models in 3D, and instead make only generic shape assumptions based on the *oriented bounding volume* model described above. This model allows sensed range values to be roughly related to the visible surface of each object. An important part of our geometry modeling is to reason about a number of distinct outcomes for each element of range data, including: (1) the object's pose agrees well with the measurement since its front surface is at the sensed depth; (2) the sensed depth is in front of the object, meaning either a pose error has occurred or the object is occluded; or (3) the sensed depth is behind the object, which could indicate transparency (e.g., the window of a car) or a pose error. We have used this approximate surface model to reason about likely positions for objects as well as the likely occlusion patterns in each view in Chapters 7 and 8.

5.3.4 Registration Likelihood

The final term in (5.11), $p({}^W_V T_t|\mathbf{X})$, represents the likelihood of a view's registration information. While every existing registration and mapping procedure is inexact and produces position estimates that are corrupted by slight errors, we have neglected these errors throughout our work. This choice has been made because the expected magnitude of registration errors using modern techniques is quite small compared to the uncertainty in object localization from images, at least in the environments we consider. So the loss in precision by neglecting this term is also assumed small. We leave joint modeling of mapping and object localization for future work, and briefly describe these extensions in Chapter 9.

We express this neglect of the localization term probabilistically with the Dirac delta function centered on the registration estimate provided by an external mapping algorithm, ${}^W_V T_t^{mapping}$. This is expressed as

$$p({}^W_V T_t|\mathbf{X}) = \delta({}^W_V T_t - {}^W_V T_t^{mapping}). \tag{5.13}$$

This function is only non-zero when the registration precisely agrees with the observed value, making it unnecessary to consider any other value. Throughout the remainder of our discussion, we remove this term from our expressions.

5.4 Parameters and Learning

Each of the generative sub-models that we have described has the potential to include a set of learned parameters. We will use θ as a generic term to describe all system parameters, and use subscripts when we refer to specific parameters. For example, $\theta_{viewpoint}$ describes the parameters used for a viewpoint recognizer, which would be expressed as $p(O_i|Z_t, \theta_{viewpoint})$. Validation data subsets are used for evaluating the behaviour of a learned model after it is produced on a training set. The separate fold of validation data allows the *post-hoc* learning of parameters for a trained model without the bias that would be introduced if the training dataset itself was used. Simply put, a recognizers performance on its own training set is not a good measure of real-world performance, but a previously unseen validation set is likely to uncover the features that can be expected on arbitrary test sets. Subsequent chapters will describe the parameters needed for each model component.

5.5 Inference

The probabilistic model that we have described above is defined in a state space with a potentially large number of continuous dimensions. Probabilistic inference is required in order to locate the most likely single object or set of objects within the state space – the task of object recognition. Depending on the specific forms of the appearance and geometry likelihoods, this inference problem often lacks analytical closed form, but approximation methods are able to yield good practical performance. We will briefly discuss some of the generic inference concerns here. Particular inference techniques will be described in later chapters.

It is instructive to consider the form of the inference problem and several potential solution techniques that can be applied. Our likelihood has a discrete-continuous nature, which is introduced due to the *tracking-by-detection* framework and related data-association procedure that we have described in Section 5.3.2. First, consider directly maximizing the likelihood in the object state space. We note that the target likelihood is discontinuous, due to data association with discrete detection evidence. However, in local areas of the space that share a fixed data association, continuous optimization may be possible, depending on the form of the appearance and geometry likelihoods. Intuitively, this step involves finding the 3D object position that best explains a set of detections across images, as well as the sensed range data, and can be thought of as a semantic triangulation. We have applied such local optimization (e.g., gradient descent) as a solution for this sub-step in our inference techniques, in some cases.

We must also consider searching the space globally for subsets of detections that explain the same 3D object. Intuitively, this process involves discarding false positive detections in image space and choosing the correct single detection for an object when the recognizer returns several nearby detections. This combinatorial optimization problem has a hypothesis space that grows as $O((|O||D|)^n)$, where there are $|O|$ objects in the environment and our recognizer returns $|D|$ detections, in each of n available images. For our problems, these values are bounded by $|O| < 20$, $|D| < 100$ and $n < 10$, and this is far too large for consideration by a brute force approach, especially since scoring each possibility requires a continuous optimization over the local 3D object pose. Efficient search is crucial. For example, we use approximate techniques such as sampling to prioritize the most promising sets of associated detections. Many pairs of detections can reasonably be discarded quickly by a rapid heuristic check for compatibility. This allows reasonable approximate inference to proceed and to give good performance in practice.

5.6 Chapter Summary

This chapter begins our discussions of models and algorithms. We have presented a generic probabilistic model that relates objects in the world to perceived sensory information. Each component of the model was discussed, and for each we outlined a variety of factors and challenges that will face the specific methods that come in the following chapters. Inference and learning were also introduced, which are the tools needed to apply the generic model to real data and to perform robotic recognition. The following chapters will describe particular methods for these tasks by instantiating the generic model in a slightly different form, and by re-visiting many of the issues we raise here. Each of these will provide a complete discussion of inference, and will include empirical results to validate the approaches.

For purposes of continuity, here we list the general model components that must be specifically defined to solve a robotic recognition task in a specific context or for a specific problem. We will revisit this list a number of times to compare and contrast between our approaches.

1. *State space*: one or more semantically meaningful objects. The geometry of each object is described by a subset of the 3D shape properties that we have mentioned here.

2. *Evidence*: images and/or point clouds from a number of viewpoints.

3. *Object Prior*: a distribution for the expectation of object positions, occurrence frequencies, or layouts, independent of observed sensory data.

4. *Appearance Likelihood*: a distribution relating detection results as a proxy for visual appearance, to 3D objects.

5. *Geometry Likelihood*: a distribution relating sensed range data to the surfaces of oriented bounding boxes that represent objects in the world.

6. *Learning Method*: a procedure to set the parameters of the above models based on training or validation data.

7. *Inference Method*: a procedure to output a distribution or point estimate of the objects present in the world based on observations.

The next chapter will describe the use of a simple instantiation of these seven elements in order to derive a robot control algorithm that guides a robot towards viewpoints that are likely to lead to confident object hypotheses. This will be followed by two chapters that describe methods for 3D object localization in challenging environments with significant occlusion and clutter. In each, we will refer back to the seven elements listed here and describe how the specific choices made later fit into the generic Bayesian multiple viewpoint model of this chapter.

Chapter 6

Viewpoint Planning

6.1 Introduction

When uncertain about the identity of an item, humans often pick the object up and rotate it or move their head from side to side in order to obtain a variety of viewpoints. In some cases this behavior allows a "canonical" viewpoint of the object to be obtained (e.g., the label on a bottle) and in other cases, the movement may allow disambiguation between similar items (e.g., searching for the logo to identify the brand of car being viewed). Humans integrate information over the numerous viewpoints that they see without effort and can rapidly decide where to move next. In contrast, the analogous scenario remains a challenge for current visually guided mobile robots.

This chapter describes a method for viewpoint planning during the Active Multi-View recognition task. Our approach is based upon learning *viewpoint-aware* detection models from training data that is annotated with viewpoint information. After a number of viewpoints have already been analyzed, we use these models to plan the movements that are most likely to result in additional useful information being obtained. An information-theoretic next-best-view planner is proposed, and its performance is demonstrated on an existing database of simple multi-viewpoint imagery that is available from the computer vision community. Simulation results verify that our viewpoint planning approach requires fewer viewpoints for confident recognition than would be needed if the robot moved randomly.

We will not emphasize pose estimation in this chapter, although we will propose a

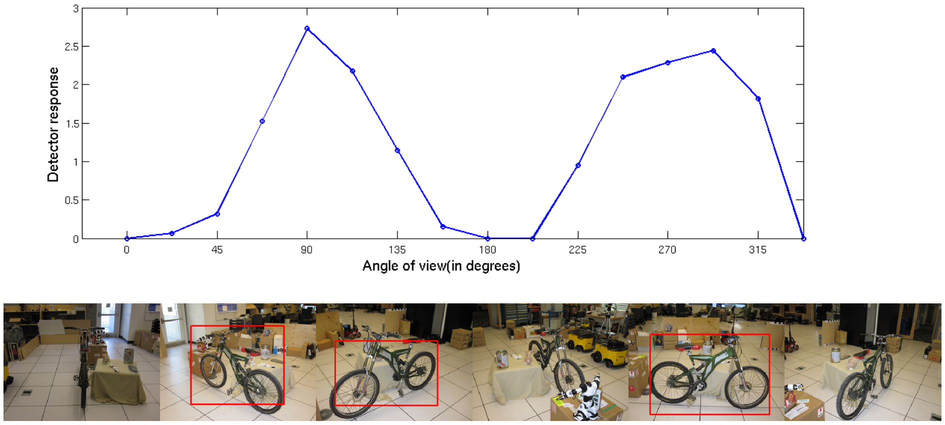


Figure 6.1: Viewpoint-dependence Example: The response of the DPM recognizer from [FGMR10] on images of a bicycle from numerous viewpoints. Images shown below align with data points, and bounding boxes drawn in images represent detector responses that exceed a threshold pre-calibrated to balance precision and recall.

method that models some aspects of an object's pose. Our focus in this chapter is on the viewpoint planning process and we have simplified the evaluation and discussion appropriately. A more detailed approach for pose estimation will be presented later in Chapter 8, which is capable of estimating the orientation of a number of self-similar objects in a crowded scene. At the time of the study performed in this chapter, models suitable for modeling the pose variation across instances of an object category were inexact in nature. Several methods included [SSFFS09, TFL$^+$09, LSS08]. Our viewpoint-dependent models of detector response can be seen as a soft form of pose estimation and are inspired by the approaches listed.

6.1.1 Relation to Generic Multiple Viewpoint Object Model

The planner described in this chapter is based upon integration of information over viewpoints using a procedure that is similar to the one described in Chapter 5. Here we will describe detailed model components that instantiate the generic Bayesian Multiple Viewpoint object model. Our focus here is the planning problem, and we show how the model can be used to determine the next best view from which to observe an object. Briefly, this involves analysis of the expected change in information contained in the model over a number of candidate actions. We describe an automated method to perform this reasoning.

Our method is centered on learning models of a category detector's response with respect to viewing direction using training data from a multi-view category database. For example, Figure 6.1 shows the detection responses of a state-of-the-art category recognizer on a number of views of a single bicycle – a motivation for the work in this chapter. Certain views of the bicycle give stronger support for the presence of the object than others, and a planner would do well to find the actions that lead to the side views being observed in order to maximize the confidence of the recognizer. We capture this intuition in the form of viewpoint-aware appearance models with parameters that are learned from training data. These models summarize the responses of a detector across numerous instances to capture its dependence on viewpoint, and are the instantiation of the generic appearance model component of Equation (5.11) within this chapter. Informative viewpoints can be chosen based on the current probabilistic estimate of the object and the learned viewpoint model, which allows an active system to recognize an object with fewer views.

For consistency, we will briefly preview in point form how the approach in this chapter instantiates all model components described in the previous chapter:

1. *State space*: binary category label (object of type c is present or absent) and discrete viewpoint (one of 8 uniform poses) for one object at a time. Localization is ignored here. Very simple version of the recognition problem that allows focus on the planning task.

2. *Evidence*: images from a discrete number of viewpoints that have been captured ahead of time to form a multi-view image dataset. Data is accessed by our system through a robot simulator.

3. *Object Prior*: uniform distribution over the object's presence or absence and over all potential poses. The maximally un-informed prior for our state.

4. *Appearance Likelihood*: learned viewpoint-aware detector models which allow computation of the expected responses given an observation of an object from a particular angle.

5. *Geometry Likelihood*: none here because no range data is present in the evidence.

6. *Learning Method*: two steps: 1) learn viewpoint-agnostic object models from a large set of images with labeled object regions that do not possess viewpoint labels and 2) use images from a held-out validation set that are labeled with viewpoints to derive viewpoint-aware models.

7. *Inference Method*: sequential Bayesian updates that integrate information one view at a time to allow planning of new poses.

6.1.2 Statement of Collaboration

The method described in this chapter was developed in collaboration with Ankur Gupta, a co-author of [MGL10]. At the time that this work was carried out, Ankur was a junior student starting in the UBC Laboratory for Computational Intelligence. The paper's concept, formulation and development were developed by the author of this thesis, but Ankur provided significant assistance with coding the tools, especially related to automatically training the Deformable Parts Model (DPM) [FGMR10] 2D category recognizer on collections of training images.

6.1.3 Chapter Outline

In the next section, we will describe how the probabilistic multi-viewpoint object model that was described in Chapter 5 is modified for sequential recognition and simplified to focus our discussion on a simple evaluation of our planning method. Subsequently, we will describe our strategy for learning viewpoint detection functions, and an entropy minimization next-best-view planning algorithm. Finally, we will present the results of our evaluation of the system on a simulator and discuss a proof-of-concept demonstration of the technology on board the Curious George robot.

6.2 Sequential Category Recognition Model

In this chapter we focus on the planning aspect of the multiple viewpoint robot recognition problem. To reduce complexity, we simplify several of the other aspects of the problem. In particular, we assume that locations of the world have been identified as potential object candidates (proto-objects), for example by a mid-level visual attention system that identifies regions of the world that are likely to be objects, but does not verify their semantic category or spatial properties. We also specify a simplified objective, of optimizing recognition performance after as few views as possible. Note that this ignores other planning objectives, such as minimizing robot travel distance. We restrict our focus to informative views here, in order to achieve a limited complexity and expect that additional planning constraints will be added when the technique is applied on-board a physical system such as Curious George. This leaves the task of choosing the next viewing angle from which to observe one of the candidate objects. Additionally, for the purposes of this chapter, we consider the scenario where a single new object has been encountered and the robot is tasked to verify its identity before moving on to the next candidate object.

We represent this restricted variant of robotic object recognition using small-case *o* to represent that there is only one object considered at a time. This corresponds to a state of $O = O_1$ in the terminology of Chapter 5. Our task is inferring a binary category label o^c, or simply c, which expresses that the proto-object is a member of the queried category or not. We will neglect some spatial properties of the object that are considered elsewhere, such as o^{cent}, through the assumption that a proto-object identified by an attention system has a fixed position. A component of our viewpoint planning process is to jointly reason about the object's orientation, so we will treat one angular dimension, rotation about the *up* vector known as azimuth, as a latent variable. We write this symbolically as o^{az} or simply az. As images are collected, we will model $p(c|d(I(\theta_1)),...,d(I(\theta_i)))$, the probability that the proto-object is a member of category, c, conditioned on the responses of an object detector over the i images, collected from viewpoints, $[\theta_1...\theta_i]$, by the robot so far. Note that we also exclude representation of data-association here. At each step, our planning system must choose the next viewing angle, θ_{i+1}, from which to observe the object, which is a simplification of the general motion planning problem for a mobile platform that neglects detailed interactions such as planning paths through maps and avoiding obstacles.

Our solution to the planning problem is inspired by the observation made by [LA06], that appearance models for objects are highly correlated to the angle from which those objects are viewed. In order to incorporate this intuition into the recognition and planning process, we jointly infer the category of the object being considered as well as the pose of the object. Specifically, we have trained a number of generative detector models: $p(d(I(\theta_i))|c,az)$, where d represents the response of an object category recognizer when presented with an image patch of an object with category c oriented at angle az and observed from viewing angle θ_i. For simplicity in much of the discussion, we will describe a detector's response as d_i, indexing only by i, the order in which the image was taken. The reader is asked to remember that the detector's response is a function of the viewpoint and environmental factors.

6.2.1 Learning a Viewpoint Function

As mentioned, a number of factors including training data bias and object shape properties affect the responses of an object detector over the viewpoints of an object. Correctly modeling this fact will allow a visual search system to correctly infer the state of the world, and so we set out to model the detection response as a function of viewpoint for several state-of-the-art object recognizers trained on a variety of datasets. In particular, we have examined three object recognition approaches that are currently used heavily in computer vision. Please note that in this chapter we have not considered so-called *bank-of-detector* approaches that are specifically targeted to recognizing individual viewpoints, as is done in later chapters of this thesis. Such approaches would give more specific viewpoint prediction capability and could potentially be a great benefit to our planning method. We have chosen standard single detectors here in order to validate that our approach works even with such inputs and we leave planning over detector banks for future work. The detectors used in this chapter are:

1. *SIFT matching* is an algorithm based on the observation that local image features can be reliably detected and described in a fashion that is largely invariant to changes in scale, lighting and in-plane rotation [Low04] (N.B. the list of invariances does not include viewpoint changes, although invariance over a small range of views is possible, as discussed in [MTS$^+$05]). In particular, we have implemented image matching based on SIFT features with RANSAC to fit a fundamental matrix to a candidate set of point matches in order to discard outliers and return highly confident match results.

2. *Bag-of-Features matching* is equivalent to SIFT-matching without checking of the geometric consistency between feature matches. This allows the method to generalize better across intra-category variation in geometry and makes the approach more suitable for category recognition. Note, for clarity, that the method we call *Bag-of-Features matching* is the simplest possible modification of SIFT matching in that we

match to nearest neighbors in the original feature space and have not utilized vector-quantized features or an SVM for classification. Those extensions and others have been attempted by [GD05] and other authors. Algorithms of this nature often share the "Bag-of-Features" description, and are likely to have different properties, so we explicitly draw the distinction for clarity.

3. *Deformable parts model* is an algorithm that combines several feature types and jointly infers parts and object labels with an SVM. This method was selected due to its strong performance on the PASCAL VOC [EVW$^+$12]. We have used the author's implementation for this method [FGMR10].

Each of the three methods was evaluated across a large number of views drawn from the "Multi-view Object Categories" dataset which has recently been collected by Savarese *et al.* [SFF07]. Recall that each image in this dataset contains a category label and a viewpoint label. The results of a detector on this dataset characterize its distribution of responses as a function of category and viewpoint. We modeled the empirical distribution of detector response with a univariate normal per $\{c, az\}$ pair. This produces a generative *viewpoint detection model*, $p(d|o, az)$, which can be evaluated for each detector response and integrated into the overall recognition framework as will be shown below.

Several viewpoint detection models for the DPM recognizer are displayed in Figure 6.2. Each row in the image represents the response given for a different category: bicycle, car and monitor. Some notable structure is present in each. Responses for the bicycle category show clear symmetries, and, as was clear in Figure 6.1, the front and back views give much lower detector responses than views from close to the side. Responses for cars have a similar shape, but the front and the back views are somewhat more recognizable due to a car's greater width and identifiable features such as headlights. Finally, the response function for monitors shows that the canonical straight-on front view is highly recognizable while there is very little information available in rear or side views of a monitor. Note that in all cases, the expected detector output (shown in red) is significantly higher for positive images where the category is truly present, than for negative images that do not contain the object. The ratio between the positive and negative responses describes the discrimination power of the recognizer. This indicates the DPM model is a relatively effective recognizer for objects in the set of images we have used for evaluation.

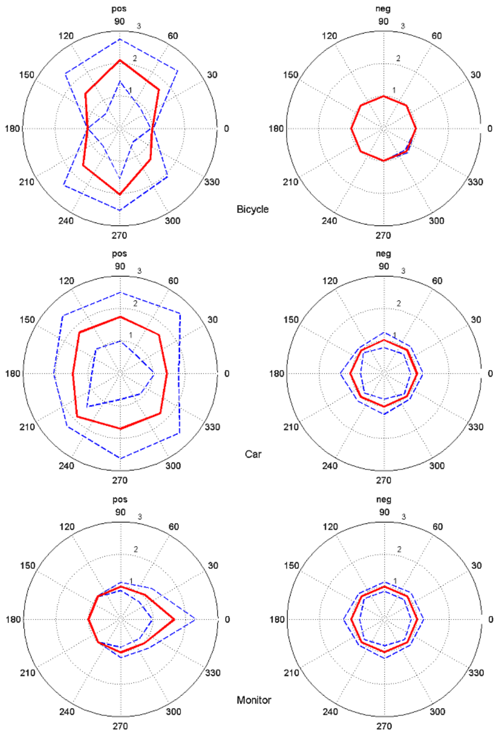


Figure 6.2: Viewpoint-Detection Functions per Class: Example viewpoint detection functions that are learned for the DPM recognizer for classes: (top) bicycle, (middle) car, and (bottom) monitor. The radial coordinate represents the detector response to positive(left) and negative(right) samples. The angular coordinate represents the viewing azimuth of the sample. The solid red line is the expected value and dotted blue lines depict the uncertainty in the response.

Figure 6.3 shows the viewpoint detection models of the *SIFT matching* and *Bag-of-Features matching* approaches when trained to recognize bicycles. The viewpoint profile of the responses for both methods are similar to those observed in the previous figure, which adds support to the observation that side views of bicycles are more readily distinguishable than front and rear views. In contrast to DPM, however, we found that these detectors' response functions for negative instances (images that do not contain bicycles) were nearly as strong as those for the positive instances (images containing bicycles) over most of the viewpoint range. This is due to the fact that the feature matching step in both of these approaches returned a small number of features for instances of the category not present in

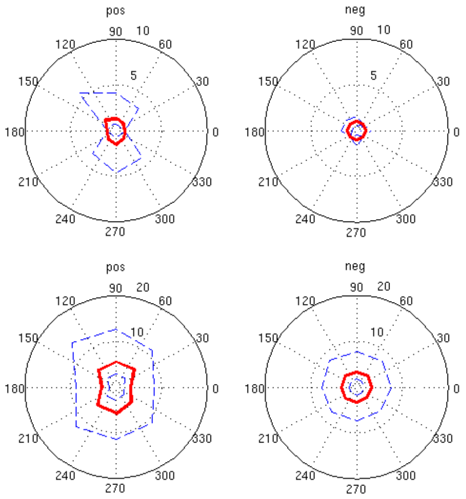


Figure 6.3: Viewpoint-Detection Functions per Detector: Viewpoint detection function for the (top)*SIFT matching* and (bottom) *Bag-of-Features matching* detectors. The radial coordinate represents the detector response to positive(left) and negative(right) samples. The angular coordinate represents the viewing azimuth of the sample. The solid red line is the expected value and dotted blue lines depict the uncertainty in the response.

the training set. That is, the local object appearance varied too greatly for correct matching. This is can be seen in the figure in that the mean values for both positive and negative responses are similar. For this reason, we have primarily focused on the DPM approach in the rest of the results given in this chapter. Integrating a specific view recognizer such as the *SIFT matching* approach with a general category recognizer is left for future work.

