Scene Understanding by Robotic Interactive Perception

Aamir Khan

Submitted in fulfilment of the requirements for the degree of
Doctor of Philosophy

School of Computing Science

College of Science and Engineering
University of Glasgow

September 2018

Abstract

This thesis presents a novel and generic visual architecture for scene understanding by robotic interactive perception. This proposed visual architecture is fully integrated into autonomous systems performing object perception and manipulation tasks. The proposed visual architecture uses interaction with the scene, in order to improve scene understanding substantially over non-interactive models. Specifically, this thesis presents two experimental validations of an autonomous system interacting with the scene: Firstly, an autonomous gaze control model is investigated, where the vision sensor directs its gaze to satisfy a scene exploration task. Secondly, autonomous interactive perception is investigated, where objects in the scene are repositioned by robotic manipulation. The proposed visual architecture for scene understanding involving perception and manipulation tasks has four components: 1) A reliable vision system, 2) Camera-hand eye calibration to integrate the vision system into an autonomous robots kinematic frame chain, 3) A visual model performing perception tasks and providing required knowledge for interaction with scene, and finally, 4) A manipulation model which, using knowledge received from the perception model, chooses an appropriate action (from a set of simple actions) to satisfy a manipulation task. This thesis presents contributions for each of the aforementioned components. Firstly, a portable active binocular robot vision architecture that integrates a number of visual behaviours are presented. This active vision architecture has the ability to verge, localise, recognise and simultaneously identify multiple target object instances. The portability and functional accuracy of the proposed vision architecture is demonstrated by carrying out both qualitative and comparative analyses using different robot hardware configurations, feature extraction techniques and scene perspectives. Secondly, a camera and hand-eye calibration methodology for integrating an active binocular robot head within a dual-arm robot are described. For this purpose, the forward kinematic model of the active robot head is derived and the methodology for calibrating and integrating the robot head is described in detail. A rigid calibration methodology has been implemented to provide a closed-form hand-to-eye calibration chain and this has been extended with a mechanism to allow the camera external parameters to be updated dynamically for optimal 3D reconstruction to meet the requirements for robotic tasks such as grasping and manipulating rigid and deformable objects. It is shown from experimental results that the robot head achieves an overall accuracy of fewer than 0.3 millimetres while recovering the 3D structure of a scene. In addition, a comparative study between current RGB-D cameras and our active stereo head within two dual-arm robotic test-beds is reported that demonstrates the accuracy and portability of our proposed methodology. Thirdly, this thesis proposes a visual perception model for the task of category-wise objects sorting, based on Gaussian Process (GP) classification that is capable of recognising objects categories from point cloud data. In this approach, Fast Point Feature Histogram (FPFH) features are extracted from point clouds to describe the local 3D shape of objects and a Bag-of-Words coding method is used to obtain an object-level vocabulary representation. Multi-class Gaussian Process classification is employed to provide a probability estimate of the identity of the object and serves the key role of modelling perception confidence in the interactive perception cycle. The interaction stage is responsible for invoking the appropriate action skills as required to confirm the identity of an observed object with high confidence as a result of executing multiple perception-action cycles. The recognition accuracy of the proposed perception model has been validated based on simulation input data using both Support Vector Machine (SVM) and GP based multi-class classifiers. Results obtained during this investigation demonstrate that by using a GP-based classifier, it is possible to obtain true positive classification rates of up to 80%. Experimental validation of the above semi-autonomous object sorting system shows that the proposed GP based interactive sorting approach outperforms random sorting by up to 30% when applied to scenes comprising configurations of household objects. Finally, a fully autonomous visual architecture is presented that has been developed to accommodate manipulation skills for an autonomous system to interact with the scene by object manipulation. This proposed visual architecture is mainly made of two stages: 1) A perception stage, that is a modified version of the aforementioned visual interaction model, 2) An interaction stage, that performs a set of ad-hoc actions relying on the information received from the perception stage. More specifically, the interaction stage simply reasons over the information (class label and associated probabilistic confidence score) received from perception stage to choose one of the following two actions: 1) An object class has been identified with high confidence, so remove from the scene and place it in the designated basket/bin for that particular class. 2) An object class has been identified with less probabilistic confidence, since from observation and inspired from the human behaviour of inspecting doubtful objects, an action is chosen to further investigate that object in order to confirm the objects identity by capturing more images from different views in isolation. The perception stage then processes these views, hence multiple perception-action/interaction cycles take place. From an application perspective, the task of autonomous category based objects sorting is performed and the experimental design for the task is described in detail.

Acknowledgements

I wish to acknowledge those mentors, friends and colleagues who have helped me with their technical advice or much needed fortitude to bring this work to life.

I would like to thank my supervisor Dr. Paul Siebert, who has given me the opportunity to become a member of Computer Vision and Autonomous System Laboratory. He has been a great mentor who have provided me with the guidance and support i needed to take up on this adventure and being inspired to dvelve into the field of vision for robotics.

I would like to thank my second supervisor Professor Phil Trinder, who have been providing me critical feedback.

I would extend my gratitude to Dr. Gerardo Aragon-Camarasa, who as a colleague supported and bear with me in all those endless discussions and as a friend, has been there in hard times.

I also thank Dr. Li Sun (kevin), a great friend, who has shared the fun and difficult times during my stay at the lab. I would cherish all those memories.

I would extend my gratitude to Iliyana (lil), who has been an inspiration, shared fun moments and great support through the hardest times.

Finally I would like to thank my parents for their unconditional support, love, and their believe in me. They have been the greatest inspiration and motivation for all the success i have achieved and endured difficult times in my life.

Table of Contents

List of Tables

List of Figures

Abbreviations

GP = Gaussian Process

SVM = Support Vector Machine

SIFT = Scale Invariant Feature Transform

SURF = Speeded up Robust Features

RGBD = Red Green Blue Depth

FPFH = Fast Point Features Histogram

CloPeMa = Clothes Perception and Manipulation

DSLR = Digital Single-Lens Reflex

PTU = Pan Tilt Unit

ROS = Robot Operating System

URDF = Uniform Robot Description Form

TF = Transform

CPU = Central Processing Unit

SEAs = Series Elastic Actuators

DOF = Degree-of-Freedom

SVGA = Super Video Graphics Array

SDK = Software Development Kit

MSER = Maximally Stable Extremal Region

ICE = Iterative Clustering Estimation

DSI = Data Systems International

BP = Belief Propagation

GC = Graph Cut

ICP = Iterative Closest Point

PCA = Principal Component Analysis

SLAM = Simultaneous Localization and Mapping

EKF = Extended Kalman Filter

PTAM = Parallel Tracking and Mapping

DTAM = Direct Tracking and Mapping

LBP = Local Binary Patterns

LoG or LOG = Laplacian of Gaussian

HOG = Histogram of Oriented Gradients

BRIEF = Binary Robust Independent Elementary Features

BRISK = Binary Robust Invariant Scalable Keypoint

ORB = Oriented Fast and Rotated BRIEF

NARF = Normal Aligned Radial Feature

HONV = Histogram of Oriented Normal Vectors

PFH = Point Feature Histogram

BoF = Bag-of-Features

GMM = Gaussian Mixture Model

VQ = Vector Quantization

kNN = k-Nearest-Neighbour

CNN = Convolutional Neural Network

SFX = Sensor Fusion Effects

MSCOI = Multiple Same-Class Object Instance

IOR = Inhibition of Return

URDF = Unified Robot Description Format

PnP = Perspective-n-Point

RANSAC = Random Sample Consensus

RMSE = Root Mean Square Error

GP-IPM = Gaussian Process Classification based Interactive Perception Model

PCL = Point Cloud Library

RBF = Radial Basis Function

SEiso = Square Exponential kernel function

BFGS = Broyden-Fletcher-Goldfarb-Shanno

RVIZ = ROS Visualization

SPFH = Simplified Point Feature Histograms

DCNN-GPC = Deep Convolutional Neural Network Gaussian Process Classification

Chapter 1

Introduction

This thesis presents a novel and generic visual architecture for scene understanding by robotic interactive perception. In this proposed visual architecture, an autonomous system interacts with the scene environment to be able to have a better understanding of what it perceives in the scene and what action it can do to improve it. Moreover, this thesis presents finding from investigating the effect of autonomous system interaction with scene environment from two aspects: On one hand, the vision sensor directs its gaze for scene exploration task, on the other hand, objects in the scene are repositioned by robotic manipulation skills. For vision sensor interaction with the scene environment, a portable active vision system performs scene exploration tasks, is devised and validated. For autonomous robot system interacting with scene environment, a visual perception architecture is proposed for scene