6.2.2 Multi-view Bayesian Estimation

This section describes our approach to integrating the scores of classifiers over images of an object from multiple viewpoints. We build upon the viewpoint detection models described previously. Consider inferring $p(c,az|d(I(\theta_1)),...,d(I(\theta_N)))$, the probability that an object is present and has orientation az, based on N detector responses in images captured from viewpoints $\theta_1,...,\theta_N$. This is derived using Bayes' Rule as

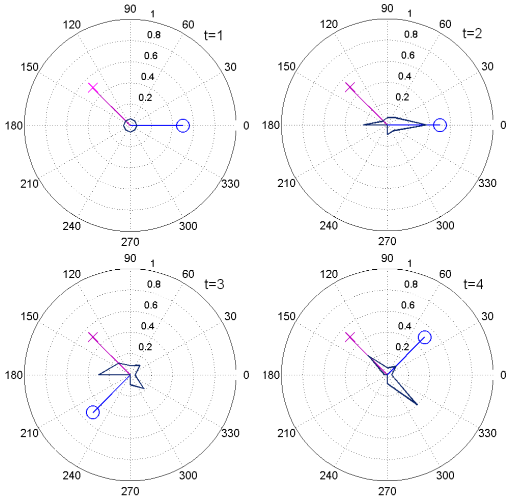


Figure 6.4: Evolution of Posterior Example: The posterior distribution over object presence and pose is updated as each image is collected. This is demonstrated for 4 steps of one robot recognition simulation trial. The graphs display: the prior top-left $p(c,az)$, the posterior after one image top-right, $p(c,az|d_1)$, and so on. In each graph, the radial coordinate represents the belief probability for the object occurring and having the pose indicated by the angular coordinate. This trial is evaluation of the category label "car" and the true world state is that a car is present with pose 135°. The magenta "x" shows the pose of the object and the blue circle shows the robot's pose at each time step.

$$\begin{aligned} p(c,az|d_1,...,d_N) &= \frac{p(d_1,...,d_N|c,az)p(c,az)}{p(d_1,...,d_N)} && (6.1) \\ &= \frac{p(d_1,...,d_N|c,az)p(c,az)}{\sum\limits_{c_i\in\{t,f\}} \sum\limits_{az_j\in\{0,\frac{\pi}{4},...,\frac{7\pi}{4}\}} p(d_1,...,d_N|c_i,az_j)p(c_i,az_j)}. && (6.2) \end{aligned}$$

We make the standard naive Bayes assumption, that each pair of classifiers is conditionally independent given the object label and viewpoint. Also, we apply an uniform prior for $p(c,az)$, so it can be factored out. The expression becomes

$$p(c,az|d_1,...,d_N) \quad \approx \quad \frac{\prod_{k=1}^{N} p(d_k|c,az)}{\sum_{c_i \in \{t,f\}} \sum_{az_j \in \{0,\frac{\pi}{4},...,\frac{7\pi}{4}\}} \prod_{k=1}^{N} p(d_k|c_i,az_j)}. \tag{6.3}$$

This expression represents the probability of a given object configuration based on the observed data (detections). It is based on the generative viewpoint detector models for $p(d_k|c,az)$ that are learned from data, as described previously. Our use of a uniform prior for $p(c,az)$ is appropriate here since we are modeling each object in an unbiased fashion. In extensions to integrated systems, it is likely to be beneficial to use domain knowledge to specify an informative prior such as the likelihood of each type of object occurring in each room of a house, or the fact that all objects in a parking lot are likely to be mutually parallel. Several examples are discussed later in this thesis. Also, please note that we have excluded a model for robot motion in this work. For simplicity, we assume that the robot's motion is known exactly. While this is not true in general, our work makes a very coarse discretization of angle into 8 bins, and so it is likely that we can correctly determine the correct bin for the robot's position a large fraction of the time from odometry or SLAM position estimates.

Equation (6.3) expresses the distribution over the object label and viewpoint after N observations, but we must also consider how to update this model as new images are acquired, along with new detection results. $p(c,az|d_1,...,d_N,d_{N+1})$ involves the addition of another viewpoint detection model term to the numerator and requires re-normalizing by computing a new denominator that involves terms for all $N+1$ images. Figure 6.4 illustrates how the joint distribution over object category and viewpoint evolves over each time step for the object category "car". As each a new observation is made, the updated function becomes narrower and eventually puts the majority of its belief on the correct object pose and its reflection. This corresponds to probabilistic estimation of pose and category.

6.2.3 Viewpoint Planning

The active component of our robot recognition system requires a decision making strategy to control the position of the camera in the world – the viewpoint from which objects are observed. The choice of camera motions allows numerous views to be collected so that, for example, the canonical viewpoint present in the training data can be observed, or a view can be obtained that allows objects with similar appearances to be disambiguated. We employ entropy as a measure to determine the confidence of the recognition system in its belief about the presence (or absence) of the object. The entropy of a random variable x is defined as

$$H(p(x)) \quad = \quad -\sum_i p(x_i)\log(p(x_i)). \tag{6.4}$$

For the viewpoint planning problem, we attempt to minimize the expected entropy of the posterior over object category membership c, after selecting the next viewpoint, θ_{i+1}. Note that the previous statement explicitly does not state our goal as entropy minimization over the joint posterior on category and orientation. This is a design decision and has been made due to the fact that our system is generally able to disambiguate an object's category, but for many objects, the orientation remains uncertain even after many views are collected. By marginalizing over orientation, we separate these concerns within the planner. In symbols, our goal is expressed as

$$\theta^*_{i+1} \quad = \quad argmin_{\theta_j \in \{0,\frac{\pi}{4},...,\frac{7\pi}{4}\}} E[H(p(c|d_1...d_i,d_{i+1}(I(\theta_j))))]. \tag{6.5}$$

The distribution $p(c|d_1...d_i,d(I(\theta_j)))$ is derived from Equation (6.3) by marginalizing over all possible orientations. This is tractable because we have discretized into discrete orientation bins. The expected value is used in Equation (6.5) because we are computing a quantity based on a yet unseen observation. $d(I(\theta_j))$ represents the value of the detection response after the robot moves to viewpoint θ_j, which has not yet happened. This is distinguished from detections up to the current time $d_1,...,d_i$, which we assume have already been integrated into our model. We estimate the expectation by averaging over samples for the response of the detector drawn based on the posterior from the previous step and our learned viewpoint detector models. This is a common technique in information theoretic planning (e.g., [PR09]).

This completes the description of our method to select next-best viewpoints based on

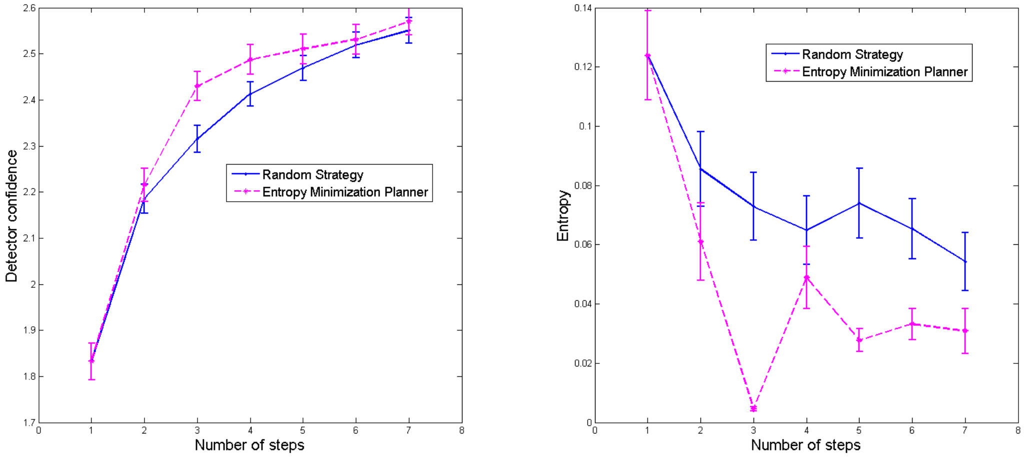


Figure 6.5: Viewpoint Planning Results: A comparison of detection results between a system using entropy minimization planning and a system which uses a random planning strategy. The graph on the left shows the sum of detector responses for true positives minus the sum of responses for true negatives, a summary statistic for classification performance. The graph on the right shows the entropy of the marginal $p(c = x|d_1, ..., d_i)$, the detector's belief in the true category label x. All results are averages over 160 random selections of an object instance and starting viewpoint.

learned viewpoint-detection models and a multi-view Bayesian model. Algorithm 2 formally outlines the overall procedure. The remainder of this chapter will describe our results and evaluation.

6.3 Experimental Results

We have constructed a simulated multiple viewpoint recognition environment to test our planning approach to this restricted robot recognition problem. Based on an early version of the *simulate from real data* protocol, our simulator models a robot's position with respect to an object, and returns a pre-collected image drawn from a hold-out portion of the Savarese *et al.* [SFF07] dataset used during validation. We evaluated the DPM category recognizer on each image and used the responses to update our recognition system's belief about the object's presence and viewpoint.

We compare our method with a non-adaptive viewpoint selection strategy that chooses a random previously unseen view at each time step. This method has been a favorite approach

Algorithm 2: Viewpoint Planning Algorithm

```
input  : Learned viewpoint-detection function $p(d|c,az)$
         Random initial robot position, $\theta_1$
Observe initial image, $I(\theta_1)$, and detector response, $d_1$;
Compute $p(c,az|d_1)$ using Equation (6.3);
/* move to n next-best views                                   */
for $i \leftarrow 1$ to $n$ do
    /* implement Equation (6.5) to find $\theta^*_{i+1}$         */
    $h_{min} \leftarrow \infty$;
    for $\theta_j \in \{0, \frac{\pi}{4}, ..., \frac{7\pi}{4}\}$ do
        /* draw m samples from generative model                */
        for $k \leftarrow 1$ to $m$ do
            $d_k \leftarrow \text{sample}(p(d|c,\theta_j))$;
            $h_k \leftarrow H(p(c|d_1,...,d_k))$;
        /* integrate over samples                              */
        $h_{\theta_j} \leftarrow \frac{\sum_k h_k}{m}$;
        if $h_{\theta_j} < h_{min}$ then
            $h_{min} \leftarrow h_{\theta_j}$;
            $\theta^*_{i+1} \leftarrow \theta_j$;
    Move the robot (real or simulated) to $\theta^*_{i+1}$;
    Observe image, $I(\theta_{i+1})$ and detector response $d_{i+1}$;
    Compute $p(c,az|d_1,...,d_{i+1})$ using Equation (6.3);
```

for contestants in the SRVC contest, and was suggested in [MFL$^+$08] as an approach that obtains coverage of viewpoints while reducing viewpoint overlap early in the search process. Compared to other non-adaptive strategies, the random approach may find interesting views faster, at the cost of additional robot motion.

Our comparisons were performed by using multiple trials of our simulator. At the outset of each trial, an object instance is chosen at random from the testing set. Also a random initial viewing angle is chosen from one of the 8 azimuth angles available in the Savarese dataset. The object's identity (whether or not it is of the target category) and its initial viewpoint are hidden from the planning approaches. So, the situation is a realistic approximation to the situation where the robot segments a proto-object from the world, has no prior knowledge about the category label or viewpoint of the object, and must infer these quantities by collecting and analyzing images.

Each trial proceeds with the simulator determining the current image based on the hidden object instance and viewing direction, as well as the known simulated robot position. The image space object recognizer is run on the image and its detection result is provided to our multiple viewpoint approach. This information is assimilated into the belief and the updated information may be used as stimulus for the next control action which is a request for a new viewpoint. Note that, in the case of the base-line random strategy, the detection information is integrated, but it does not impact the selection of the next action. The simulator responds to the motion command, updating its simulated viewing direction. The process repeats with the next simulated image. For statistical significance, 160 trials were conducted with an equal probability ($p = 0.5$) of the target object category or a distracting object being selected for each trial.

Figure 6.5 summarizes the results of the simulation trials. Planning to reduce entropy allows the recognition system to confidently infer the category label from fewer test images, since it is able to use the history of detector responses to determine the viewpoints that are most likely to discriminate the object. As more and more views are collected, the probability that the random strategy finds these views increases also, and once each method has exhausted the available viewpoints, performance is identical. Note that in this dataset, 8 unique views corresponds to every available image having been seen so all methods give similar performance at this point. The right of the figure demonstrates the our system becomes more confident, as measured by posterior entropy, with active planning. The rapid initial in entropy results from the planner discovering discriminative views, and the subsequent small increase results from the fact that we force the planner to continue even after it has essentially converged on its decision about the object. Later in the process, it encounters the viewpoints that are difficult to discriminate (recall that we only choose between 8 viewpoints overall and do not allow for planners to repeat the same view). A decision could likely have been made before this point. In both cases, the results demonstrate that adaptive, entropy minimization planning aids in the sequential object recognition process.

6.3.1 Application to Visual Search with the Curious George Platform

The Curious George platform uses visual saliency and depth cues to locate possible objects in the environment. As mentioned above, these mid-level vision techniques limit the search space which includes infinite locations and point of views. Figure 6.6 shows a sample

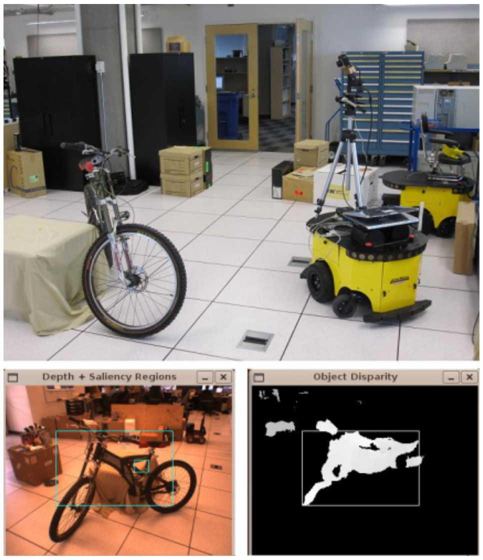


Figure 6.6: Example Use-case for Viewpoint Planning: Curious George looks at a bicycle and segments it from the background using its visual attention system.

scenario where robot has identified a proto-object in its view. The bicycle is correctly segmented based on depth and visual saliency features, in real-time, and this candidate object is passed to our system for evaluation. The viewpoint planning method described above is integrated with this pre-existing feature of the robot. We have previously applied a planning algorithm which weighs between multiple objectives such as map building, coverage of the environment and certainty of object labels, and the adaptive recognition method described here is an additional component that we plan to integrate into Curious George's planning suite in the future.

6.4 Chapter Summary

This chapter has outlined an active multi-view framework that can be used by an embodied visual searcher to infer the identity of a target object being considered. We have demonstrated the dependence of state-of-the-art object recognizers on the viewpoint from which an object is seen. This relationship is always likely to be present given the wide variety of appearance amongst category members across viewpoints. We have learned viewpoint-detection models for a number of detectors, and demonstrated that the sequential Bayesian estimation approach is capable of leveraging these models to provide improved recognition performance when compared to planning strategies that do not adapt to current models. Our method has been evaluated on a simulator based on a dataset of challenging images and its applicability has been illustrated for a physical embodied platform: Curious George.

There are several natural extensions to the current work. In this paper we have evaluated three object detection algorithms, but have chosen the one which performed best overall to use in the majority of experiments. Instead, a visual search planner could be given the opportunity to integrate information from all detectors, or better yet, the visual searcher could choose which method to run at each viewpoint, prioritizing computation towards detection results that are likely to be informative. Also, we have focused our analysis to the visual search problem involving only a single target object. In a home environment, a robot is faced with a large number of potential targets, and it may also be tasked with exploring new regions to discover new objects. In this case, a visual search platform must choose between numerous potential objects as well as between the viewpoints for each object. This is a challenging problem, but solving it will produce an active visual search robot capable of determining the semantic categories of objects within a home and subsequently performing useful tasks for the human inhabitants.

Chapter 7

Object Parts and Occlusion Modeling

7.1 Introduction

This chapter describes a method for 3D object localization applied to indoor kitchen environments. Our approach focuses on one of the most challenging issues facing object recognition in the real world: visual occlusion caused by clutter. Typical kitchens are often cluttered to the point where even humans struggle to find what they are looking for (e.g., the lost keys scenario). The performance of current automated recognizers is often quite poor in such environments, with only fully visible objects being recognized at a high confidence level. Occlusions hide a portion of an object's appearance so that the corresponding features are unlikely to support a successful detection. Several image space recognizers explicitly handle occlusions (e.g., [GFM11]) and others achieve inherent robustness, such as methods based on local feature patches, for example. However, even these approaches tend to be increasingly less confident as the level of occlusion increases, as they lack an external cue to measure occlusion and the image space appearance eventually resembles a typical background patch.

Here we describe a multi-view recognition system specifically designed to locate objects in cluttered indoor environments using the sensor sequence available from a moving intelligent system. We note that occlusions can be less problematic for a moving camera system, because we are more likely to obtain at least one clear view. However, typical information fusion approaches (e.g., Equation (5.8), our probabilistic model) attempt to explain an object's appearance in all views. So, heavy clutter will still lead to lower scores, espe-

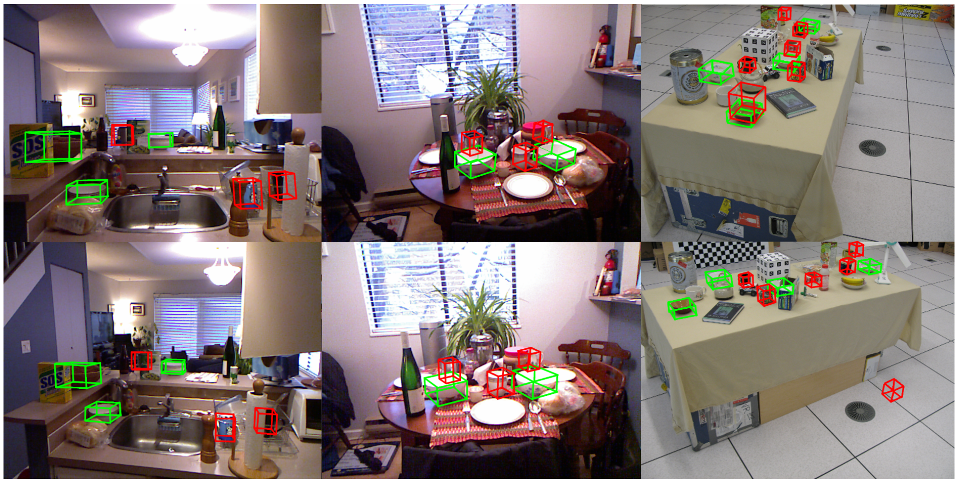


Figure 7.1: Sample Results from Parts-Model: Our method's results for two real home scenes (left and mid) and a synthetic lab scene (right). 3D wire-frames indicate mugs (red) and bowls (green). Thresholded at 90% precision. All figures are best viewed in colour.

cially when the occlusion occurs in several images. To overcome this, we use sensed 3D geometry, along with the fact that we perform recognition in three dimensions, to explicitly compute the expected visibility of objects and to discount the information from occluded viewpoints. This is effective when at least one image has a clear view of the object, but we go further by applying learned partial-object appearance templates (e.g., a left-half mug detector) and reasoning about occlusions at this partial-object level. This improves our per-view occlusion handling even further and allows recognition of quite heavily occluded instances. Figure 7.1 demonstrates the ability of our method to locate such objects in 3D.

7.1.1 Relation to Generic Multiple Viewpoint Object Model

The model described in this chapter is an instance of the generic multiple viewpoint 3D object model. The object state is a full 3D location and scale, but here we neglect object orientations because each of the object categories considered is nearly symmetric, which leads to uninformative viewpoint-appearance distributions. We focus on occluded objects and therefore augment the appearance likelihood with additional partial-object models. 3D objects are matched to whole-object and partial-object detections. An occlusion mask derived from the current 3D object state plus range information measured by a laser allows our model to down-weight the contributions of occluded portions of an object.

For continuity, we briefly describe the relation of components in this chapter to the generic model components described in Section 7:

1. *State space*: category label, 3D position and 3D scale for each object in the scene.

2. *Evidence*: visual images and point clouds from a number of viewpoints, as contained within the UBC VRS dataset, which is used for evaluation here.

3. *Object Prior*: category-specific size prior on each of the height and radius of the mostly cylindrical indoor objects.

4. *Appearance Likelihood*: whole and partial object detectors learned independently using the DPM method [FGMR10]. Greedy data association of 3D objects to full and partial detections. Weighing of contributions of each type based on estimated amount of occlusion.

5. *Geometry Likelihood*: scan-matching term to measure the agreement of object surfaces and sensed ranges.

6. *Learning Method*: the training procedure of the DPM classifier, repeated one additional time for each partial template.

7. *Inference Method*: Monte-Carlo search with refinement using local gradient information.

7.1.2 Statement of Collaboration

This chapter represents nearly the same material that the author published as [MWSL11]. The co-authors of [MWSL11] include Christian Wojek and Bernt Schiele, who worked with the author of this thesis during a research stay with the Computer Vision and Multi-modal Computing group at the Max-Planck Institute, in Saarbrücken, Germany. The author of this thesis was responsible for the overall concept, implementation, evaluation and analysis of the method under supervision and with advice from the co-authors. In particular, our approach was inspired by the previous work of Dr. Wojek [WWRS11] for detecting occluded pedestrians in street scenes. We extended his previous mixture-of-experts model by integrating sensed 3D information, developed a novel efficient inference approach and applied the technique for recognizing objects in cluttered kitchens.

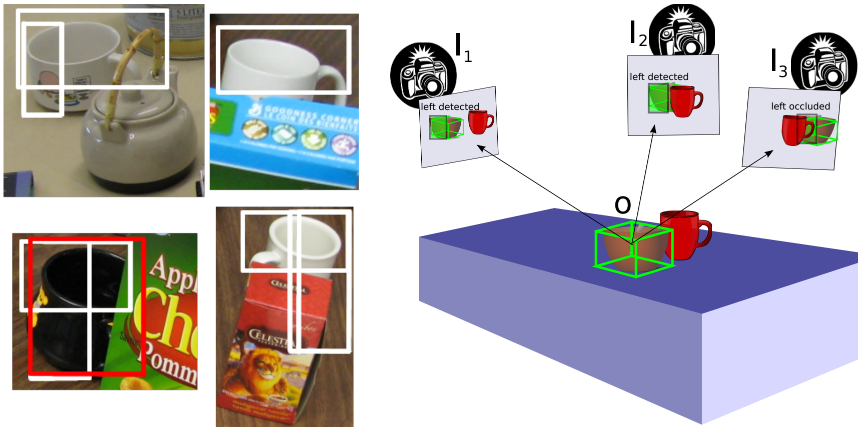
7.1.3 Outline

We will continue by describing a 3D object model that has been designed to be robust to occluded object instances in Section 7.2. Learned visual appearance templates for portions of an object allow for strong discrimination even under occlusion. We then describe an efficient inference procedure based on data-driven sampling with geometric refinement in Section 7.3. Finally, in Section 7.4 we demonstrate our 3D object detection technique on the UBC VRS dataset that has been described previously, as well as on data from the Microsoft Kinect collected in real home environments. Results show that our method improves robustness to occlusion when compared to a state-of-the-art visual category detector.

7.2 Object Parts and Occlusions for 3D Object Modeling

Consider the problem of recognizing an object that is partially occluded in an image. The visible portions are likely to match learned appearance models for the object, but hidden portions will not. This is a primary cause of poor recognition performance for modern approaches. The (hypothetical) ideal system would consider *only* the visible object information, correctly ignoring all occluded regions. In purely 2D recognition, this requires inferring the presence and nature of occlusion, which is a significant challenge since the number of possible occlusion masks is large. We simplify the problem, considering only half-object occlusions, leaving four partial detectors: top, bottom, left, and right. Note that, while our partial-object templates may still not perfectly match the object's visibility, they will often match more closely than a full-object object template. We train a partial-object detector tailored exactly to each chosen case. In addition, we reason about objects in 3D and incorporate sensed geometry, as from an RGB-depth camera, along with visual imagery. This allows explicit occlusion masks to be constructed for each object hypothesis. The masks specify how much to trust each partial template, based on their overlap with visible object regions. This comes close to the intuition – only the visible evidence contributes to our object reasoning.

This occlusion reasoning has been implemented within the framework of our multiple



(a) (b)

Figure 7.2: Part-detector Results and Model Overview: (a) Real results from partial detectors (shown in white) often respond when full-object models (shown as red boxes) do not, due to occlusion. (b) An object is projected and associated with partial detections where available.

viewpoint 3D object model. Figure 7.2(b) illustrates how partial-object detectors fit within our system's view of a scene. Each candidate 3D object location projects into all views and is associated to image space detections produced by visual category recognizers for the object's complete appearance as well as for a subset of the possible occlusions. Fully visible objects are likely to align well with strong detections in each image for both the entire object as well as its parts. However, occlusion can cause weak detection results. The sensed depth information allows us to estimate the occlusion of each part of the object in each view. The occlusion estimate is incorporated into the scene score, allowing our system to ignore meaningless appearance information from occluded regions and more faithfully representing the underlying geometry.

This section explains a model to compute the likelihood of any proposed 3D object, but does not consider how these objects should be proposed. That is left for following section, which outlines our sampling-based inference procedure.

7.2.1 Top-level Object Likelihood

We follow the generic road-map for describing 3D objects based on observed evidence, which has been previously described in Chapter 5. We briefly reviewed terms and equations here for consistency. Objects, O_i, are here represented as category label, 3D position and scale. We do not perform joint reasoning about groups of objects here, until a final non-maxima suppression step, so we will often refer to a single object with small-script o. Orientation is neglected here because we will deal with symmetric indoor objects such as bowls. Sensory observations include images, I_t, and point clouds, C_t, as well as mapping or registration information, ${}^W_V T_t$, from an external module such as structure-from-motion. The observed data from each view is $Z_t = \{I_t, C_t, {}^W_V T_t\}$ and all observations since the start of time are $Z^t = \{Z_1, Z_2, ...Z_t\}$.

We express the likelihood of an object given the available data using the naive Bayes assumption and by factoring it into independent generative models for each observation type as was previously described for Equation (5.11):

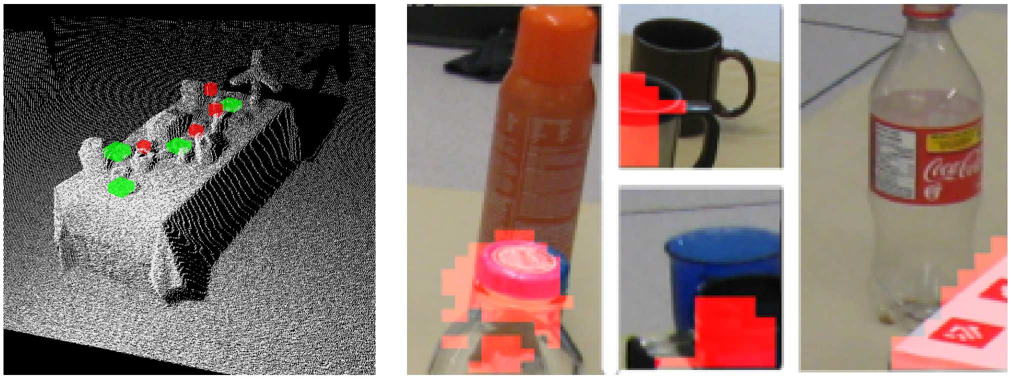
$$
\begin{aligned}
p(o|Z^t) \propto p(o)p(Z^t|o) &\approx p(o)\prod_t p(Z_t|o) && (7.1)\\
&= p(o)\prod_t p(I_t, C_t, {}^W_V T_t|o) && (7.2)\\
&= \underbrace{p(o)}_{object\ prior} \prod_t \underbrace{p(I_t|o, C_t, {}^W_V T_t)}_{appearance} \underbrace{p(C_t|o, {}^W_V T_t)}_{geometry} \underbrace{p({}^W_V T_t|o)}_{registration}. && (7.3)
\end{aligned}
$$

The size prior for each object category is written $p(o)$. Here, we model this as a normal distribution on both the height and radius of the object, which is appropriate given the cylindrical nature of the objects studied here. Other shape priors can easily be substituted. We will continue by describing the specific geometry and appearance likelihood terms that are used within this chapter in detail.

7.2.2 Geometry Model

The geometry model relates sensed depth data given to an inferred object location. As shown in Figure 7.3(a), inferred 3D object regions are placed in the same coordinate frame as measured 3D data. This allows us to process both forms of 3D data and to derive information that feeds our probabilistic models. For each pixel in the depth image within the projected object region, we note three types of outcome:

1. The measured depth is near to the inferred depth: they agree



(a) (b)

Figure 7.3: 3D Geometry Example: Point clouds and inferred 3D objects, as shown in (a), allow computation of occlusion masks for each object in each image, as in (b). Regions shown in red are deemed to be occluded by our automated reasoning approach.

2. The measured depth is greater than the inferred depth, which indicates the inferred object region is unoccupied: they conflict

3. The measured depth is less than the inferred depth: the object is occluded

The geometry term in Equation (7.3) is constructed from pixels that fall into the first and second outcome only, as occluded regions cannot tell us anything about the object's geometry. We employ a mixture of two Gaussians to model the expected error in the depth sensor and the rare occurrence of outliers far from the expected value. We compute the product of this model over all pixels expected to fall on the object. This model is common in geometric inference, and has been used previously in robotic mapping, for example in [TML$^+$03].

Figure 7.3(b) shows pixels marked with the third outcome above: occluded. The ratio of occluded pixels within the region considered by each partial-object appearance template forms a visibility score v used for the appearance model, as will be described in the next section.

7.2.3 Appearance Model

The likelihood of the image appearance given an object is evaluated through a version of the data association framework described in Section 5.3.2. To specifically target occluded instances, here we augment the usual learned model for the entire object with a number of sub-parts. We scan the image captured at time, t with each partial object model, where the type of part is indexed by p. This yields many hypothesized detections, $D^p(I_t) = \{d_1^{pt}, ..., d_j^{pt}, ...\}$, in each image, where j indexes a specific detection. As described for the generic multiple viewpoint model, the 3D object, o, is projected into each image and assigned to nearby detections using a data association function. *Greedy Matching* (Algorithm 1) is carried out once for each partial model, p.

We have modified the association function to consider occlusion mask data. Only visible object portions are considered for association. As previously described, up to one detection of each part-type is associated with each object, o, in each image. We express this associated detection using function notation: $A(o, {}^W_V T_t, D^p(I_t)) = d^{pt}$. This detection will contribute to the probabilistic model.

To express the likelihood of all associated partial detectors given an object, we employ a mixture of visibility-weighted experts model similar to that proposed in [WWRS11]. We project the 3D object, o, into the image using a projection matrix P_t that is derived from registration information, ${}^W_V T_t$, and the known camera calibration. Each associated detection (expert) is weighted by the visibility of the corresponding object region. As with all of our multiple viewpoint 3D object models, we also enforce soft geometric consistency by penalizing error in alignment between the object and an associated detection. The full likelihood is written symbolically as

$$p(I_t | o, C_t, {}^W_V T_t) \quad = \quad \frac{1}{\sum_p v^{pt} \delta(v^{pt} > \theta)} \sum_p v^{pt} \delta(v^{pt} > \theta) \Psi_s(d^{pt}) \Psi_g(P_t \cdot o, d^{pt}). \quad (7.4)$$

Recall that visibility, v^{pt}, is derived from the sensed depth within the region explained by the object (or object part) when the object and 3D data are rendered from the viewpoint of the camera at time t. δ is an indicator function to completely discount contributions of parts that are more occluded than a hand-chosen threshold, θ. We assume that when almost none of the image evidence represents the object, there is no meaning to the detector's score. Ψ_s is a potential function related to the detector's score. We have implemented Ψ_s here with a linear mapping of detector score to the range $[-1, 1]$ based on scores on a set of validation data. Platt Scaling, as in [NMC05], is an alternative that provides a meaningful probabilistic interpretation. We previously attempted to remap our scores in this way, but it gave no benefit in our experimental results. Ψ_g measures the geometric agreement between a projected object and its associated detection. This potential is implemented as a three-dimensional Gaussian distribution on scale-normalized error in object center (both x and y position) as well as error in scale in image space.

7.3 3D Object Inference

The previous section described a 3D object likelihood model to relate the presence and spatial properties of an object to the observed data. However, for every test environment, we must infer the objects that maximize this likelihood. As previously described in Section 5.5, exact inference has a high computational cost and scales poorly as many images are available. Instead, we employ data-driven sampling of likely regions, followed by refinement of each sample and non-maxima suppression. This allows only the most promising regions to be considered and saves considerable computation. The remainder of this section describes our inference procedure in detail.

Data-Driven Sampling

While it is expensive to search for the global maximum that simultaneously explains all observed data, we can efficiently compute the local maxima relative to each view by considering the terms in the product of Equation (7.3) one at a time. First, we draw a detection with probability proportional to the confidence score and constrain the 3D object center to align with the center of the detector's bounding box. This constrains the sampling-space to the ray in 3D over all (infinite) positive depth values. We must also choose a depth and a scale for each proposed 3D sample. We sample a depth from a distribution formed by constructing a histogram over the sensed range values within the detector's bounding box. Scales are drawn from the prior on the object's size. This one sample is saved for further processing and the process begins again by selecting a new detection.

Position Refinement

After the sampling stage, a set of 3D regions is available, and it is possible to score each region directly with Equation (7.3). However, unless a large number of samples is used, we have observed that the results are quite poor both in terms of 3D localization accuracy and in the likelihood score of each object. Due to the data-driven sampling being driven only by the evidence in a single view, a wide variety of poor locations can be chosen. Therefore, we refine each sample's location and scale to locally maximize the likelihood of corresponding appearance data in all views. Given a fixed data association, we derive the gradient of the geometry likelihood functions for all assigned detection evidence with non-zero weights from occlusion reasoning. We update the object information iteratively based on this gradient, until a local minimum is reached.

We found that refined object positions could also help to improve the initial data association. Therefore, we implemented another level of iterative refinement based on coordinate descent. First, the 6 degree-of-freedom (3D position and scale) object pose is optimized given a fixed data association as we have described. Second, the data association is re-computed for the new object location. The intuitive and observable effect of alternating these steps is that, optimizing the hypothesized object's pose using several images is likely to position it near to the true 3D location, which may lead to correct association with previously unassigned detections in other views. Projection is non-linear and greedy data association is a discrete process which makes our optimization space discontinuous. Therefore, we can make no guarantee on the convergence of the optimization, but we have found that the procedure works well in our empirical evaluation.

Non-Maxima Suppression

As with many detectors, our 3D object inference procedure tends to find many slightly shifted versions of each true object with high likelihood scores. We suppress detections which are not local maxima based on their overlap in 3D. Note that our approach can tolerate very cluttered scenes where two objects occupy nearly the same region in image space. As long as these objects have different depths, we will be able to maintain both hypotheses (there is no overlap in 3D). This is in contrast to many detectors that apply image space non-maxima suppression which performs poorly when two objects of the same category are nearby in the image.

This completes the discussion of our occlusion-aware object inference method. Algorithm 3 provides an overview of the procedure. The next section will continue by describing a practical implementation of the method and present results.

7.4 Experimental Setup

We have implemented a complete 3D object detection system, instantiating each of the model components described above in a robust fashion in order to evaluate their performance in realistic, cluttered indoor scenes. We evaluate our approach both on the UBC

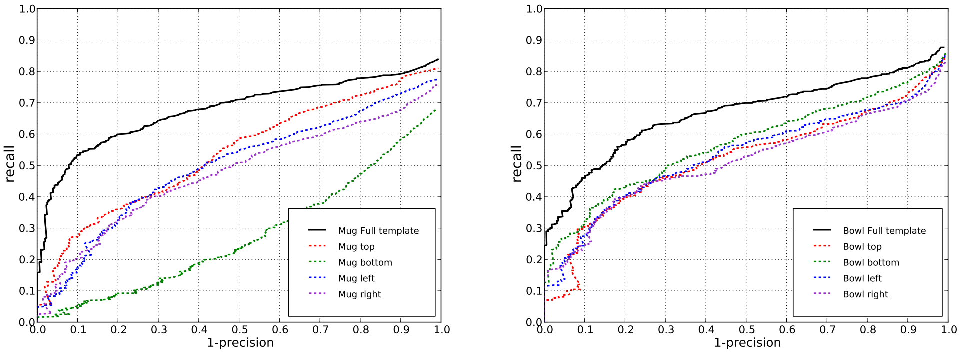
Algorithm 3: Occlusion-aware 3D Object Inference Algorithm

```
input : Sensory data, $I_t$ and $C_t$, from each of $n$ robot positions
        Registration information, ${}^{W}_{V}T_t$, for each position
        Whole and partial detections, $D^p(I_t)$, in each image
output: A set of 3D objects, $X$
$X \leftarrow \emptyset$;
/* sample and refine m objects                                   */
for $i \leftarrow 1$ to $m$ do
    Sample an object, $O_i$, from detection and depth ;
    while not converged do
        for $t \leftarrow 1$ to $n$ do
            for $p \in \{full, left, right, top, bottom\}$ do
                Assign $d_i^{pt}$ to $P_t \cdot O_i$ using GreedyMatching, Algorithm 1;
        while not converged do
            $O_i \leftarrow O_i - \nabla \sum_p \sum_t \Psi_g(P_t \cdot O_i, d_i^{pt})$;
    Compute object likelihood with Equation (7.1);
    insert $O_i$ into $X$;
return Non-Maxima-Suppression($X$);
```

VRS test set as well as on several scenes in a real home captured with the Microsoft Kinect sensor. The DPM recognizer is used to detect the bowls and mugs within these scenes and our multiple viewpoint model uses registration information to fuse information across viewpoints of each scene. This section will describe the practical details of the evaluation including the learned visual recognizers used as input, the test data, and the structure-from-motion algorithms employed.

7.4.1 Visual Detectors

We learn detectors for each of four half-sized templates: top, bottom, left, and right. Each partial-object detector is trained independently, as this allows the hard negatives for each template to be included, maximizing resulting detection performance. We employ the Deformable Parts Model of Felzenszwalb *et al.* [FGMR10] both for appearance learning and also for test-time detection in images. Both full object models and partial templates are learned from the same training data, specifically the UBC VRS training images. This requires modifying the positive annotations to include our four half-object partial models.



(a)

(b)

Figure 7.4: Performance Evaluation for Whole Object and Part Detectors: Results for categories: (a) mug and (b) bowl.

Figure 7.4 shows the performance of our full template and partial object recognizers over a set of validation images containing annotated examples of each category. The clear trend is that the complete template achieves the best performance overall, which is intuitive because it considers the largest image region and can therefore most strongly discriminate objects from background. We note that the performance of a partial detector does not change at all if an instance is occluded in regions ignored by the template (i.e., a left detector is unaffected by occlusion on the right), while the full model is always affected. This can be leveraged during 3D inference.

7.4.2 Evaluation Scenarios

We evaluate our method on two scenarios involving indoor clutter and occlusion. First, we locate objects in the UBC VRS test set. We evaluate the performance of our method using the recognition scenario that has been defined for our dataset and that utilizes robot simulation software to approximate the visual experience of a robot moving through a variety of scenes. The UBC VRS simulator's view selection strategy was asked to choose sequential images, which best replicate the path traveled by the robot during data collection. We have performed quantitative analysis of our approach on this task by performing precision and recall analysis with the average precision statistic (details found in Section 2.6) for a number of variants of our approach as well for the DPM method as a comparative baseline. Since our approach localizes objects in 3D, we projected its object hypotheses into image space and compared these against the same labeled bounding boxes that are used to evaluate DPM.

Our second method for evaluation involves a small amount of novel data collected for this chapter with the Microsoft Kinect sensor in a real home. Here, the sensor was hand-held and we performed marker-less position registration, as will be described below. We present our results from this portion of the data qualitatively, as insufficient annotations have been collected so-far to achieve meaningful quantitative comparison. The goal is to demonstrate our method's strong performance on realistic scenarios.

7.4.3 Structure From Motion

Our experiments include two separate structure-from-motion techniques. The images in the UBC VRS are registered using a target made of ARTag fiducial markers [Fia05]. For a complete description of the registration process see Section 4.2.2. Our Microsoft Kinect data contains unstructured, real home scenarios and no calibration target has been used. Here, we have employed an off-the-shelf technique named "RGBDSLAM - 6DOF SLAM for Kinect-style cameras"[1]. Like many structure-from-motion solutions for hand-held cameras, Speeded-Up Robust Feature (SURF) [BTV] points are tracked between frames, and a set of geometrically consistent inliers is found with Randomized Sampling and Consensus (RANSAC) [FB81]. Long-range performance and loop-closure is achieved by refining poses globally using the technique described by [GKS$^+$10]. Registration is not sub-pixel accurate in this scenario, which demonstrates robustness to errors in our geometry model.

7.5 Experimental Evaluation

This section presents results from the evaluation described above. We will first describe the qualitative performance of our system on the hand-held Kinect data and we will then discuss quantitative results on the UBC VRS test set.

7.5.1 Qualitative Results

Figure 7.5 shows a number of example results from our method. In many scenes, our technique can leverage the reliable information available from visible parts of objects, and confidently locate their position in 3D, even in clutter. However, our system returns false

[1] http://openslam.org/rgbdslam.html

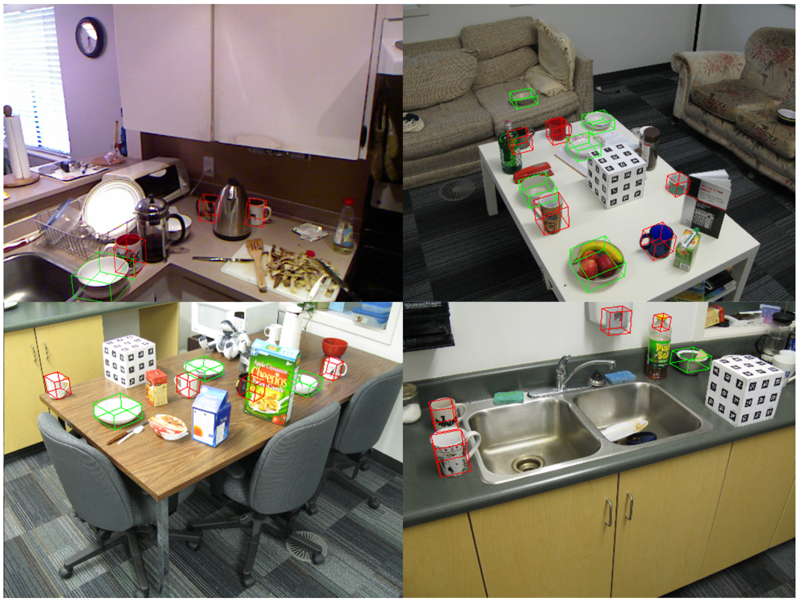
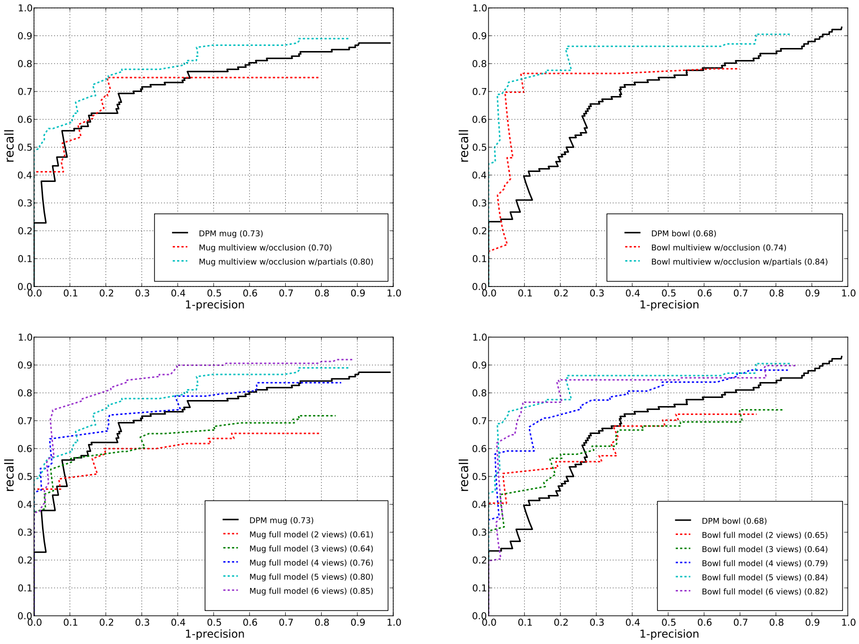


Figure 7.5: Qualitative Parts-Model Results: Sample results of our 3D object detection method, thresholded at 90% precision. The top-left image is from the Kinect sensor, the remainder are robot-collected.

positives on objects whose visual appearance and structure are similar to the searched category. The bottom-right image in Figure 7.5 shows a soap dispenser and the top of a bottle which are both labeled “mug”. Additional geometric constraints may be able to filter these objects as being in unlikely positions (not resting on table).