understanding by interactive perception. This proposed visual architecture is fully integrated into autonomous systems performing object perception and manipulation tasks. More specifically, from the application perspective, this thesis makes refinement to the existing autonomous objects sorting pipeline by including the procedures of recognizing, confirmation of recognition and sorting objects. From the scientific perspective, this thesis contributes advancement into scene understanding by robotic interaction perception over once shot recognition without employment of rigorous supervised learning methods. In summary, this thesis advances the state-of-the-art for scene understanding by robotic interactive perception.

1.1 Motivation and objectives

In robotic scene understanding, the existing techniques are able to perform one or some of the following operations: learning object's appearance, visual exploration - what and where is the object, identification, segmentation, recognition, localization of objects and manipulation of objects by grasping. However, these techniques focus on one or a few aspects of the scene understanding pipeline.

For the task of visual exploration for scene understanding, active robot vision systems have been developed to perform actions and fulfill tasks by exploiting recovered information from the imaged scene, such as [Ballard, 1991]. Typically, active robot vision systems are comprised of hard-wired, ad-hoc visual functions and the intended purpose is to have the capability of robustly exploring a scene and finding objects contained in a database of pre-trained object examples as reported by [Chen et al., 2011] and [Collet et al., 2011]. An active binocular robot head architecture [Aragon-Camarasa and Siebert, 2009, Aragon-Camarasa et al., 2010] is devised that is able to execute vergence, localization, recognition and simultaneous identification of multiple target object instances in a scene, however, its lack of portability and robustness in terms of consistent performance constrain the scope of potential applications for such vision systems. Thus, there is a need to demonstrate the portability and functional accuracy of the reported system [Aragon-Camarasa and Siebert, 2009, Aragon-Camarasa et al., 2010].

For objects manipulation tasks by a robot, it is important to estimate the parameters of a lens and image sensor of an image/video camera, this process is known as camera calibration. These parameters are used to correct lens distortion, and to be able to measure the size of an object in world units in the scene. Additionally, the vision system needs to be integrated into the robot's kinematic chain, essential to perform accurate manipulation tasks. Pre-dominantly, two types of approaches are used for camera calibration: Rigid calibration methods [Kwon et al., 2007, Neubert and Ferrier, 2002, Salvi et al., 2002] and hybrid approaches such [Mueller and Wuensche, 2016]. However, these approaches fall short in terms of accuracy and precision performance since they attain an overall precision within 0.5cm and 1cm [Hansen et al., 2012]. In view of the said limitations present in these approaches, a reliable and accurate camera calibration and the hand-eye calibration is needed.

Variety of visually guided scene understanding techniques have been investigated such as [Fitzpatrick, 2003, Gemignani et al., 2016, Gupta et al., 2015, Li, 2011]. These techniques fall short in terms of generic visual competence and are limited to objects of specific shape and/or color. For scene understanding by robotic interactive perception, it is important to determine, when the object in the scene needs interaction. In this thesis, the task of scene understanding is limited to what objects are present in a given scene, and what kind of interaction is needed to achieve the objective of a task such as objects sorting according to categories. For this purpose, the classification stage of the visual architecture needs to have the capability of providing prediction confidence in addition to providing the class label. This prediction confidence can form the basis to reason, whether the predictive class needs to improve its confidence by interacting with the object or not. Gaussian Process(GP) based multi-class classifier is better suitable for this kind of tasks than SVM like multi-class classifier.