7.5.2 Comparison to Image Space Recognition

Figure 7.6(top) shows the results of our 3D object recognition approach, a variant without partial detectors, and the purely image space DPM model [FGMR10]. In all cases, evaluation is on the test set of the UBC VRS dataset. For the 3D detection methods, object volumes are projected to form bounding boxes for scoring. In some cases, our complete model detects 40% more of the annotated objects, for the same miss rate, than the DPM detector. The effect of partial detections is shown by the improvement of the complete model over the variant using only full appearance templates. Further inspection reveals that partial detections improve performance primarily on occluded objects.



(a) (b)

Figure 7.6: Result Comparison with State-of-the-art 2D Recognizer: (Top) The performance of our method evaluated over 5 viewpoints vs. the state-of-the-art DPM model by [FGMR10]. (Bottom) Full model performance vs. number of viewpoints. Columns: (a) mugs, (b) bowls. The summary statistic is Average Precision.

7.5.3 Altering the Number of Viewpoints

Figure 7.6(bottom) shows the results of our method as the number of views considered is increased from two to six. The method must generally observe four or more views of the scene before it outperforms the image space detector, although all multi-view models perform better at high precision. Visual inspection shows that the issue is poor 3D location of objects when only two or three views are available. We note that the baseline available to our system in this case can be as small as five degrees, the spacing between consecutive frames in some scenes of the UBC VRS data. When 3D locations are poorly estimated, objects project to incorrect image space locations, degrading performance. We have run a similar two-view experiment with a setting for the UBC VRS simulator's view selection criteria that yields maximally separated views of the scene, rather than nearby sequential views. Our method's performance in this trial was 0.75 AP for mugs and 0.72 AP for bowls, which improves upon the DPM score, and rivals the four-view approach run on consecutive frames. This result indicates that achieving a significant baseline is an important consideration for a system designer, or for an active viewpoint planning strategy, as we have previously described.

7.6 Chapter Summary

We have developed a method that relates occluded 3D objects to incoming image and geometry data from many views. Our approach performs explicit occlusion reasoning and leverages learned partial-object appearance models. Results demonstrate the potential for robust object detection in home scenarios, where intelligent systems will soon be deployed. Our approach is integrated with a structure-from-motion system, and in combination the techniques form a semantic mapping system suitable for object-centric tasks such as scene description to disabled users or mobile manipulation.

Our motivation to select four half-object partial templates in this chapter was that it provided a simple starting point for our analysis of the use of detailed information for the 3D object recognition task. We do not claim that these detector shapes are in any way optimal, and the next chapter will continue our exploration by forming models with many more parts, and by enforcing 3D continuity for those parts by deriving them from labeled 3D models of objects.

Chapter 8

Detailed Object Parts for 3D Scene Understanding

8.1 Introduction

This chapter describes a system to locate automobiles in images that are captured while driving through a crowded parking lot. Strong performance on this task would enable applications such as anti-collision systems, however the visual task is challenging and has required a number of extensions to our previous methods. Some portions of parking lots exhibit even denser clutter than the kitchens we have previously considered. This has motivated us to continue the study of parts modeling that was introduced in the previous chapter in order to make best use of the, potentially small, portion of each vehicle that is visible through the clutter. As is depicted in Figure 8.1, we consider a larger number of parts per object, model those parts as first-class 3D entities to allow parts to remain consistent over viewpoints and choose semantically meaningful parts such as the wheels and doors of the automobiles. We utilize multiple (up to 4) image space recognizers, some of which are targeted specifically at understanding consistent 3D parts across viewpoints, in place of the half-object DPM models that were used in the previous chapter.

In parking lots, automobiles are arranged in lines, one behind the other. This dense stacking makes separation of multiple nearby object hypotheses problematic, even in 3D. In this chapter we consider a more principled method for reasoning about the dependence between objects than in our previous methods. Specifically, we have extended our proba-

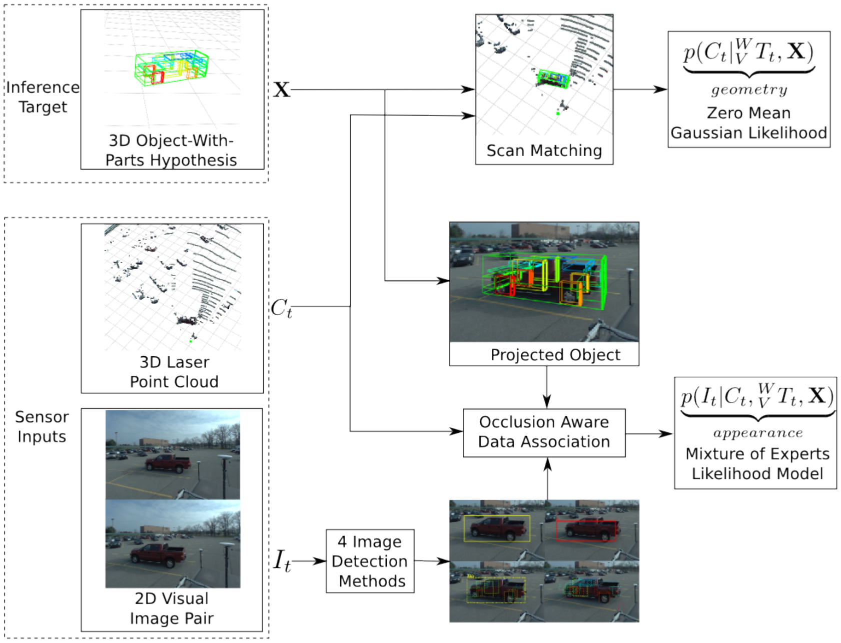

Inference Target
3D Object-With-Parts Hypothesis
$\mathbf{X}$
Scan Matching
$\underbrace{p(C_t|^W_V T_t, \mathbf{X})}_{geometry}$
Zero Mean Gaussian Likelihood
Sensor Inputs
3D Laser Point Cloud
C_t
Projected Object
Occlusion Aware Data Association
$\underbrace{p(I_t|C_t, ^W_V T_t, \mathbf{X})}_{appearance}$
Mixture of Experts Likelihood Model
2D Visual Image Pair
I_t
4 Image Detection Methods

Figure 8.1: 3D Parts Model Overview: A schematic overview of the components that make up the scene understanding model described by this chapter.

bilistic inference approach with the ability to reason jointly about all of the objects – a task known as *scene understanding*. In this framework, 3D and 2D overlap can be enforced by considering cross-terms in the object likelihood during inference rather than leaving this step until post-processing. The cost of joint modeling is a higher-dimensional state space. We have utilized the Markov chain Monte Carlo (MCMC) [MU49] method to recover likely configurations of objects within this space.

A primary goal of the work in this chapter is to show that modeling meaningful object parts such as the wheels, doors and windows of automobiles improves the recognition of objects within densely packed scenes with significant occlusion, such as parking lots. Our results show some instances where occluded objects can be understood from only a single image or a short sequence based on few visible parts. This demonstrates that the inference method correctly relates object parts to the 3D world.

8.1.1 Relation to Generic Multiple Viewpoint Object Model

The method discussed in this chapter relaxes several assumptions made in previous chapters and represents the most complete instantiation of the generic multiple viewpoint 3D object model that we will attempt within this thesis. Using the common terminology outlined in Chapter 5, our model has the following properties:

1. *State space*: a fixed number of objects with category label, location, scale and orientation. Objects have 3D parts, but these are fixed to the object's frame, thus they do not increase the effective state space.

2. *Evidence*: images and point clouds collected by a car driving through a crowded city.

3. *Object Prior*: category-specific object scale, mutual exclusion constraints between objects in 3D and ground plane layout.

4. *Appearance Likelihood*: a relation between 3D objects and four types of image space appearance models, some including object pose estimates and detailed part information. Impact weighted based on occlusion.

5. *Geometry Likelihood*: an occlusion-aware scan-matching term that compares the object's surface to sensed ranges.

6. *Learning Method*: individual learning procedures for a variety of visual detector types.

7. *Inference Method*: MCMC in the space of a fixed number of 3D objects.

8.1.2 Statement of Collaboration

The method described in this chapter has been created recently by the author of this thesis, with supervision and assistance from Michael Stark, from the Stanford University, and Bernt Schiele, from the Max-Planck Institute. The model was inspired by a joint insight of all collaborators: that the part-aware recognizer developed recently by Dr. Stark, [SGS10] had a strong potential to be used in 3D multiple viewpoint object modeling. While the project was a collaboration with Dr. Stark, who was also directly contributing new components, the author of this thesis had primary responsibility for: design of the occlusion-aware

3D parts models, adaptation of the basic inference tools to multiple viewpoint 3D recognition, implementation of the mixture-of-experts probability model, implementation of the data association method, interface with the Ford Campus dataset [PME11], executing experiments and analyzing performance of the technique. We made use of several software tools and libraries that were developed at the Max-Planck Institute, Computer Vision and Multi-modal Computing group by Dr. Stark, Christian Wojek, and Mykhaylo Andriluka. The material presented in this chapter has not yet been published in a refereed venue in this format, although several tools developed during this work have been used to support [SKP$^+$12]. The material described in this chapter will be submitted in the near future.

8.1.3 Outline

The remainder of this chapter will present the details of our approach and discuss its results. We will begin by describing how the generic probabilistic object model from Equation (5.11) is modified to allow reasoning about objects with 3D parts and multiple types of detection evidence. We will then describe our use of an MCMC-based inference technique to recognize objects using the model. Finally, we will describe experiments that validate our approach on data collected by a vehicle moving through real parking lots and street scenes in a busy city.

8.2 A Scene Model with Detailed Object Parts

Our goal is to recover the hidden state describing the set of objects present in a scene, $\mathbf{X} = \{O_1, ..., O_i, ...\}$. Each object is described by a category label, O_i^c, and an *oriented bounding volume* spatial representation. That includes centre position, $O_i^{cent} = \left[O_i^x, O_i^y, O_i^z\right]^T$, scale (composed of length, width and height), $O_i^S = \left[O_i^l, O_i^w, O_i^h\right]^T$ and a 1D orientation composed only of azimuth, $O_i^{ori} = \left[O_i^{yaw}\right]^T$. We neglect full 3D orientation estimation since cars nearly always rest flat on the ground and are rarely on their sides or vertical.

In addition, we model spatial information for a set of thirteen 3D parts that make up each object. Each part corresponds to a semantically meaningful sub-component of an automobile (e.g., doors, windows, bumpers, wheels). The top-left of Figure 8.1 illustrates our part layout. Each of these parts has unique location and scale, measured relative to the entire object's shape. We do not allow each part location to move independently. Rather, we fix the parts spatially within the object's outline based on the mean values for the object category *automobile*, which we computed using a number of accurate Computer Aided Design (CAD) descriptions labeled with part information. Fixing the part information makes our model less adaptable to variation between object instances, but it also greatly reduces the size of the state space. 3D part information still has great utility in our model. Rather than the image space partial templates that were describe previously in Chapter 7, defining object parts in 3D allows projection into each of the multiple available images. In this way, part information can provide valuable constraints on the likely position of the entire 3D object, as we will discuss below.

The evidence observed by the platform's sensors over a trajectory, Z^t, is a sequence of images, I_t, and point clouds, C_t, similar to those that we have described previously. In the autonomous urban driving setting that we consider here, high data-rate laser scanners such as the Velodyne HDL [Vel07] are commonly used. We assume that the sensors are calibrated with respect to the vehicle's frame, which is done highly accurately for the experimental data that we will consider later.

The relationship between observations and hidden states is expressed using Equation (5.11), which we repeat here for clarity:

$$p(\mathbf{X}|\mathbf{E}) \approx \underbrace{p(\mathbf{X})}_{object\ prior} \prod_t \underbrace{p(I_t|C_t, {}^W_V T_t, \mathbf{X})}_{appearance} \underbrace{p(C_t|{}^W_V T_t, \mathbf{X})}_{geometry} \underbrace{p({}^W_V T_t|\mathbf{X})}_{registration}. \tag{8.1}$$

Each term is described below, except the registration component, which we omit as we have done throughout this thesis. Autonomous driving vehicles typically posses sophisticated inertial measurement units and we have used data of this type to perform our experiments. A major difference between Equation (8.1) and our previous formulations is that it expresses a distribution over a set of objects, rather than a single object, o, as we had done previously. This has implications for each term in our likelihood, as they can account for dependences between objects and for our inference approach, since it must reason about a state space with a larger number of dimensions. We will continue by describing each component of the model.

8.3 Scene Prior

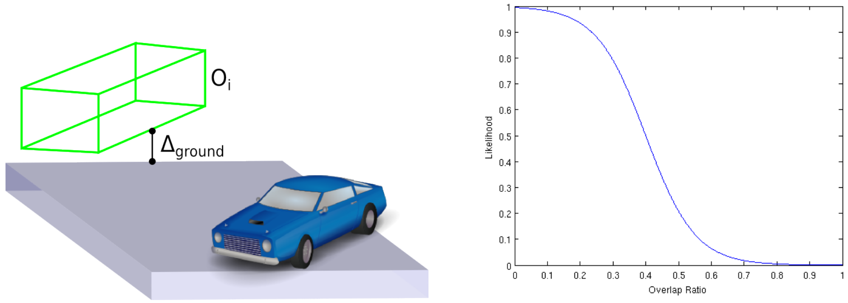
For the first time in this thesis, we attempt to express the constraints that exist across an entire set of objects as an informative prior within the scene inference process. While this prior could be learned from training data, no appropriately labeled training information was available for the parking lot scenes that we study. Recall that our method would need all of the 3D automobiles in a scene to be labeled accurately in a consistent 3D coordinate frame. Our prior, $p(\mathbf{X})$, is formed as the product of three intuitive probabilistic constraints. Our first constraint encodes the domain knowledge that all cars rest upon the ground, which we assume to be roughly planar in the area around our data collection vehicle. Second, we model a physical mutual exclusion constraint, enforcing that no two cars may occupy the same volume in 3D. Finally, we model the expected length, width and height of automobiles. We will continue by describing how each of these intuitive constraints is formulated in our probabilistic model.

8.3.1 Ground Plane Constraint

An intelligent automobile is typically aware of the location and orientation of the ground beneath its wheels (e.g., through system calibration). Further, we assume the environment surrounding the vehicle is nearly planar, so that extending the plane local to the data collection vehicle is a good approximation for the ground throughout the scene. As depicted by Figure 8.2, we compute the error between the height of the bottom of each inferred vehicle and the estimated surface of the ground. The liklelihood of the object's position is computed as a zero-mean Gaussian on this error. Objects perfectly resting on the ground plane have zero distance and are most likely. Floating or subterranean objects are penalized equally. Each object contributes a single ground-plane term to the overall scene prior.

8.3.2 3D Object Overlap Constraint

Prominent vehicles with strong appearance signatures can lead to multiple strong image space detections. Without regularization, our model can explain this data with multiple 3D objects in nearly the same position, one for each detection. In order to prevent this, we compute 3D overlap between all pairs of hypothesized objects. A one-minus sigmoid penalty function, as shown in Figure 8.2(b), for each pair contributes a likelihood of near 1 for objects without overlap and near 0 for completely overlapping objects. The scene prior



(a) (b)

Figure 8.2: Geometric Constraint Illustrations: (a) The ground plane object prior is illustrated. The blue car represents our data collection vehicle, which rests on a plane defined perpendicular to its up direction. Hypothesized automobiles, O_i, are most likely when they rest on this plane with $\Delta_{ground} = 0$. (b) The sigmoid likelihood function used to penalize overlapping objects.

contains $n(n-1)/2$ pairwise overlap terms when reasoning about a scene that contains n objects.

8.3.3 3D Size Prior

The likelihood of the length, width, and height of each hypothesized automobile is computed based on a distribution derived from a large set of CAD data for automobiles. Our 3D size prior is a product of three Gaussian distributions, one for each size dimension. Each object contributes one such term to the overall scene prior.

8.4 Appearance Likelihood

We extend the *tracking-by-detection* model, which relates 3D objects in the world to the visual appearance of images through the proxy of detections from an object recognizer. In order to reason about object pose and parts layout, our model includes up to four image space object recognizers including: 1) the DPM [FGMR10] approach that we have used in previous chapters; 2) a viewpoint-aware extension of DPM that we will call *DPM-bank*, inspired by the work of [BS11] who used such recognizers previously on the experimental dataset we consider; 3) the object-part constellation model of [SGS10] (see Chapter 2 for a detailed description); and finally 4) the output of each individual part detector from

[SGS10], treated as an independent recognizer (i.e., discarding dependence between parts). Each of the four recognizer-types produces detection evidence for every image, but each contains a unique set of additional spatial information. All recognizers but the basic DPM model predict the pose of the object in image space. Part information is present in the results of both the constellations and independent part detectors. We have extended both the data association and mixture-of-experts likelihood portions of our generic multiple viewpoint 3D object model in order to accommodate these properties.

As mentioned, we extend the visibility-weighed mixture-of-experts likelihood. This model explains the appearance of an image, I_t, taken at time, t, based on the hypothesized 3D objects. Each object can be represented in image space by projection through the matrix P_t, which combines registration information, ${}^W_V T_t$, and the known camera calibration (not expressed symbolically here). Instead of raw image pixels, we reason about detections from several detector types, p. Each object, O_i, is associated with up to one detection of each type in each image. We express each associated detection as $d_i^{pt} \in D^p(I_t)$, and all detections that are associated to a particular object in and image as $A(O_i, D^*(I_t), {}^W_V T_t) = \{d_i^{1t}, ..., d_i^{mt}\}$. While expanding our appearance likelihood, we assume data association has already been determined. The next section will give details on the association procedure and we will discuss how association and likelihood computation are interleaved during discussion of our inference method. Here we combine terms contributing to the visibility weight of each object part and express them as w_i^{pt}. This term is the product of v_i^{pt}, the ratio of visible pixels of each object part as derived from the sensed point cloud C_t, and an indicator function that allows us to ignore heavily occluded parts, $\delta(v_i^{pt} > \theta)$. The overall probabilistic relation is expressed as

$$
\begin{aligned}
p(I_t|O, C_t, {}^W_V T_t) &\approx \prod_i p(d_i^{1t}, ..., d_i^{mt}|O_i, C_t, {}^W_V T_t) \qquad (8.2) \\
&= \prod_i \frac{1}{\sum_p w_i^{pt}} \sum_p w_i^{pt} \Psi_{sp}(d_i^{pt}) \Psi_{gp}(P_t \cdot O_i, d_i^{pt}). \qquad (8.3)
\end{aligned}
$$

Equation (8.2) expresses the approximation that the image space detections are independent of one another and conditionally independent of all unassigned objects, given O_i, the one to which they are assigned. While we know that nearby objects have an impact on each other's appearance (e.g., they may mutually occlude or cast shadows), this reasoning is neglected at this level, to maintain a tractable liklelihood function. Note that other portions of our model (i.e., the object prior), do describe interactions between different objects.

Equation (8.3) describes the mixture-of-experts model that combines information from all detector types to produce a score for each object. Recall that Ψ_{sp} is a potential function over object scores and Ψ_{gp} is a potential on the geometric agreement in image space between the detection and the projected object. $\delta(v^{pt})$ is used to completely discount objects and parts near to full occlusion. Ψ_{gp} takes the same form as we have described previously. That is, we compute the scale-normalized center agreement as well as the agreement in scale and produce a likelihood with a zero-mean Gaussians on each term. The object that best describes a detection is the one which precisely agrees with its bounding box when projected into the image. Deviations are penalized with lower likelihood scores. When an object cannot be associated to a particular detector type, the corresponding terms in the mixture are replaced by a penalty related to the likelihood of a recognizer not responding at all when an object is present. This data-driven quantity is estimated during validation of the model and is different for each evidence type.

In Equation (7.4), we previously used a similar formulation to reason over whole-object detections and four half-object types, so p took five unique values. Here, we have simply expanded the set of detector-types. In our complete model, p takes up to sixteen values: one each for the three whole-object detectors and one each for the thirteen part detectors. A notable difference in Equation 8.3 from the previous formulation is that both potential functions, Ψ, are indexed by the detector type, p. Unlike the previous chapter, where we applied the same model for each object part, the four distinct types of recognizer used in this chapter have a wide range of response characteristics. It was important to model this property within our formulation.

We will continue by describing the procedural data association method that is required to determine which detection evidence should be used to compute the likelihood for each object.

8.4.1 Data Association Method

The *tracking-by-detection* method requires detections from image space object recognizers of various types to be related to projected 3D objects with the association function, $A(O_i, D^p(I_t), {}^W_V T_t)_t = d_i^{pt}$. In this chapter, we attempt to relate objects to the outputs of several object recognizers. Each object possesses both an orientation and parts and these elements must be considered in the data association function. As before, we assume one-to-one matching between objects and detections from each recognizer (i.e., that no detection is explained by more than one object and that no object explains more than one detection of each type). This one-to-one constraint is also applied between 3D object parts and image space part detections.

We consider orientation when associating to recognizers that predict viewpoint. This means, for example, that if the 3D object is oriented such that the robot should see a side-view in image I_t, our data association procedure will not associate detections that are labeled as front or rear views. The registration information in our system allows the world-space object orientation to be related to each view as a *viewing azimuth* in the same coordinate system as the recognizers. Knowledge of the discretization of each individual recognizer allows the formation of equally spaced bins in orientation space around the predicted viewpoint. We only associate detections that agree with our hypothesis, within a threshold on the number of these angle bins.

Association of image space part detections to the parts of our 3D object hypotheses is performed in a similar fashion. One difference in implementation is that the part information is expressed relative to the entire object's location and scale, rather than directly in the global frame. So, the projection of parts to form a bounding box in each image is a two-stage process: 1) we map a part to the global frame by composing the rigid-body transformation of the part with the entire object's transformation; and 2) we project that global-frame information into the image and form a bounding box. Each part has a specific label (e.g., right-front wheel), and matching is only done between image space detections and 3D parts of the same label. Note that each part also possesses an orientation, and that our part detector estimates orientation. We apply the same orientation constraint to parts as to whole objects: the detection and hypothesis must agree within a threshold in order to be associated.

The object recognizers that we consider often produce many near-duplicate responses in image space for each true 3D object (see [HCD12] for a recent empirical study of this effect). We have found that when observing multiple mutually-occluding objects, such as a line of parked cars, it is important for the scene understanding system to choose only one out of the potentially many nested detections around each object. Therefore, when assigning a detection and removing it from further consideration, we disallow subsequent assignment of all other detections that completely cover or are completely covered by the assigned detection. This ensures that each detection in the set of finally associated detections contributes at least some small number of new pixels to the portion of the image that is explained.

We have implemented a data association procedure that satisfies all of the above constraints and assumptions. A modified version of the *Greedy Matching* (Algorithm 1) is executed once per whole-object detector type and once for each part type. The result of the modified matching process is a mapping between objects and associated detections. This mapping is used to compute a mixture-of-experts appearance likelihood, as we have described.

8.5 Geometry Likelihood

The geometry likelihood term captures the agreement between the point cloud information collected by the vehicle's laser range finder and the hypothesized surfaces of 3D objects. Each proposed object suggests a depth along each ray within its extents. We efficiently compute the depth values at each 3D corner and linearly interpolate over the object's faces as rarely as possible (i.e., only in locations where a sensed distance is available). This efficient approximation makes a small error due to the interpolation occurring in $(x, y, depth)$ space which is non-linear. However, since the objects are typically far from the camera and the error in depth sensing is fairly large, the magnitude of the projective effect is negligible.

The point cloud measured by the laser range finder suggests depths for a sparse set of pixels. The laser has a much coarser sampling resolution than our images and this resolution varies with distance. So, the number of sensed depth values per object can range between ten and several hundred. In a similar fashion to the previous chapter, each depth value is considered as one of three discrete outcomes: 1) near, 2) unoccupied or conflicting, 3) occluded. A zero-mean Gaussian is applied to the depth differences (i.e., difference between the laser-sensed depth and the hypothesized object surface), only for un-occluded regions. As before, where the object is occluded, the depth errors are not counted within the geometry score. The final value is normalized for the number of visible points observed. We have previously attempted an alternative solution based on a median filter over sub-regions, as this is more robust to non-uniform sampling. In practice that technique did not improve performance. The geometry likelihood term in our overall probability model, $p(C_t|^W_V T_t, \mathbf{X})$

is the product of the likelihoods for each object, which again makes the approximation that the local point cloud evidence related to one object is conditionally independent of all other objects, for reasons of tractability.

We note that although we utilize a relatively detailed visual appearance model in this chapter, capturing the layout and appearance of 13 object parts, our geometric object model is quite simplistic. In part, this choice has been made following our guiding motivations to primarily pursue visual models (see Chapter 1). However, when a point cloud is available, we do aim to make best use of this geometric information. Our simple 3D bounding volume model of automobile geometry could be replaced by a more detailed geometric model, such as a triangle mesh that captures the layout of wheels, doors and windows. It is unlikely that a single model would generalize over the variety of shapes present across the automobile category, including cars, trucks and vans among others. A potential approach would be to fit an instance-specific shape template, such as the best-fitting CAD model from a learned database of samples, to each hypothesized object. This would potentially improve the orientation estimation of our approach at the cost of introducing another level of complexity and is left for future study.

8.6 MCMC-based Scene Inference

Our goal is to locate the set of cars that are likely to occur in each considered parking lot scene. This requires maximizing the likelihood that we expressed in the previous section, but direct optimization cannot be achieved due to the numerous dimensions and the variety of information sources that are considered. It is not even clear how to sample from our target probability, $p(\mathbf{X}|\mathbf{E})$, directly. However, we can compute the likelihood of a sampled set of objects, $p(X^{(i)}|\mathbf{E})$, with minimal effort. This allows us to utilize Markov chain Monte Carlo [MU49] sampling along with a simple auxiliary proposal function to produce a set of samples $\{X^{(i)}\}$ that approximates our desired distribution. This can be used to estimate the maximum.

We construct our Markov chain using the Metropolis Hastings [MRR+53, Has70] algorithm. This involves iteratively drawing samples from a proposal function that is based on the previous state, $q(X^{(i+1)}|X^{(i)})$. The acceptance ratio for each sample is

$$\alpha(X^{(i)}, X^{(i+1)}) \quad = \quad \min\,(1, \frac{p(X^{(i+1)})q(X^{(i+1)}|X^{(i)})}{p(X^{(i)})q(X^{(i)}|X^{(i+1)})}\,). \tag{8.4}$$

With probability α, the new sample is accepted and it is added to the chain. Otherwise, the sample is rejected, and $X^{(i)}$ must be used to draw a new proposal. In general, MCMC methods converge to the target distribution if the transition between states is aperiodic and irreducible [AdFDJ03]. For Metropolis-Hastings, these conditions are satisfied as long as the support of q includes the support of p [Tie94]. The chain is also typically biased by its starting state and so we must discard some number of initial samples in order to achieve a fair sampling.

We currently describe fixed-dimensional sampling, which means the number of objects must be set beforehand, or that numerous instances of inference are needed, one for each possible number of objects. See the discussion of future work for description of a method that simultaneously estimates the number of objects present.

The quality of the MCMC estimate and the number of samples required are highly dependent upon the choice of a proposal function and proposals that approximate the target distribution well through use of domain knowledge can often be more efficient than naive samplers. We alternate several proposals that leverage our 3D understanding of the recognition process and our detection evidence. As described by [AdFDJ03], such cycle proposals cause the chain to converge to p as long as each component individually meets the convergence criterion. The next section describes our data-driven mixture proposal. We will then describe how the results of the Markov chain are used to produce final estimates of the objects in a scene.

8.6.1 Bootstrapping

Our state space has many continuous dimensions but our likelihood is only meaningful in the small neighborhoods where objects are sufficiently near to detection evidence for some data association to occur. In sampling terminology, our likelihood is highly peaked. So, careful selection of starting states for the chain is important to achieve meaningful results – a process known as bootstrapping. We initiate a fixed number of objects in 3D by projecting rays through 2D detection evidence (see Figure 8.3). The bootstrapping procedure iterates the following steps:

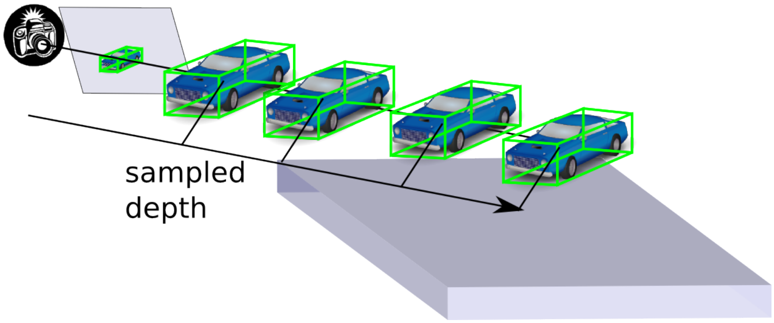


Figure 8.3: Sampling-along-ray Illustration: After selecting a 2D detection region, samples are proposed based on depth values drawn from a distribution capturing the point cloud data and/or the expected object's scale.

- Select a view uniformly at random.
- Select a detection within the view randomly, weighted by detection score.
- Cast a ray through the detection (deterministic vector math).
- Select a depth value randomly, based on a Gaussian distribution formed by the depth values within the bounding box and/or the expected object's scale.
- Add an object to the state space with initial centre and scale as determined by the constructed ray and sampled depth (deterministic vector math).
- Select an orientation for the object uniformly at random.

A fixed number of objects are initialized in this way, and the result is the first sample from the chain, $X^{(1)}$. Our procedure continues by computing the likelihood using Equation (8.1). This includes computing data-association, evaluating the object overlap priors, and performing depth interpolation, among other tasks. We continue by describing the diffusion and re-sampling proposals that allow the chain to explore new states.

8.6.2 Diffusion Moves

During the normal operation of the chain (i.e., after it is initialized), new states are considered primarily by altering one of the estimated quantities by a small amount. This process, known as diffusion, is implemented within our system by iterating the following steps:

- Select an existing object O_i uniformly at random.
- Select a dimension uniformly at random from centre, scale and orientation.
- Select a new value for that dimension randomly, based on a Gaussian distribution centered on the current value.
- Re-associate all objects to the detection evidence to account for inter-object interactions which may have shifted due to the changed values (deterministic procedure).

8.6.3 Re-sampling Moves

In principle, only diffusion moves are required to sample the entire space of scenes within our model. However, we have observed that, in practice, a chain based solely on this proposal requires many samples to move away from the starting position. To address this situation, we have added a re-sampling move that makes much larger steps in the continuous object state space. We select one of the existing objects for removal and randomly draw an object to replace it based on projection through the detection evidence. This procedure allows large moves to new regions. The details involve:

- Iterating over existing objects. At each re-sample step removing the next object in line and freeing its associated evidence.
- Performing the same steps described for bootstrapping in Section 8.6.1 to instantiate a single new 3D object from the free evidence.

This procedure is equivalent to a paired delete and add move. It keeps dimensionality constant, but moves a potentially large distance through the state space by selecting a new seed from the set of 2D detection evidence.

Our chain samples the space by alternating the diffusion and re-sampling moves. Many iterations of diffusion are performed in a row and we rarely re-sample. At each step, we compute p, q and the Metropolis Hastings acceptance ratio to decide if the new sample should be accepted or rejected. After many iterations, our chain will have explored the state space and its samples will approximate $p(\mathbf{X}|\mathbf{E})$. We will conclude the description of our inference method by describing several methods for deriving an estimate of the objects in the scene from a set of MCMC samples.

8.6.4 Producing an Object Configuration

Our overall goal is to infer the most likely set of objects in an environment. This could be achieved by taking the MCMC sample with the highest likelihood score. However, the evaluation that we will conduct is based on precision-recall computation that requires confidence scores for every individual object. We must choose how to assign these confidence values. One option is to assign the likelihood of the entire scene sample to each individual object. Some other external re-scoring approach may also be considered. For example, a state-of-the-art approach in 3D scene understanding for pedestrians within street scenes [WWRS11] describes a procedure for transferring the confidence estimates from MCMC back to original bounding boxes from the image space recognizer, as this removed any localization errors introduced by the projection process.

We have attempted several variants both for the technique used to extract objects and also for the method used to score the resulting objects. We have found that, in practice, using $p(\mathbf{X}|\mathbf{E})$ to score all objects equally does not provide strong performance on our evaluation metrics. The localization performance of the 3D objects for that method is strong, but when incorrect objects are given the same score as correct objects, the evaluation procedure discounts the positive performance. Our results are best when we utilize the 3D information from the MCMC sampling, but assign each object an individual score that is derived only from terms in the likelihood specifically produced by the object itself. That is, we do not include cross-terms from the prior. Putting this procedure in context, it means that the scene understanding portion of our technique is useful to ensure that we estimate correct object locations and to remove completely incorrect objects. However, since the evaluation is tailored to object recognition, the optimal object scoring is one that focuses only on evidence local to a single object.

We have now described a scene likelihood formulation that relates many objects in a scene to many types of visual and 3D evidence. The MCMC sampling technique allows us to locate sets of objects that are likely given this model. In combination, these components form a complete object recognition approach. We will now continue by describing our evaluation scenario and the results of our method.

This concludes the discussion of our inference procedure for the detailed-parts model. Algorithm 4 provides an overview of the approach, linking the individual components that we have described above. The next section will discuss the evaluation of our approach on

Algorithm 4: Detailed Parts Inference Algorithm

```
input  : Sensory data, $I_t$ and $C_t$, from each of $n$ robot positions
         Registration information, ${}^W_V T_t$, for each position
         Detections, $D^p(I_t)$, from each detector type in each image
output: A set of 3D objects, $X$
/* sample a random first scene (Section 8.6.1)                */
$X^{(0)} \leftarrow$ Bootstrap($D^p(I_t), C_t, {}^W_V T_t$);
$i \leftarrow 0$;
/* draw $n$ samples from Markov Chain                          */
while $i < n$ do
    /* propose with diffusion, Section (8.6.2)                 */
    /* or re-sampling, Section (8.6.3)                         */
    $X^{(i+1)} \leftarrow$ dataDrivenProposal($X^{(i)}$);
    Compute $p(X^{(i+1)})$ using GreedyMatching (Algorithm 1) and Equation (8.1);
    if $X^{(i+1)}$ satisfies Metropolis-Hastings criterion, Equation (8.4) then
        /* add $X^{(i+1)}$ to chain, increment $i$              */
        $i \leftarrow i + 1$;
    else
        /* reject $X^{(i+1)}$, do not increment $i$             */
/* choose sample maximizing Equation (8.1)                     */
$i^* \leftarrow argmax_i(p(X^{(i)}|E))$;
/* produce final answer, Section 8.6.4                         */
return finalize($X^{(i^*)}$);
```

realistic urban driving data.

8.7 Experimental Setup

We have evaluated our scene understanding approach on a sub-set of the the Ford Campus Vision and Lidar dataset [PME11]. This dataset was collected by mounting a sensor suite on-board an automobile traveling through a busy urban setting in Dearborn, Michigan. Images were collected with an omni-directional camera. They have been rectified to remove distortions from the spherical lens, and the camera has been accurately calibrated. A Velodyne HDL [Vel07] captured detailed point clouds at each location. Registration information was obtained using a high quality inertial measurement unit (IMU), which yields registration between images that is accurate to the pixel level for short trajectory sequences. Overall, there are roughly fifty thousand rectified images (each spherical omni-directional frame is split into 5 square sub-images) and ten thousand laser scans contained in the Ford Campus data.

The data collection vehicle's trajectory covers several kilometers. GPS information allows this trajectory to be defined roughly in a geo-referenced coordinate system, but for our purposes a world coordinate frame is defined to be co-incident with the vehicles base frame at the start of the trajectory. Accurate IMU information allows all sensor positions to be referenced back to this world frame. We also define all of the 3D information that we model and infer based on this frame.

The subject matter captured in the Ford Campus data includes numerous parking lots and densely occupied street-parking regions as well as long stretches of travel on open road where cars occur less frequently. This entire trajectory is far too large to be processed as a single entity, in particular since any single car is only visible for several neighboring frames. The authors of the Semantic Structure from Motion (SSFM) technique, Bao *et al.* [BS11], who previously studied the problem of recognizing cars in the Ford Campus data, have selected several of the most interesting sub-sequences of data, which capture the most crowded parking scenes. They have annotated the automobiles present in these sub-sequences and defined a protocol for testing automated perception results against these annotations. We refer to this as the *Test Pairs* evaluation protocol and will continue by describing the visual task that they have outlined in the following section.

8.7.1 Test Pairs Evaluation of Bao *et al.*

The authors of [BS11] have selected nine *scenes*. Each scene is defined to be the environment visible from a sequence of nine sequential vehicle positions. The timing of image capture in the Ford Campus data is roughly fifteen frames per second so each scene corresponds to roughly 0.6 s. Although the vehicle's speed is not constant, the maximum baseline between camera positions for the same scene is on the order of 3 m. Four scenes are designated for training and five are provided for testing. All of the results we describe below correspond to data only from the test scenes. From the test scenes, 354 *image pairs* have been selected via quasi-random sampling from all scene images, with the constraint that there must be a large overlap between the two images within a pair (satisfied for two views from the same camera at nearby times or views from neighboring cameras at the same time). This pair list, along with the provided 2D annotations, defines the evaluation protocol that the authors of [BS11] suggest. The 3D scene inference algorithm is meant to consider each pair independently, compare its hypotheses with the annotations and report results. That is, even though 2 unique pairs are drawn from the same scene, information should not be re-used but instead the process should restart from scratch for each pair. To conform with the standards of the research community, we have executed our tests exactly in this fashion.

Figures 8.4 and 8.5 provide a sampling of the images that define the *Test Pairs*. Not all images are shown from each scene due to space constraints. Note that many cars are occluded in each scene. Also, the scale of automobiles in the images is typically quite small. Overall, the visual task of recognizing cars in these test images is difficult relative to other benchmark tasks, such as recognition in the UBC VRS data, the PASCAL VOC challenge and other standard contests.

We have obtained the publicly available object annotations created by [BS11]. These include 2D bounding boxes for the visible cars in each image. We have augmented the author's evaluations by annotating many of the smaller and more highly occluded cars which were omitted from their data, as well as by correcting some of the existing annotations to make them pixel-tight to the object content. Additionally, we have used the annotation software suite developed for the UBC VRS dataset (see Section 4.2.3) to label each 2D annotation with occlusion information and also to create 3D object annotations in the coordinate frame of each scene.

8.7.2 Detector Models and Training

The four types of image space object recognizers were trained in an appropriate fashion for recognizing cars within the Ford Campus data. All of the visual appearance models that we consider are data-driven and must be presented with a number of labeled examples during their training phase. The DPM appearance model was learned using the car training set of the PASCAL VOC 2009 contest data that is publicly accessible. Note that this is a standard data source, and the DPM model is a top performing method, as we have previously mentioned. This makes the DPM input a notable base-line for comparison of performance on the 2D object localization task.

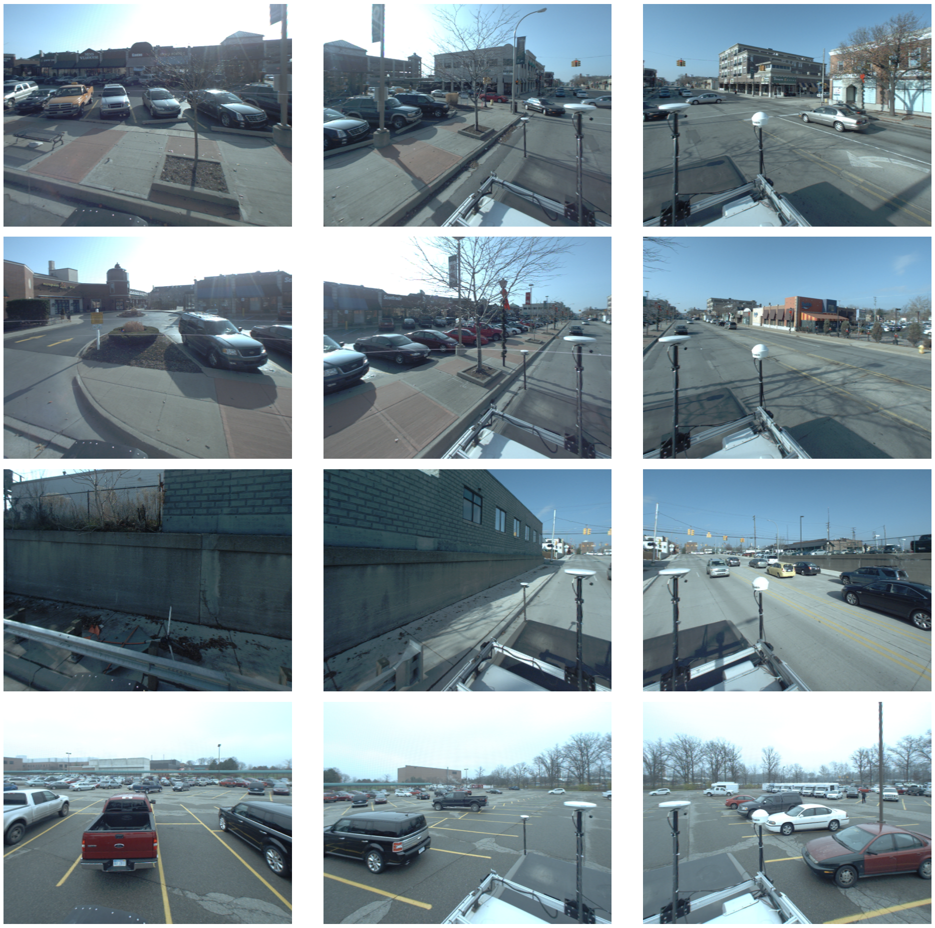


Figure 8.4: SSFM Dataset Training Examples: Camera views 1, 2 and 3 from left to right and scenes 5, 6, 7, and 8 from top to bottom.

The viewpoint-aware DPM-bank refers to a model where numerous different DPM models are trained, each only based on positive examples from a specific viewpoint. The so-called bank of detectors are all run on each image, and the highest scoring viewpoint model at each location is returned. Training DPM-bank requires both bounding box annotations as well as viewpoint labels. This information is provided by the Multi-View Object Categories database of Savarese *et al.* [SFF07], which is used for our experiments in Chapter 6. The objects in this dataset are given one of eight different viewpoint labels. Hence, the DPM-bank provides viewpoint estimates at a resolution of 45°.