Often, vision techniques in uncontrolled lighting and complex environments tend to yield low accuracy in understanding a scene without interaction or feedback. Since, understanding challenging scenes, single shot recognition fail or have low success rates while manipulating objects. In [Gibson, 1966], it is believed that physical interaction further augments perceptual processing beyond what it can be achieved by invoking deliberate pose changes. Therefore, the interaction with the scene by the vision system and/or interacting with the objects in the scene, can significantly improve the success rate while detecting object classes in front of a service robot. A high-level concept of interactive perception model is shown in Fig. 1.1.

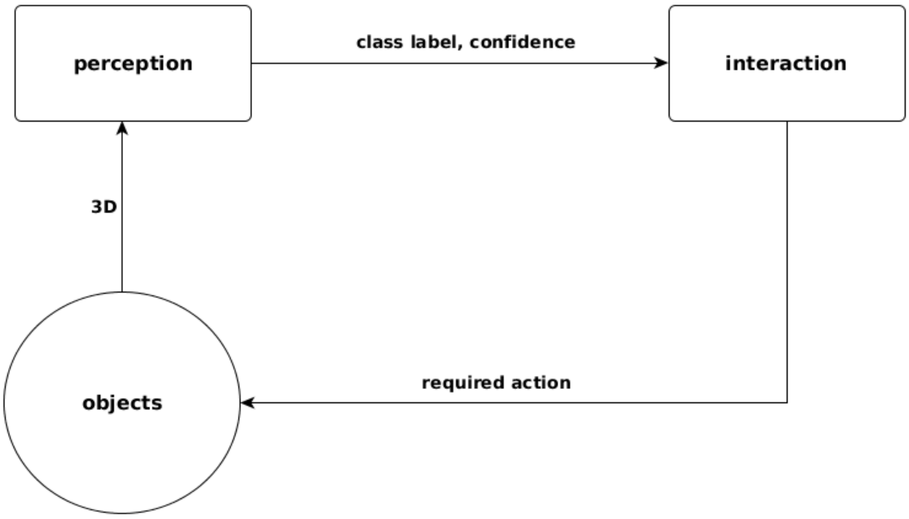


Figure 1.1: High-level concept of our interactive perception model for object sorting .

To the best of author's knowledge, a generic visual architecture capable of carrying out the task of scene understanding by robotic interaction in one single framework is desirable.

1.2 Thesis statement

The objective of this thesis is to advance the state-of-the-art in scene understanding by robotic interactive perception. Inspired by the mechanisms which appear to be operating in mammalian brains, an integrated autonomous robot system is devised following multiple perception-manipulation cycles. This thesis reports an investigation into the proposed visual perception architecture for scene understanding by robotic interaction from two perspectives:

1 From the perspective of a portable active vision system that can explore a scene in detail by following the underlying principle of structuring visual perception as what

and where systems, whose joint interaction affords the necessary visual understanding required to conduct robotic hand-eye grasping and manipulation tasks.

2 From the perspective of interacting with the object in the scene by repositioning it, in order to obtain new, potentially more diagnostic, views of it while performing the scene understanding task.

This work reported in this thesis builds on, and extends, an active binocular vision system reported by [Aragon-Camarasa and Siebert, 2009, Aragon- Camarasa et al., 2010]. Based on human intuition the following hypothesis is proposed:

An autonomous system endowed with visual capabilities must interact with the scene in its immediate vicinity in order to be capable of discovering what is there and what it can do with it. This mechanism is necessary to improve the autonomous systems overall understanding of the scene and its components. Accordingly, interaction with the scene is essential to allow any visual architecture integrated into an autonomous system for scene understanding to be able to resolve visually ambiguous observation situations of visual ambiguity.