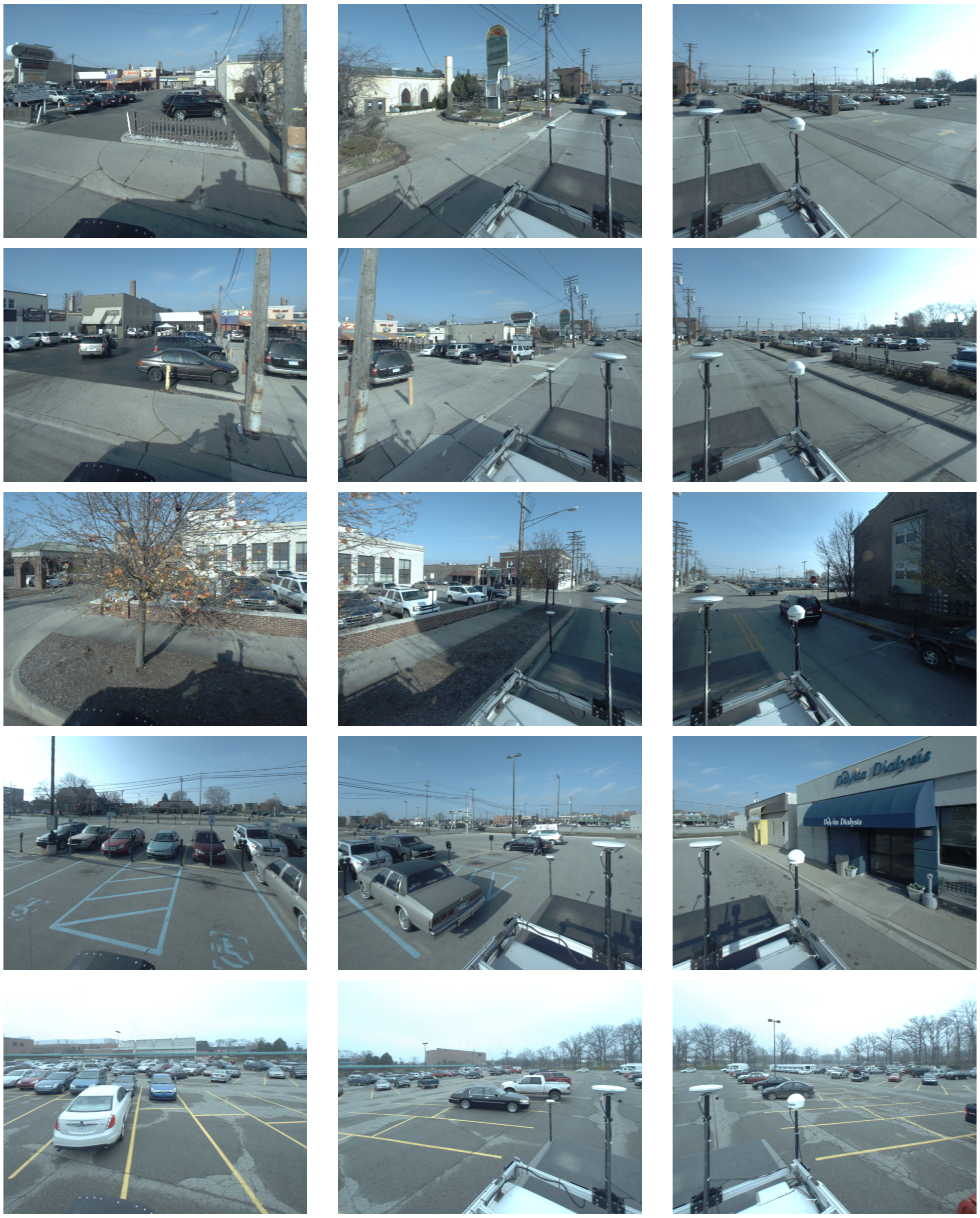


Figure 8.5: SSFM Test Examples: Camera views 1, 2 and 3 from left to right and scenes 1, 2, 3, 4, and 9 from top to bottom.

Viewpoint-aware part-constellations based on [SGS10] have been trained from CAD data with part labels. As part of the training process, the inherently 3D CAD information is rendered into one of 36 horizontal orientations as well as a number of different elevations. The constellation model learns the appearance of the entire object, as well as that of each of its parts, in each of these viewpoints. Note that because the labeled parts are rendered accurately as part of the training process, the learned part templates have highly constrained physical meaning. That is, they represent a consistent 3D part of the object. The output of the viewpoint-aware part-constellation is a bounding box with detection confidence and viewpoint prediction at a resolution of 10° and a collection of 13 part hypotheses nested inside the full object detection. For the constellation detector, each part shares the viewpoint prediction of the whole detection, however the part locations vary based on image content.

Single part detectors are trained in a similar fashion to that we have just described for the constellation models. CAD data is rendered into a number of viewpoints to form synthetic images with labeled object parts. The difference from the previous approach is that, instead of capturing the spatial distribution of the part within the object region, the job of the constellation, the single part detectors treat each part independently. Viewpoint information is still available. So, the response of each individual part detection is a bounding box for one single part with a viewpoint prediction accurate to 10°. One can imagine that the responses of single parts often closely agree with the responses of the constellation model. This is particularly true for objects that are completely visible and large, but noticeable differences exist both for objects that at a small scale within images and also for occluded objects. For small objects, the entire constellation is much more likely to respond, because it is able to pool visual information from all available pixels, while each individual part may provide too little information for the independent detector to be confident. However, for occluded objects, the entire constellation faces a large amount of negative evidence, while the single part detectors for the visible portions of the car (e.g., the one visible wheel), may still respond strongly.

8.8 Experimental Results

We have executed our 3D scene understanding approach on the *Test Pairs* evaluation protocol. This section will analyze its results qualitatively, by providing sample images, and quantitatively in both 2D and 3D.

8.8.1 Qualitative Performance

Figure 8.6 demonstrates several typical results obtained by our scene understanding method on the Ford Campus data. Our method is typically able to locate and label a number of the foreground automobiles reliably. The results for smaller background objects are less reliable, but there are many instances where small and occluded objects are still located correctly by the method. The 3D localization accuracy can be judged in the figure by comparing the blue hypothesized 3D volumes to the green annotated ground truth, as well as to the point cloud displayed in the background.

Overall, it is visually clear that our system has a basic ability to make reasonable guesses about object locations, but that there are numerous errors. Our method is often able to locate a reasonable object centroid while the orientation has large error (e.g., Figure 8.6 second, third and fourth rows). We have examined several such cases and found that these objects are primarily explaining detections from the DPM recognizer that has no viewpoint information. In these cases our method can only obtain viewpoint cues from its weak geometric model. Our viewpoint-aware recognizers all achieve less recall than the basic DPM approach, which leaves our method with impoverished pose information for some objects. An exception to this explanation is the nearest car in the third row. The car is large and prominent and although our method correctly recovers the 3D position and associates to detections from all four input-types, the hypothesized 3D orientation is far from correct. In this case, a single uncommonly confident DPM-bank result with the incorrect viewpoint is the cause for the error. A larger quantity of viewpoint-annotated training data may be required to remove situations of this type.

Other failure-modes of our approach include incorrectly hypothesizing multiple nearby 3D objects for a single true car position, as seen in the fourth row of Figure 8.6. Although our object prior and data association method both have components to discourage this type of explanation, in some cases multiple detections in the image space detection evidence strongly support such an object layout and our method is unable to recover. Finally, the third row of Figure 8.6 provides an example of what is perhaps the most universal failure in our approach – missing recall on automobiles that are far from the camera. These automobiles typically have lower scoring image space detections. At present, our detection score potential, Ψ_{sp}, does not account for the fact that smaller objects are likely to have lower scores, although this is certainly a trend that we could learn from the validation data.

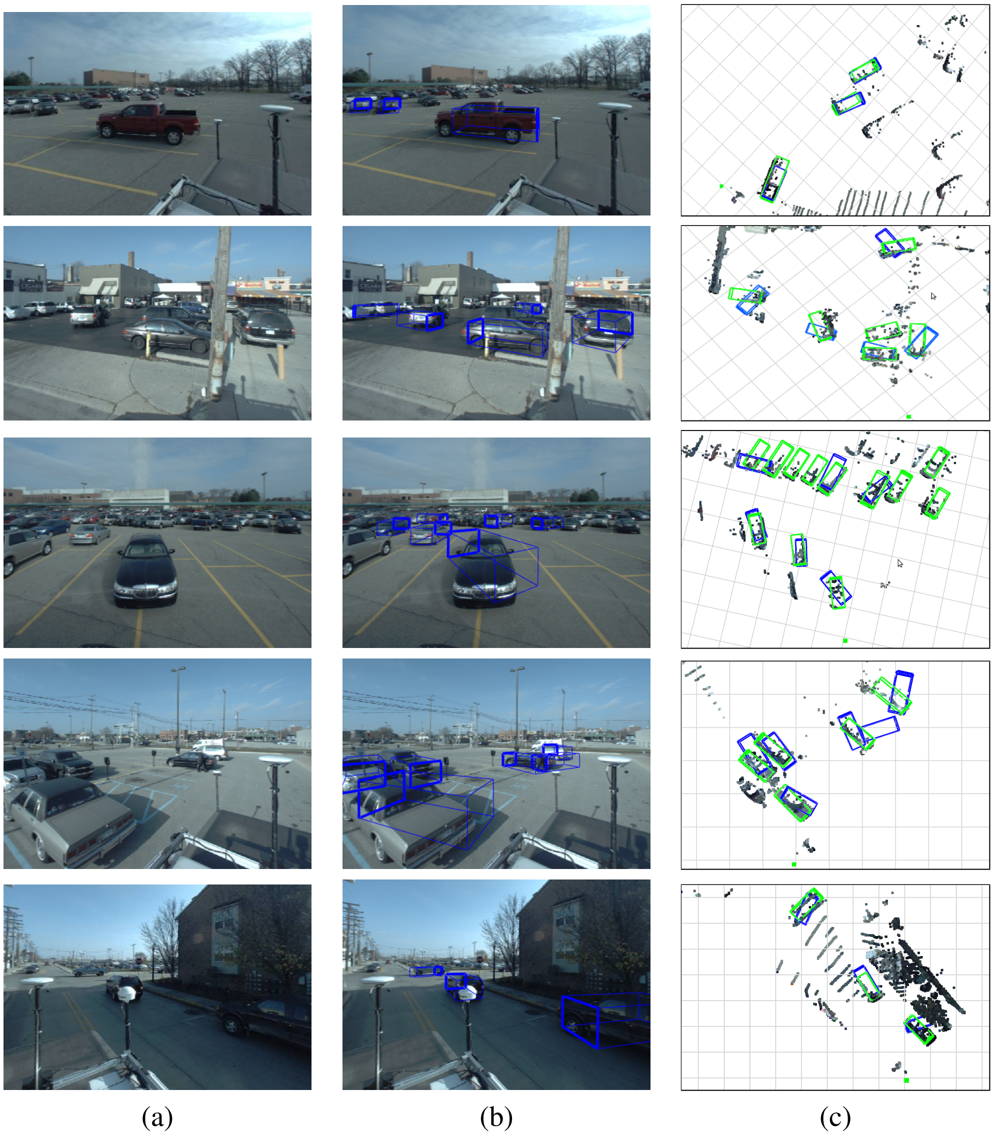


Figure 8.6: Qualitative Results for 3D Parts Model: In each scene, the results of our inference are thresholded to 80% precision. One image from the input pair is shown in (a). In (b), inferred objects are projected into the image and overlaid in blue. The 3D scene components, including the point cloud, inferred objects (blue) and ground truth objects (green), are rendered from a top-down view in (c).

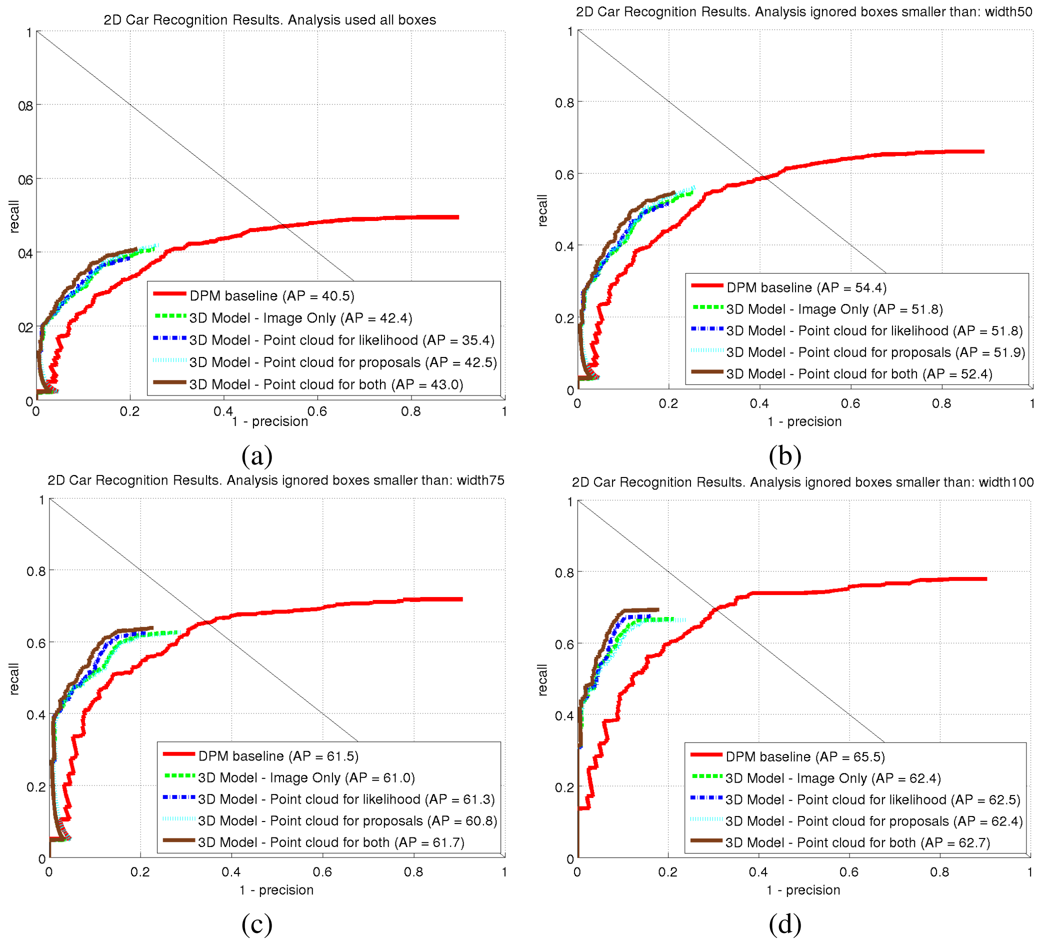
We plan to pursue this avenue in future work. We will continue by analyzing the results quantitatively.

8.8.2 2D Detection Evaluation

Next we compare the effectiveness of our method at detecting objects in 2D by projecting our scored 3D objects into each image and conducting PR evaluation (see Section 2.6 for details). Our goals were to compare the performance of the 3D scene understanding approach to a standard baseline to validate its competitiveness to the state-of-the-art. Figure 8.7 provides comparison between our method and the viewpoint-agnostic DPM baseline, which is exactly the recognizer that placed first in the PASCAL VOC 2009 for the category "car". The trend is that our method achieves higher recall than the baseline in the high-precision region. This means that it correctly reports a high confidence for more of the objects. In terms of total recall, the 3D scene understanding approach recognizes slightly fewer overall objects. As discussed during our qualitative analysis, these are mainly the smallest objects in the test set with insufficient visual evidence. The DPM approach, which takes its strongest cues from the object's outline, is able to outperform our method for small objects.

We have run the inference procedure with and without consideration of the point cloud evidence during both our sampling and likelihood computation. The trend is that the point cloud is generally helpful and that our method performs the best when it has access to this sensed range information at all stages. The improvement in performance based on the point cloud data and occlusion reasoning is not as large, relatively, as we saw for kitchen scenes. While we are still investigating this factor, the lower resolution of the Velodyne data is a potential explanation. It is informative to look at the results labeled "Image Only", as this represents the performance of our method with only a pair of visual images and the corresponding detections results. The fact that our method still performs well in this scenario means that it is able to recover 3D semantic information only from the registered pair of images.

The four different plots in Figure 8.7 differ in the minimum scale of annotations that were used for analysis. When all boxes are included (top-level graph), a method is required to correctly recall every annotation. The three remaining graphs represent results with increasingly many of the smallest annotations discarded. The trend in the high pre-



(a)

(b)

(c)

(d)

Figure 8.7: 3D Parts Model Results: Precision and recall curves for variants of our method and DPM, a 2D detection baseline. Each graph represents evaluation with a different minimum annotation size, measured in pixels of width, ranging from 0 in (a) to 100 in (d).

cision region is that our method's improvement over the DPM baseline is most prominent for large objects. This is likely due to the fact that we employ more detailed recognizers whose performance gives the greatest contribution for these large objects. This analysis also suggests that performing recognition on higher-resolution images, which are increasingly available as optical hardware improves, would help our method to do well over larger portions of these scenes.

8.8.3 Pose Estimation

We also evaluate our method's ability to correctly determine the azimuth angle of each recognized vehicle. We compare annotated 3D regions which have had azimuth carefully labeled by a human against each of our object hypotheses. As was discussed in Section 2.6, there is a necessary matching step in this evaluation, since our system estimates both 3D position and viewpoint. We must determine which inferred 3D region corresponds to each 3D annotation. We first project both the ground truth and the inferred objects into 2D, but record the view-specific orientation estimate for each bounding box. We then match between annotations and hypotheses with the PASCAL VOC evaluation protocol. Finally, for all true positive 2D object locations (those that match best with overlap over a threshold), we compare the estimated azimuth to the annotated value.

We perform two types of evaluation of the pose estimation errors. First, we consider a viewpoint classification task, where we bin angles into eight 45 degree bins and observe how often our method labels the correct bin. Chance level for this task is an accuracy of 0.125 and optimal performance is 1.0. We visualize these results with confusion matrices. Second, we measure the raw error, in degrees, between continuous annotated viewpoint and the inferred value, where 0 error represents optimal performance. We display the results as a histogram. Figure 8.8 illustrates the results of each of these evaluations, for four variants of our approach where different combinations of the input detection evidence are considered. Note that, due to the two-stage evaluation required, each approach is being evaluated on a different number of true positive (TP) detections. This must be considered when comparing relative performance of different methods. A better recognizer will recover a larger number of true positives, but this set of objects may include more distant objects whose pose is harder to recover.

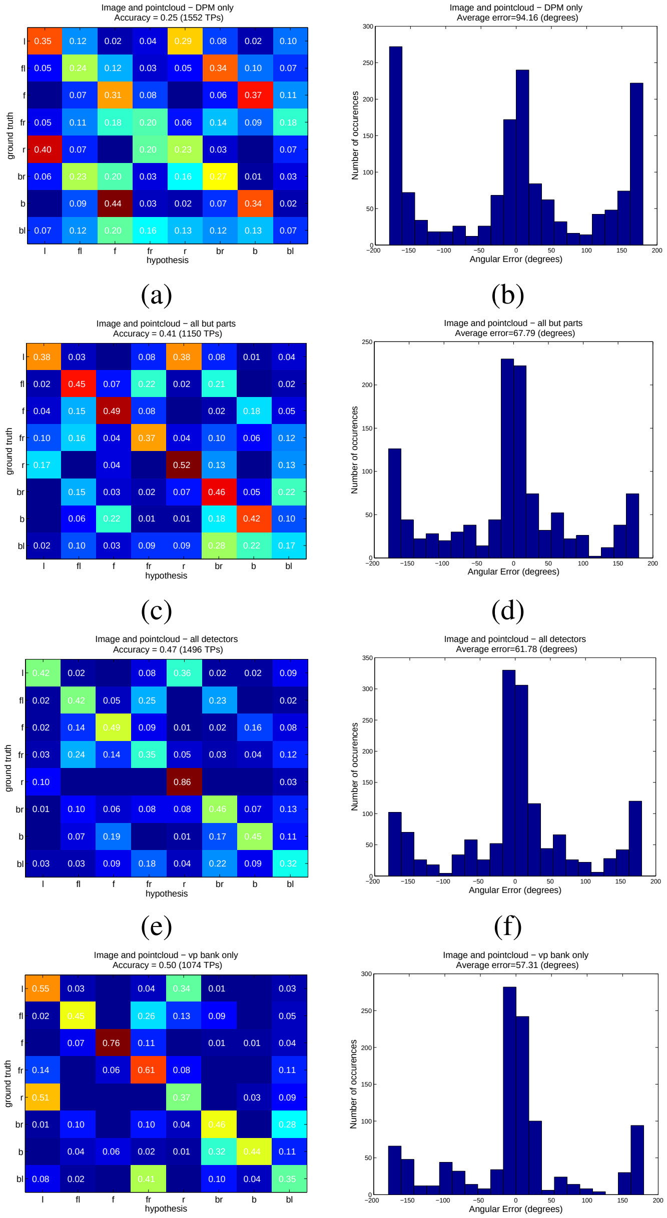
In all cases, it is apparent that our approach favors the correct object pose, however there is a wide range between methods. Adding detector types that include pose estimates improves the ability of our 3D method to estimate pose. Notably, graphs (e) and (f) show a strong improvement over (c) and (d), since there are both more true positives recovered and the pose estimation accuracy is better. The only difference between these methods is the use of the independent object part detectors. This indicates that the object parts make a significant impact on our method's ability to correctly localize the objects in 3D. Graphs (g) and (h) show a variant of our method that is targeted directly at obtaining accurate pose estimation, at the price of recognizing fewer objects overall. We consider only the DPM-bank detector, which is our strongest input pose estimator. Note the low number of true positives but high pose classification accuracy of this approach.

8.9 Chapter Summary

We have described a detailed object-parts model and explained its use for the problem of multiple viewpoint 3D scene understanding. Visual object recognizers trained for the task of predicting both the parts layout as well as the object category and location within an image are used as input to our procedure. Our state space instantiates 3D object parts that fit within the entire object shape in 3D, and these are used to associate data between the model and observations. An inference procedure based on MCMC searches over possible scene layouts guided by our likelihood model. We have analyzed the performance of this technique on the Ford Campus dataset [PME11], SSFM *Test Pairs* task from [BS11]. Results demonstrate that our method is capable of utilizing the parts information to make confident and correct predictions for many objects and to predict the viewpoint of these objects in 3D.

The fixed number of objects in each scene assumed by our model is a major limitation for use of this technique on real platform faced with a rapidly changing and unpredictable visual experience. We have begun examining the use of a trans-dimensional sampling technique known as *Reversible Jump MCMC*, which has the promise to simultaneously infer the number of objects in a scene and the layout of those objects. This work is in a preliminary form as of the writing of this thesis and will be targeted for publication in the near future.

We have now concluded discussion of all technical contribution in this thesis. The final chapter will provide an overall summary and discuss future directions at a broader scope.



(a) (b)

(c) (d)

(e) (f)

(g) (h)

Figure 8.8: Evaluation of Viewpoint Prediction Accuracy: (left) Viewpoint confusion matrices with discrete viewing angles labeled {l:left, lf:left-front, f:front, fr:front-right, r:right, br:back-right, b:back and bl:back-left} and error histograms for four variants of our system.

Chapter 9

Conclusions

9.1 Thesis Summary

This thesis has described several contributions to robotic object recognition. During the course of this thesis, we developed a physical robot system for recognizing objects, overcame many challenges related to directing the robot's camera and integrated numerous planning and vision components. This led to the observation that the core task required for a robot to recognize objects is reasoning about the 3D positions of objects based on multiple images. We collected and released a dataset for repeatable evaluation of this task and began to develop algorithms to solve a number of the sub-problems. Our methods include path planners that account for the appearance of objects from various viewpoints and recognizers that are robust to occlusion. We will briefly summarize our primary contributions in each of these areas.

The Curious George intelligent system is capable of building a detailed semantic representation of its environment, which was demonstrated by three first place finishes in the Semantic Robot Vision Challenge over three years. The development of this robot platform clearly highlighted some of the limitations of existing methods and motivated the remaining work within the thesis.

The UBC VRS dataset and benchmark captures realistic sensory data collected by a physical robot platform in challenging environments. The goal of the UBC VRS project was to enable repeatable study of the multiple viewpoint robot recognition task for indoor scenes in a fashion that has not previously been feasible by capturing the actual sensor experiences of a platform within realistic environments and preserving richer spatial information than image-only datasets. As part of the UBC VRS project, we have developed and released a software suite suitable for representing and simulating the object recognition problem facing a mobile robot and have shown that our tools can be adapted to other datasets, such as urban driving data.

The algorithmic contributions of this thesis centre on a probabilistic model that relates images from multiple views of a scene to the 3D objects that are present. First, this model was used to develop an approach for entropy-guided motion planning that enables accurate recognition of objects more quickly than by allowing the robot to move randomly. Second, we developed an occlusion-aware 3D object model that leveraged rough, viewpoint-agnostic partial object templates and explicit occlusion reasoning based on sensed 3D data. We showed that this approach has the ability to robustly recognize mugs and bowls in cluttered kitchen scenes. Third, we extended the previous approach by including detailed, viewpoint-aware, semantically meaningful 3D object parts. This model was shown to be an effective approach to scene understanding in urban scenes such as parking lots.

In general, the author has attempted to draw a clear connection between entities that have often been treated independently, namely: semantic object labels; intermediate representations such as visual features and object recognizers; the 3D world where our object models and robot trajectories are grounded; and finally a robot's control policies and decision making process. Our results support the utility of 3D reasoning as part of the recognition process. The trend is that multiple viewpoint approaches achieve significant improvements in the high precision region. This is a useful trait in realistic applications, since we often want to be quite certain of the presence of an object before attempting an action, such as grasping. Our approaches typically achieve slightly lower recall than methods that operate in the space of single images, which needs addressing in future work. An important product of this thesis is the software implementation of each of the methods described. Much of that software has been made publicly available. Where not clearly stated previously in the thesis, this code can be obtained from a permanent website hosted at UBC[1].

[1] http://www.cs.ubc.ca/labs/lci/meger-code/

9.2 Open Problems

Object recognition requires reasoning about the complex nature of the real world, and current techniques are far from perfect in any but the simplest environments. Solutions are often stochastic and driven by data, so some amount of uncertainty is the norm. The techniques presented in this thesis do not perform perfectly, even on the evaluation tasks that we have defined and practical systems may still fall short of the needed performance for success in the real world. This section will identify a number of the areas that offer the most promise for improvements in the near future.

9.2.1 3D Appearance Models

The vast majority of physical objects that a robot seeks to recognize are non-planar, however this 3D shape information is still rarely considered in the learning or application of the appearance models used to recognize objects within visual images. Recently, methods that have a basic awareness of the shape of objects, such as their parts-layout or the viewing direction of the training data have shown strong progress to improve recognition performance (e.g., [SXBS10, SGS10, PSGS12]). However, these models all still fall well short of the spatial and physical information that a human is able to infer while they recognize an item, which includes: orientation and position of visible surfaces; environmental lighting properties such as reflections and cast shadows; moderately accurate scale; occlusion-boundaries between items; material properties such as specularity; and the detailed physics and articulation of the target objects. These forms of information have all been studied, to various extents, in relation to visual tasks including object recognition, but current models are typically only minimally influenced by the information.

The models developed in this thesis relate appearance patches in numerous images by associating them to a common 3D object, reasoning about the spatial properties of this object, and projecting these properties into each image. The location, scale, orientation and even 3D parts that we model are drastically over-simplified versions of the true rich 3D structures that make up real objects. Our most complicated model was still only a 3D box full of 3D boxes. Improved knowledge of each object's spatial and physical properties could significantly improve our predictions for apperance transfer. We chose simple models in our work because the 2D visual appearance models that we relied upon only output information at this level of detail (i.e., 2D bounding boxes, or at most 2D boxes full of

2D part boxes). The extension of our work to include more detailed spatial models for the 3D object properties in the future should allow for better recognition performance and specifically more spatial resolution in our ability to localize objects in 3D. A key element in these methods will have to be an increased ability to recover 3D properties from each image patch, so that these spatial properties can be related to observed data.

9.2.2 Dynamic Objects

Throughout this thesis, we have assumed that objects in the world are stationary. This assumption is often violated by instances of the object categories that are of interest to an intelligent system. The problem of continuously locating an object through a dynamic trajectory is referred to as *target tracking*. A number of the techniques proposed in this thesis have application to the tracking problem. Several have already been considered by other authors, such as the use of a 3D state space to apply meaningful matching constraints across views for analysis of moving objects (e.g., [UFF06, ARS10]). Occlusion reasoning for moving objects has been studied, for example by [WRSS10], although the occlusion reasoning has typically been simpler than the analysis of sensed 3D range that we have implemented.

The main technical challenge in applying our probabilistic inference to the tracking problem is that the size of the state space grows with the trajectory length, and thus the approaches would likely need to be applied within a filtering framework in order to maintain tractable inference. However, we have shown that our cross-view constraints assist object reasoning with as few as two images (i.e., the *Test Pairs* scenario), so our technique can reasonably be used in this fashion. In the tracking scenario, the prior term that we place on static object locations would becomes a motion model that favors certain likely motions, such as those with continuous velocities in a reasonable range.

9.2.3 Registration and Mapping

The second major assumption of this thesis that could be relaxed in the future is that an external process is able to accurately register sensor positions into a global coordinate frame. While mapping approaches are becoming increasingly powerful, there are still many vehicles whose computational or sensory limitations make on-board accurate mapping impossible. Our methods currently would not be directly applicable to those situations. An obvious direction would be to replace the delta function form of registration likelihood that we have used in this work and ask our methods to recover precise and detailed image registration along with the object positions, as is done by [BS11].

Potentially a more interesting direction is to consider looser spatial constraints, such as those that are introduced in topological mapping in robotic systems (e.g., [IDD99]). Rather than knowing the exact camera and object positions, it may often be sufficient to know the name of the place where the robot is located, the types of objects that are located there and if the objects that were seen before have remained in place or have moved. A proof-of-concept of such a navigation system was proposed by Vasudevan *et al.* [VGNS07]. The use of an approach such as ours over a local set of frames within a room *scene* would be an ideal input to higher-level place reasoning. In either case, the semantic mapping problem will be an interesting avenue for research in the future, and many of the techniques in this thesis will be appropriate components to such systems.

9.2.4 Segmentation

Our methods only describe a few specific objects in each scene, with the rest of the world treated as an unstructured *non-object* background. Many techniques are available for forming groups of the world with self-similar appearance, regardless of their object label: a problem referred to as segmentation. The results of segmentation could be of great use to our inference procedures. Object boundaries are likely to coincide with segment boundaries. Large uniform segments of the world are likely to be support surfaces and can be eliminated from the object search process. Several authors have previously considered segmentation as part of the object understanding process, such as [WRSS10]. However, state-of-the-art segmentation methods often make errors and are not repeatable in realistic images. Therefore, the best we can hope for is an additional probabilistic cue that could be considered along with the output of object recognizers. The combination of recognizers and segmentation is a promising direction for future work.

9.2.5 Online and Life-long Learning

All of the experiments that we have presented in this thesis shared a common overall procedure. First, a human system designer instructed our system which object categories would be the targets for recognition. Training data was provided to our system in some format.

This data primarily took the form of labeled bounding boxes in images, except for the SRVC contest experiments, where we used web data with weaker labels only on the entire image. Appearance models were learned based on this training data. Afterwards, the robot explored the world and the fixed appearance models were used to analyze the sensory data obtained by our system and to locate objects in the world.

The procedure outlined above is somewhat limited. It does not provide a means for an intelligent system to adapt to an environment that might change during the course of its operation. New objects might be introduced, or the robot might move from one area to another where the set of objects encountered is quite different. Our Curious George platform encountered one instance of this circumstance when attending the SRVC contest in Anchorage, Alaska where the target object *Ulu* was not previously in the robot's vocabulary of objects and the web training data that the robot attempted to use to learn its appearance was very limited.

Lifelong learning is a different approach that may prove to be more flexible in such circumstances. This refers to adapting the system's existing models, adding new models or discarding those that are no longer necessary to adapt to a changing environment. One primary challenge in autonomous life-long learning is that it requires a system to go beyond the static set of labeled data that it is given at the start of its existence. Semi-supervised learning is one potential option. This involves using the small set of labeled images that are initially given in training plus the large number of unlabeled images seen while exploring the real world together in an on-going learning process.

Another option is to allow human-robot interaction during normal system operation. For example, a robot could query the user for new labels when introspection reveals that the robot is sufficiently uncertain or a user could audit the performance over a period and provide corrective input. In each of these processes, cross-view, 3D, and affordance constraints can be used during the robot's normal operation to allow it to adjust and augment its models. For example, suppose a new type of car is introduced into the city where our autonomous driving platform operates. These vehicles might not match well to the existing appearance models available to our system, but the objects could be segmented from the background environment, and would meet the same set of motion, size, and layout properties that the existing set of automobiles possess. These would be consistent for that object across all views. Collection of new training data and learning of additional training models could potentially be triggered once enough of these criteria are met with confidence. Our system is well-positioned for this task.

9.2.6 Place and Task Context

We have considered the tasks of locating and recognizing objects mostly independent of the many other functions that a platform is likely to be performing. Practical robot systems are usually not passively exploring, but rather they continuously perform tasks under the direction of a human user. This could require objects to be manipulated, for the robot to travel between a number of workspaces or for some joint task such as *hand-off* between the robot and a person. In each of these contexts, a significant amount of additional information is available to assist with the recognition process. Temporal continuity can be used to track an object that is identified by the user with a label in one single image. This removes the need to re-recognize in each subsequent frame, except to safe-guard against tracking errors. Also, each location will have different usual sets of objects, and those objects will have usual positions, which allows for much stronger priors than we have assumed in our work. As methods such as ours are applied on practical systems that carry out tasks in real environments, it will be important to utilize the context information that is available from tasks and places.

9.3 Outlook and Final Remarks

For the approaches described in this thesis to be useful to the robot practitioners of the future, additional advances will be needed at a number of steps. We must continue to work on our vocabulary of discriminative features, so that all objects of interest can be reliably described and detected within sensory data. The level of detail in our understanding of the world should be increased, so that we reason not only about boxes and cuboids, but rather about surfaces, linkages and materials. Integration between communities is needed, for example, between the perceptual models that have lately been successful in the computer vision community and the planning approaches that exist in the study of robotics. Humans carry out a vast array of motions that allow us to interact with objects, to simultaneously perceive those objects, and eventually to perform tasks effectively. Current techniques in intelligent perception are, for the most part, unaware of the motion or action space of their vehicle and planning approaches often make over-simplified assumptions about the outputs of their perception. This thesis and its extensions will hopefully bring perception slightly closer to the space where robot task planning is already successful and to show the way for these tasks to be more tightly coupled in the near future.

Bibliography

[AdFDJ03] Christophe Andrieu, Nando de Freitas, Arnaud Doucet, and Michael I Jordan. An Introduction to MCMC for Machine Learning, 2003. → pages 139

[AF99] Tal Arbel and Frank P. Ferrie. Viewpoint Selection by Navigation through Entropy Maps. In *Proceedings of International Conference on Computer Vision (ICCV)*, pages 248–254, 1999. → pages 25

[AMFM11] Pablo Arbeláez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(5):898–916, May 2011. → pages 20

[ARS10] Mykhaylo Andriluka, Stefan Roth, and Bernt Schiele. Monocular 3D Pose Estimation and Tracking by Detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, San Francisco, USA, 2010. → pages 33, 159

[AWB88] J. Aloimonos, I. Weiss, and A. Bandyopadhyay. Active Vision. *International Journal of Computer Vision*, 1(4):333 – 356, 1988. → pages 25

[BK91] J Borenstein and Y Koren. The Vector Field Histogram – Fast Obstacle-Avoidance for Mobile Robots. *IEEE Journal of Robotics and Automation*, 7(3):278–288, June 1991. → pages 47

[BL97] J.S. Beis and D.G. Lowe. Shape indexing using approximate nearest-neighbour search in high-dimensional spaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1000–1006, 1997. → pages 16

[BMP00] S Belongie, J Malik, and J Puzicha. Shape context: A new descriptor for shape matching and object recognition. In *Proceedings of the Conference on Neurial Information Processing Systems (NIPS)*, 2000. → pages 15, 17, 19

[Bra] Gary Bradski. Website: http://solutionsinperception.org/index.html/. → pages 30

[BRL+09] Michael D Breitenstein, Fabian Reichlin, Bastian Leibe, Esther Koller-Meier, and Luc Van Gool. Robust Tracking-by-Detection using a Detector Confidence Particle Filter. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2009. → pages 32

[BS11] Sid Yingze Bao and Silvio Savarese. Semantic Structure from Motion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011. → pages 29, 31, 32, 133, 144, 145, 154, 160

[BSS11] S Y Bao, M Sun, and S Savarese. Toward Coherent Object Detection and Scene Layout Understanding. *Image and Vision Computing*, 2011. → pages 23

[BTV] H Bay, T Tuytelaars, and L Van Gool. Surf: Speeded-up Robust Features. In *Proceedings of the European Conference on Computer Vision (ECCV)*. → pages 15, 17, 123

[CF08] Stephen Cawood and Mark Fiala. *Augmented Reality A Practical Guide*. Pragmatic Bookshelf, 2008. → pages 66

[DBPT10] Jennifer Dolson, Jongmin Baek, Christian Plagemann, and Sebastian Thrun. Upsampling Range Data in Dynamic Environments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010. → pages 27

[DCTO97] S.J. Dickinson, H.I. Christensen, J.K. Tsotsos, and G. Olofsson. Active object recognition integrating attention and viewpoint control. *Computer Vision and Image Understanding*, 67(3):239–260, 1997. → pages 25

[DDS+09] J Deng, W Dong, R Socher, L.-J. Li, K Li, and L Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. → pages 13

[dic00] *The American Heritage Dictionary of the English Language*. Houghton Mifflin Company, Boston, MA, fourth edition, 2000. → pages 4

[DLD+09] J Deng, K Li, M Do, H Su, and L Fei-Fei. Construction and Analysis of a Large Scale Image Ontology. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Vision Sciences Society, 2009. → pages 30, 54

[DT05] N Dalal and B Triggs. Histograms of Oriented Gradients for Human Detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 886–893, San Diego, USA, June 2005. → pages 17

[EESG10] Markus Enzweiler, Angela Eigenstetter, Bernt Schiele, and Dariu Gavrila. Multi-Cue Pedestrian Classification with Partial Occlusion Handling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010. → pages 22

[EHE$^+$12] F Endres, M Hess, N Engelhard, J Sturm, D Cremers, and W Burgard. An Evaluation of the RGB-D SLAM System. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2012. → pages 77

[Eid10] Robert Eidenberger. Autonomous Knowledge Acquisition of a Service Robot by Probabilistic Perception Planning. In *IROS 2010 Workshop on Semantic Mapping and Autonomous Knowledge Acquisition*, 2010. → pages 26

[EJK06] Staffan Ekvall, Patric Jensfelt, and Dana Kragic. Integrating Active Mobile Robot Object Recognition and SLAM in Natural Environments. In *IEEE/RSJ International Conference on Robotics and Automation (IROS)*, Beijing, China, 2006. IEEE. → pages 37

[ESLV10] Andreas Ess, Konrad Schindler, Bastian Leibe, and Luc Van Gool. Object Detection and Tracking for Autonomous Navigation in Dynamic Environments. *The International Journal of Robotics Research*, 29(14):1707–1725, December 2010. → pages 31, 32

[EVW$^+$12] M Everingham, L Van Gool, C K I Williams, J Winn, and A Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html, 2012. → pages 13, 28, 29, 55, 100

[FB81] M A Fischler and R C Bolles. Random Sample Consensus: A paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM*, 24:381–395, 1981. → pages 16, 40, 123

[FBR77] J Friedman, J Bentley, and Finkel R. An Algorithm for Finding Best Matches in Logarithmic Expected Time. *ACM Transactions on Mathematical Software*, 3:209–226, 1977. → pages 16

[FBT98] Dieter Fox, Wolfram Burgard, and Sebastian Thrun. Active Markov Localization for Mobile Robots. *Robotics and Autonomous Systems (RAS)*, 25:195–207, 1998. → pages 25

[FBT99] D Fox, W Burgard, and S Thrun. Markov Localization for mobile robots in dynamic environments. *Journal of Artificial Intelligence Research*, 11, 1999. → pages 46

[FDU12] Sanja Fidler, Sven Dickinson, and Raquel Urtasun. 3D Object Detection and Viewpoint Estimation with a Deformable 3D Cuboid Model. In *Proceedings of the Conference on Neural Information Processing Systems (NIPS)*, 2012. → pages 23

[FEHF09] A Farhadi, I Endres, D Hoiem, and D Forsyth. Describing Objects by their Attributes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1778–1785. Ieee, June 2009. → pages 20

[FFFP04] L Fei-Fei, R Fergus, and P Perona. Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshop on Generative-Model Based Vision*, 2004. → pages 29

[FGM10] P Felzenszwalb, R Girshick, and D McAllester. Cascade Object Detection with Deformable Part Models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010. → pages 18

[FGMR10] P Felzenszwalb, R Girshick, D McAllester, and D Ramanan. Object Detection with Discriminatively Trained Part Based Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9), 2010. → pages 10, 17, 18, 89, 95, 97, 100, 113, 121, 124, 125, 133

[FH05] P Felzenszwalb and D Huttenlocher. Pictorial Structures for Object Recognition. *International Journal of Computer Vision*, 61(1):55–79, 2005. → pages 17

[Fia05] M Fiala. ARTag, a Fiducial Marker System Using Digital Techniques. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 590–596, November 2005. → pages 66, 69, 123

[FML$^+$08] Per-Erik Forssén, David Meger, Kevin Lai, Scott Helmer, James J Little, and David G Lowe. Informed Visual Search: Combining Attention and Object

Recognition. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2008. → pages iii, 11

[For07] Per-Erik Forssén. Learning Saccadic Gaze Control via Motion Prediction. In *Proceedings of the 4th Canadian Conference on Computer and Robot Vision*. IEEE Computer Society, May 2007. → pages 44

[FPZ03] R Fergus, P Perona, and A Zisserman. Object Class Recognition by Unsupervised Scale-invariant Learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 264–271, 2003. → pages 16, 17

[FS97] Y Freund and R E Schapire. A Decision-theoretic Generalization of on-line learning and an Application to Boosting. *Journal of Computer and System Sciences*, 1997. → pages 19, 32

[FSD10] Mario Fritz, Kate Saenko, and Trevor Darrell. Size Matters: Metric Visual Search Constraints from Monocular Metadata. In *Proceedings of the Conference on Neural Information Processing Systems (NIPS)*, 2010. → pages 26

[GAK$^+$07] S Gould, J Arfvidsson, A Kaehler, B Sapp, M Meissner, G Bradski, P Baumstarck, S Chung, and A Ng. Peripheral-Foveal Vision for Real-time Object Recognition and Tracking in Video. In *Twentieth International Joint Conference on Artificial Intelligence (IJCAI-07)*, 2007. → pages 24

[GAP$^+$11] M Göbelbecker, A Aydemir, A Pronobis, K Sjö, and P Jensfelt. A Planning Approach to Active Visual Search in Large Environments. In *Workshops at the Twenty-Fifth AAAI Conference on Artificial Intelligence*, 2011. → pages 25

[GBQ$^+$08] Stephen Gould, Paul Baumstarck, Morgan Quigley, Andrew Y Ng, and Daphne Koller. Integrating Visual and Range Data for Robotic Object Detection. In *Proceedings of the ECCV workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications (M2SFA2)*, 2008. → pages 26

[GD05] K Grauman and T Darrell. The Pyramid Match Kernel: Discriminative Classification with Sets of Image Features. In *Proceedings of the International Conference on Computer Vision (ICCV)*, Beijing, China, October 2005. → pages 100

[GFM11] R Girshick, P Felzenszwalb, and D McAllester. Object Detection with Grammar Models. In *Proceedings of the Conference on Neural Information Processing Systems (NIPS)*, 2011. → pages 17, 111

[GGV11] Helmut Grabner, Juergen Gall, and Luc Van Gool. What Makes a Chair a Chair? In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011. → pages 27

[GHP07] G Griffin, A Holub, and P Perona. Caltech-256 Object Category Dataset. Technical Report 7694, California Institute of Technology, 2007. → pages 29

[Gib79] J J Gibson. *The ecological approach to visual perception*. Houghton Miflin, Boston, MA, 1979. → pages 14

[GKS$^+$10] Giorgio Grisetti, Rainer Kuemmerle, Cyrill Stachniss, Udo Frese, and Christoph Hertzberg. Hierarchical Optimization on Manifolds for Online 2D and 3D Mapping. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2010. → pages 123

[GSS96] W Gilks, S. Richardson, and D J Spiegelhalter. *Markov chain Monte Carlo in practice*. Chapman and Hall, 1996. → pages 20

[Has70] W Hastings. Monte Carlo Sampling Methods Using Markov chains and their Applications. *Biometrika*, 57:97–109, 1970. → pages 138

[HCD12] Derek Hoiem, Yodsawalai Chodpathumwan, and Qieyun Dai. Diagnosing Error in Object Detectors. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2012. → pages 89, 136

[HEH06] D Hoiem, A A Efros, and M Hebert. Putting objects in perspective. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 80. Springer, 2006. → pages 23

[HH12] Edward Hsiao and Martial Hebert. Occlusion Reasoning for Object Detection under Arbitrary Viewpoint. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. → pages 22

[HL10] Scott Helmer and David Lowe. Object Recognition Using Stereo. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, 2010. → pages 26