The above thesis statement is further supported by the following descriptions:

- A visual perception architecture integrated with robotic manipulation skills performing object repositioning in the scene can improve the classification accuracy compared to non-interactive perception models.
- The base capability of the proposed visual perception architecture integrated into a fully autonomous robotic system can be demonstrated by performing objects sorting of household objects that requires an understanding of what (household object) is in the scene and what it can do with it (grasp & reposition the object).

1.3 A brief overview of the proposed approach

The proposed visual architecture for scene understanding by robotic interactive perception is two-fold: On one hand, the vision sensor interacts with the scene environment by gaze control to achieve scene exploration task, and on the other hand, objects in the scene are repositioned by robotic manipulation skills.

For scene exploration i.e what and where to look for, a portable active binocular robot head architecture is devised that is capable of vergence, localisation, recognition and simultaneous identification of multiple target object instances. The architecture of this vision system is shown in Fig. 1.2. In order to demonstrate the portability and functional accuracy of the proposed system, the proposed system has been validated with different - hardware, visual

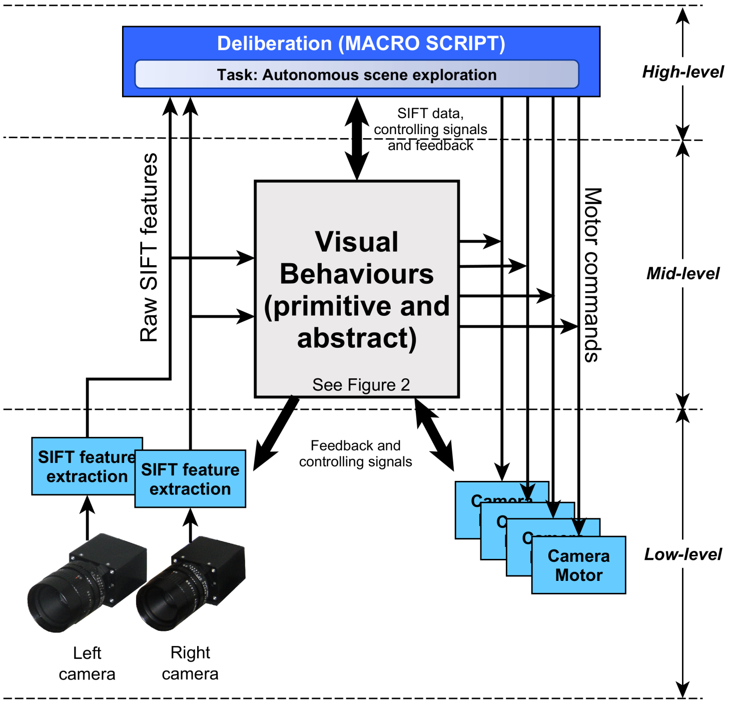

Deliberation (MACRO SCRIPT)
Task: Autonomous scene exploration
High-level
SIFT data, controlling signals and feedback
Raw SIFT features
Visual Behaviours (primitive and abstract)
See Figure 2
Motor commands
Mid-level
SIFT feature extraction
SIFT feature extraction
Feedback and controlling signals
Camera Motor
Low-level
Left camera
Right camera

Figure 1.2: Active binocular robot vision architecture [Aragon-Camarasa and Siebert, 2009, Aragon-Camarasa et al., 2010].

representation and view(s) of the scene. Hence, we present experiments with three different state-of-the-art feature extraction techniques, namely SIFT [Lowe, 2004b], SURF [Bay et al., 2008b] and KAZE [Alcantarilla et al., 2012] and, different hardware and scene settings.

This thesis also presents a methodology for the active robot head calibration and the hand-eye calibration, and, integration of the robot head and RGB-D cameras within a robot's kinematic frame. A simplified hand-eye calibration method is illustrated in Fig. 1.3.

Scene understanding by robotic interactive perception in terms of repositioning the object in the scene, a perception model is developed that is portable, invariant to 6 DOF object pose changes and operates in real-time as depicted in Fig. 1.4.