[HMM$^+$10] Scott Helmer, David Meger, Marius Muja, James J Little, and David G Lowe. Multiple Viewpoint Recognition and Localization. In *Proceedings of the Asian Computer Vision Conference*, 2010. → pages 33, 65, 78

[HW79] D H Hubel and T N Wiesel. Brain Mechanisms of Vision. *Scientific American*, 241:150–162, 1979. → pages 14

[HZ00] R I Hartley and A Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521623049, 2000. → pages 72

[HZ07] Xiaodi Hou and Liging Zhang. Saliency Detection: A Spectral Residual Approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, June 2007. → pages 41

[IDD99] Ioannis M. Rekleitis, Vida Dujmović, and Gregory Dudek. Efficient Topological Exploration. In *Proceedings of International Conference in Robotics and Automation*, pages 676–681, Detroit, USA, May 1999. → pages 160

[JFB02] Allan D Jepson, David J Fleet, and Michael J Black. A Layered Motion representation with occlusion and compact spatial support. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 692–706, 2002. → pages 22

[JKJ$^+$11] A Janoch, S Karayev, Y Jia, J T Barron, M Fritz, K Saenko, and T Darrell. A Category-Level 3-D Object Dataset: Putting the Kinect to Work. In *Proceedings of the ICCV Workshop on Consumer Depth Cameras in Computer Vision*, 2011. → pages 30

[KB01] Timor Kadir and Michael Brady. Scale, Saliency and Image Description. *International Journal of Computer Vision (IJCV)*, 45(2):83–105, 2001. → pages 15

[KB06] D Kragic and M Björkman. Strategies for Object Manipulation using Foveal and Peripheral Vision. In *Proceedings of the IEEE International Conference on Computer Vision Systems (ICVS)*, 2006. → pages 38

[KC05] D Kulić and E A Croft. Safe planning for human-robot interaction. *Journal of Robotic Systems*, 22(7):383–396, 2005. → pages 27

[LA06] Catherine Laporte and Tal Arbel. Efficient Discriminant Viewpoint Selection for Active Bayesian Recognition. *International Journal of Computer Vision*, 68:1405–1573, 2006. → pages 21, 25, 98

[LBRF11] Kevin Lai, Liefeng Bo, Xiaofeng Ren, and Dieter Fox. A Large-Scale Hierarchical Multi-View RGB-D Object Dataset. In *Proceedings of the IEEE*

International Conference on Robotics and Automation (ICRA), 2011. → pages 13, 30

[LBRF12] Kevin Lai, Liefeng Bo, Xiaofeng Ren, and Dieter Fox. Detection-based Object Labeling in 3D Scenes. In *Proeedings of the IEEE International Conference on on Robotics and Automation (ICRA)*, 2012. → pages 33

[LCCV07] B. Leibe, N. Cornelis, K. Cornelis, and L. Van Gool. Dynamic 3D Scene Analysis from a Moving Vehicle. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007. → pages 31

[Lin90] Tony Lindeberg. Scale-Space for Discrete Signals. *IEEE Transactions of Pattern Analysis and Machine Intelligence (PAMI)*, 12(3):234–254, 1990. → pages 14

[LLS08] B Leibe, A Leonardis, and B Schiele. Robust Object Detection with Interleaved Categorization and Segmentation. *International Journal of Computer Vision Special Issue on Learning for Recognition and Recognition for Learning*, 77(1-3):259–289, 2008. → pages 16, 17

[LOL08] Wei-Lwun Lu, Kenji Okuma, and James J. Little. Tracking and Recognizing Actions of Multiple Hockey Players using the Boosted Particle Filter. *Image and Vision Computing*, 2008. → pages 32

[Low04] David G Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, 2004. → pages 15, 99

[LSD12] Alex Levinshtein, Cristian Sminchisescu, and Sven Dickinson. Optimal Image and Video Closure by Superpixel Grouping. *International Journal of Computer Vision*, 100(1):99–119, May 2012. → pages 20

[LSP06] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2169–2178, New York, June 2006. IEEE Computer Society. → pages 50

[LSS08] Joerg Liebelt, Cordelia Schmid, and Klaus Schertler. Viewpoint-Independent Object Class Detection using 3D Feature Maps. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008. → pages 20, 95

[MCUP02] Jiri Matas, Ondrej Chum, Martin Urban, and Tomas Pajdla. Robust Wide Baseline Stereo from Maximally Stable Extremal Regions. In *Proceedings of the 13th British Machine Vision Conference (BMVC)*, pages 384–393, September 2002. → pages 41

[ME85] H Moravec and A Elfes. High-resolution maps from wide-angle sonar. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 116–121, St. Louis, MO, USA, 1985. → pages 46

[MFL$^+$07] David Meger, Per-Erik Forssén, Kevin Lai, Scott Helmer, Sancho McCann, Tristram Southey, Matthew Baumann, James J. Little, David G. Lowe, and Bruce Dow. Curious George: An Attentive Semantic Robot. In *Proceedings of IROS Workshop: From sensors to human spatial concepts*, San Diego, CA, USA, November 2007. IEEE. → pages iii, 10

[MFL$^+$08] David Meger, Per-Erik Forssén, Kevin Lai, Scott Helmer, Sancho McCann, Tristram Southey, Matthew Baumann, James J. Little, and David G. Lowe. Curious George: An Attentive Semantic Robot. *Robotics and Autonomous Systems Journal Special Issue on From Sensors to Human Spatial Concepts*, 56(6):503–511, November 2008. → pages iii, 11, 36, 107

[MFS$^+$07] Gérard Medioni, Alexandre R J François, Matheen Siddiqui, Kwangsu Kim, and Hosub Yoon. Robust real-time vision for a personal service robot. *Computer Vision and Image Understanding*, 108(1-2):196–203, 2007. → pages 23

[MGL10] David Meger, Ankur Gupta, and James J Little. Viewpoint Detection Models for Sequential Embodied Object Category Recognition. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, 2010. → pages iii, 11, 97

[Min09] Silvio Savarese Li Fei-Fei Min Sun Hao Su. A Multi-View Probabilistic Model for 3D Object Classes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009. → pages 64

[ML09] Marius Muja and David G Lowe. Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration. In *Proceedings of the International Conference on Computer Vision Theory and Applications (VISAPP)*, 2009. → pages 16

[ML11] David Meger and James J Little. Mobile 3D Object Detection in Clutter. In *Proceedings of the IEEE/RSJ Conference on Robots and Intelligent Systems (IROS)*, 2011. → pages iii, 11, 78

[ML12] David Meger and James J Little. The UBC Visual Robot Survey: A Benchmark for Robot Category Recognition. In *Proceedings of The International Symposium on Experimental Robotics (ISER)*, Quebec City, Canada, 2012. → pages iii, 11, 65

[MRR$^+$53] Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of State Calculations by Fast Computing Machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1953. → pages 138

[MS79] R B Marimont and M B Shapiro. Nearest Neighbour Searches and the Curse of Dimensionality. *IMA Journal of Applied Mathematics*, 24(1):59–70, 1979. → pages 16

[MS04] Krystian Mikolajczyk and Cordelia Schmid. Scale and Affine Invariant Interest Point Detectors. *International Journal of Computer Vision (IJCV)*, 2004. → pages 15

[MTKW03] M Montemerlo, S Thrun, D Koller, and B Wegbreit. FastSLAM 2.0: An Improved Particle Filtering Algorithm for Simultaneous Localization and Mapping that Provably Converges. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1151–1156, Acapulco, Mexico, 2003. IJCAI. → pages 46

[MTS$^+$05] K Mikolajczyk, T Tuytelaars, C Schmid, A Zisserman, J Matas, F Schaffalitzky, T Kadir, and L Van Gool. A Comparison of Affine Region Detectors. *International Journal of Computer Vision*, 65:43–72, 2005. → pages 15, 99

[MU49] Nicholas Metropolis and S Ulam. The Monte Carlo Method. *Journal of the American Statistical Association*, 44(247):335–341, 1949. → pages 128, 138

[MWSL11] David Meger, Christian Wojek, Bernt Schiele, and James J Little. Explicit Occlusion Reasoning for 3D Object Detection. In *Proceedings of the 22nd British Machine Vision Conference (BMVC)*, 2011. → pages iii, 11, 65, 78, 113

[MY09] J M Morel and G Yu. ASIFT, A New Framework for Fully Affine Invariant Image Comparison. *SIAM Journal on Imaging Sciences*, 2(2):438–469, 2009. → pages 15

[NMC05] A Niculescu-Mizil and R Caruana. Obtaining Calibrated Probabilities from Boosting. In *Proceedings of the Conference on Uncertainty and Artificial Intelligence*, 2005. → pages 19, 118

[OPZ06] A Opelt, A Pinz, and A Zisserman. Incremental Learning of Object Detectors Using a Visual Shape Alphabet. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006. → pages 21

[OTD$^+$04] K. Okuma, A. Taleghani, N. DeFreitas, James J. Little, and D. G. Lowe. A Boosted Particle Flter: Multitarget Detection and Tracking. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2004. → pages 32

[Pal99] Stephen E Palmer. *Vision Science: Photons to Phenomenology*. MIT Press, 1999. → pages 49

[PG11] Devi Parikh and Kristen Grauman. Relative attributes. In *Proceedings of the International Conference on Computer Vision (ICCV)*, pages 503–510. Ieee, November 2011. → pages 20

[PME11] Gaurav Pandey, James R McBride, and Ryan M Eustice. Ford Campus Vision and Lidar Data Set. *International Journal of Robotics Research*, 30(13):1543–1552, November 2011. → pages 31, 77, 130, 143, 154

[PR09] Samuel Prentice and Nicholas Roy. The Belief Roadmap: Efficient Planning in Belief Space by Factoring the Covariance. *International Journal of Robotics Research*, 2009. → pages 25, 105

[PSGS12] Bojan Pepik, Michael Stark, Peter Gehler, and Bernt Schiele. 3D2PM – 3D Deformable Part Models. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2012. → pages 158

[PTPB12] Dejan Pangercic, Moritz Tenorth, Benjamin Pitzer, and Michael Beetz. Semantic Object Maps for Robotic Housework - Representation, Acquisition and Use. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012. → pages 24

[PTZ$^+$10] Victor Adrian Prisacariu, Radu Timofte, Karel Zimmermann, Ian Reid, and Luc Van Gool. Integrating Object Detection with 3D Tracking Towards a Better Driver Assistance System. In *Proceeings of the International Conference on Pattern Recognition*, pages 3344–3347. IEEE, August 2010. → pages 27

[QBG$^+$09] Morgan Quigley, Siddharth Batra, Stephen Gould, Ellen Klingbeil, Quoc V Le, Ashley Wellman, and Andrew Y Ng. High-Accuracy 3D Sensing for Mobile Manipulation: Improving Object Detection and Door Opening. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2009. → pages 26

[RBHK09] B. Rasolzadeh, M. Bjorkman, K. Huebner, and D. Kragic. An Active Vision System for Detecting, Fixating and Manipulating Objects in the Real World. *The International Journal of Robotics Research*, 29(2-3):133–154, August 2009. → pages 24

[RD06] Ananth Ranganathan and Frank Dellaert. A Rao-Blackwellized Particle Filter for Topological Mapping. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 810–817, Orlando, USA, May 2006. → pages 26

[RD07] A Ranganathan and F Dellaert. Semantic Modeling of Places using Objects. In *Proceedings of Robotics: Science and Systems (RSS)*, 2007. → pages 37

[Ren00] Ronald Rensink. The Dynamic Representation of Scenes. *Visual Cognition*, 7(1/2/3):17–42, 2000. → pages 38

[RLXF11] Liefeng Bo Ren, Kevin Lai, Ren Xiaofeng, and Dieter Fox. Object Recognition with Hierarchical Kernel Descriptors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011. → pages 26

[RTMF08] Bryan C Russell, Antonio Torralba, Kevin P Murphy, and William T Freeman. LabelMe: a Database and Web-based Tool for Image Annotation. *International Journal of Computer Vision*, 77:157–173, 2008. → pages 30

[Rus09] R B Rusu. *Semantic 3D Object Maps for Everyday Manipulation in Human Living Environments*. PhD thesis, Technical University of Munich, 2009. → pages 40

[SBFC03] Dirk Schulz, Wolfram Burgard, Dieter Fox, and Armin B. Cremers. People Tracking with Mobile Robots Using Sample-Based Joint Probabilistic Data Association Filters. *The International Journal of Robotics Research*, 22(2):99–116, February 2003. → pages 23

[SBV07] M Schlemmer, G Biegelbauer, and M Vincze. Rethinking Robot Vision - Combining Shape and Appearance. *International Journal of Advanced Robotic Systems*, 4:259–270, 2007. → pages 23

[SBWK10] Agnes Swadzba, Niklas Beuter, Sven Wachsmuth, and Franz Kummert. Dynamic 3D Scene Analysis for Acquiring Articulated Scene Models. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 134–141. Ieee, May 2010. → pages 27

[SC00] B Schiele and J L Crowley. Recognition without Correspondence using Multidimensional Receptive Field Histograms. *International Journal of Computer Vision (IJCV)*, 36(1):31–50, 2000. → pages 17

[SCN08] A Saxena, S H Chung, and A Y Ng. 3D Depth Reconstruction from a Single Still Image. *International Journal of Computer Vision*, 76(1):53–69, 2008. → pages 20

[SD10] G Schindler and F Dellaert. Probabilistic Temporal Inference on Reconstructed 3D Scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1410–1417. IEEE, 2010. → pages 22

[SDN09] Ashutosh Saxena, Justin Driemeyer, and Andrew Y Ng. Learning 3-D Object Orientation from Images. In *International Conference on Robotics and Automation (ICRA)*, 2009. → pages 24

[SFF07] Silvio Savarese and Li Fei-Fei. 3D Generic Object Categorization, Localization and Pose Estimation. In *Proceedings of the International Conference on Computer Vision (ICCV)*, Brazil, October 2007. → pages 29, 30, 100, 106, 146

[SGS10] Michael Stark, Michael Goesele, and Bernt Schiele. Back to the Future: Learning Shape Models from 3D CAD Data. In *British Machine Vision Conference (BMVC)*, Aberystwyth, Wales, 2010. → pages 17, 19, 29, 129, 133, 134, 148, 158

[SJC00] M Seiz, P Jensfelt, and H I Christensen. Active Exploration for Feature Based Global Localization. In *Proceedings IEEE International Conference on Intelligent Robots and Systems (IROS)*, Takamatshu, October 2000. → pages 25

[SK00] Henry Schneiderman and Takeo Kanade. A statistical model for 3D object detection applied to faces and cars. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2000. → pages 20

[SKP$^+$12] M Stark, J Krause, B Pepik, D Meger, James J Little, B Schiele, and D Koller. Fine-Grained Categorization for 3D Scene Understanding. In *Proceedings of the 23rd British Machine Vision Conference (BMVC)*, Surrey, UK, 2012. → pages 130

[SL06] Tristram Southey and James J Little. Object Discovery through Motion, Appearance and Shape. In *AAAI Workshop on Cognitive Robotics, Technical Report WS-06-03*. AAAI Press, 2006. → pages 38

[SLP+08] Kristoffer Sjö, Dorian Galvez Lopez, Chandana Paul, Patric Jensfelt, and Danica Kragic. Object Search and Localization for an Indoor Mobile Robot. *Journal of Computing and Information Technology*, 2008. → pages 24

[SRS12] Wandi Susanto, Marcus Rohrbach, and Bernt Schiele. 3D Object Detection with Multiple Kinects. In *2nd Workshop on Consumer Depth Cameras for Computer Vision in conjunction with ECCV 2012*, Firenze, Italy, 2012. Springer. → pages 33

[SRV] SRVC. Website: http://www.semantic-robot-vision-challenge.org/. → pages 11, 29

[SSFFS09] Hao Su, Min Sun, Li Fei-Fei, and Silvio Savarese. Learning a dense multi-view representation for detection, viewpoint classification and synthesis of object categories. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Kyoto, Japan, October 2009. → pages 17, 20, 95

[SXBS10] Min Sun, Bing-Xin Xu, Gary Bradski, and Silvio Savarese. Depth-Encoded Hough Voting for Joint Object Detection and Shape Recovery. In *Proceedings of the European Conference on Computer Vision (ECCV)*, Crete, Greece, 2010. → pages 26, 158

[TFL+09] Alexander Thomas, Vittorio Ferrari, Bastian Leibe, Tinne Tuytelaars, and Luc Van Gool. Using Multi-View Recognition and Meta-data Annotation to Guide a Robot's Attention. *International Journal of Robotics Research*, 2009. → pages 17, 20, 21, 89, 95

[Tie94] Luke Tierney. Markov chains for exploring posterior distributions. *The Annals of Statistics*, pages 1701–1728, 1994. → pages 139

[TL87] R Y Tsai and R K Lenz. A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV cameras and Lenses. *IEEE Journal of Robotics and Automation*, pages 323–344, 1987. → pages 45, 70

[TMF04] A Torralba, K P Murphy, and W T Freeman. Sharing features: efficient boosting procedures for multiclass object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004. → pages 17, 21

[TML+03] S Thrun, C Martin, Y Liu, D Hahnel, R Emery-Montemerlo, D Chakrabarti, and W Burgard. A Real-Time Expectation Maximization Algorithm for

Acquiring Multi-Planar Maps of Indoor Environments with Mobile Robots. *IEEE Transactions on Robotics and Automation (TRO)*, 20(3):433–442, 2003. → pages 117

[TXLK11] Grace Tsai, Changhai Xu, Jingen Liu, and Benjamin Kuipers. Real-time indoor scene understanding using Bayesian filtering with motion cues. In *Proceedings of the International Conference in Computer Vision (ICCV)*, pages 121–128, November 2011. → pages 23

[UFF06] R Urtasun, D J Fleet, and P Fua. 3D people tracking with Gaussian process dynamical models. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 238–245, 2006. → pages 159

[UH05] Ranjith Unnikrishnan and Martial Hebert. Fast Extrinsic Calibration of a Laser Rangefinder to a Camera. Technical Report CMU-RI-TR-05-09, Robotics Institute, Pittsburgh, PA, July 2005. → pages 45

[VdF08] J Vogel and N de Freitas. Target-directed attention: Sequential decision-making for gaze planning. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2008. → pages 25

[Vel07] Velodyne. Velodyne HDL-64E: A high definition LIDAR sensor for 3D applications (whitepaper), 2007. → pages 131, 143

[VFJM09] Fredrik Viksten, Per-Erik Forssen, Björn Johansson, and Anders Moe. Comparison of Local Image Descriptors for Full 6 Degree-of-Freedom Pose Estimation. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2009. → pages 30

[VG09] S Vijayanarasimhan and K Grauman. What's It Going to Cost You? : Predicting Effort vs. Informativeness for Multi-Label Image Annotations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Miami, FL, June 2009. → pages 30

[VGNS07] S Vasudevan, S Gächter, V Nguyen, and R Siegwart. Cognitive maps for mobile robots—an object based approach. *Robotics and Autonomous Systems*, 55(5):359–371, 2007. → pages 26, 160

[VM07] Julia Vogel and Kevin Murphy. A non-myopic approach to visual search. In *Proceedings of the Fourth Canadian Conference on Computer and Robot Vision CRV*, pages 227–234, Montreal, Canada, May 2007. → pages 25

[Von67] H Von Helmholtz. *Handbuch der physiologischen Optik: mit 213 in den Text eingedruckten Holzschnitten und 11 Tafeln.* Voss, Leipzig, 1867. → pages 14

[VZ09] A Vedaldi and A Zisserman. Structured output regression for detection with partial occlusion. In *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, 2009. → pages 21

[WBDS08] K. Wyrobek, E. Berger, H.F.M. Der, Van Loos, and K. Salisbury. Towards a Personal Robotics Development Platform: Rationale and Design of an Intrinsically Safe Personal Robot. In *International Conference on Robotics and Automation*, 2008. → pages 24

[WCL$^+$08] C Wu, B Clipp, X Li, J.-M. Frahm, and M Pollefeys. 3D Model Matching with Viewpoint Invariant Patches (VIPs). In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008. → pages 15

[WF94] Peter Whaite and Frank Ferrie. Autonomous Exploration: Driven by Uncertainty. Technical Report TR-CIM-93-17, McGill U. CIM, March 1994. → pages 25

[WGK10] H. Wang, S. Gould, and D. Koller. Discriminative Learning with Latent Variables for Cluttered Indoor Scene Understanding. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2010. → pages 23

[WHY09] X Wang, T Han, and S Yan. A HOG-LBP human detector with partial occlusion handling. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2009. → pages 21

[Wit84] A Witkin. Scale-space filtering: A new approach to multi-scale description. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, (ICASSP)*, volume 9, pages 150–153, March 1984. → pages 14

[WK06] D Walther and C Koch. Modeling attention to salient proto-objects. *Neural Networks*, 19:1395–1407, 2006. → pages 41, 42, 43

[WRSS10] Christian Wojek, Stefan Roth, Konrad Schindler, and Bernt Schiele. Monocular 3D Scene Modeling and Inference: Understanding Multi-Object Traffic Scenes. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2010. → pages 31, 159, 160

[WWR$^+$] Christian Wojek, Stefan Walk, Stefan Roth, Konrad Schindler, and Bernt Schiele. Monocular Visual Scene Understanding: Understanding Multi-Object Traffic Scenes. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 35(4), 2013. → pages 31, 32

[WWRS11] Christian Wojek, Stefan Walk, Stefan Roth, and Bernt Schiele. Monocular 3D Scene Understanding with Explicit Occlusion Reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Colorado Springs, 2011. → pages 22, 113, 118, 142

[XK10] Changhai Xu and Benjamin Kuipers. Towards the Object Semantic Hierarchy. In *2010 IEEE 9th International Conference on Development and Learning*, pages 39–45, August 2010. → pages 27

[YHRF10] Yi Yang, Sam Hallman, Deva Ramanan, and Charles Fowlkes. Layered Object Detection for Multi-Class Segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010. → pages 22

[YSA98] B Yamauchi, A C Schultz, and W Adams. Mobile robot exploration and map-building with continuous localization. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 2833–2839, Leuven, Belgium, 1998. → pages 48

[YT99] Yiming Ye and John K Tsotsos. Sensor Planning for 3D Object Search. *Computer Vision and Image Understanding*, 73:145–168, 1999. → pages 23