This proposed interactive perception pipeline comprises of object segmentation, visual representation, classification, semantic visualization and autonomous robotic manipulation. For classification task, a multi-class Gaussian process classifier is opted for equal or better classification accuracy compared to multi-class classifier such as SVM, but also, provides with classification confidence which plays an essential role in our proposed system.

From an application perspective, this thesis achieves the task of sorting and removing objects from a tabletop into the object's designated bins. After receiving point cloud from the Kinect Xbox camera from the scene, one shot recognition takes place. A list of objects identified in along with their respective classification confidences is maintained, such as a bottle or

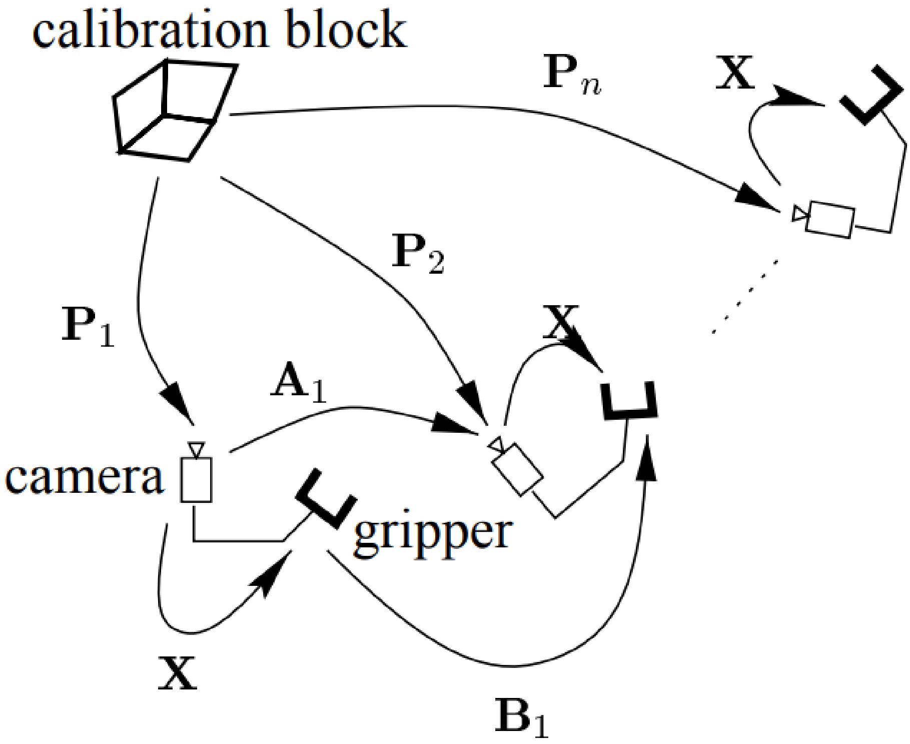


Figure 1.3: A general hand-eye calibration method is depicted. From images of a calibration object, the corresponding camera poses Pi can be computed, providing the rigid motions of the camera Ai . With the relative recorded robot motions Bi , the hand-eye transformation is computed.

juice box, which can potentially be partially occluded, have been influenced by varying lighting, background. Based on the classification label along with the object's corresponding classification values (i.e. confidence scores), objects are picked-and-placed into bins, each bin designated for an object category. For objects with a classification confidence score lower than a pre-defined threshold, this object is actively explored by manipulating and interacting with the object. The method repeats until all objects' classification confidence is higher than a predefined threshold and removed from the table top into each object's designated bin.

1.4 Contributions

The major contributions of this thesis are summarised as follows:

- A portable active binocular robot vision architecture is proposed that integrates a number of visual behaviors extending the work reported by [Aragon-Camarasa and Siebert, 2009, Aragon-Camarasa et al., 2010]. This vision architecture inherits the abilities of vergence, localization, recognition and simultaneous identification of multiple target object instances.

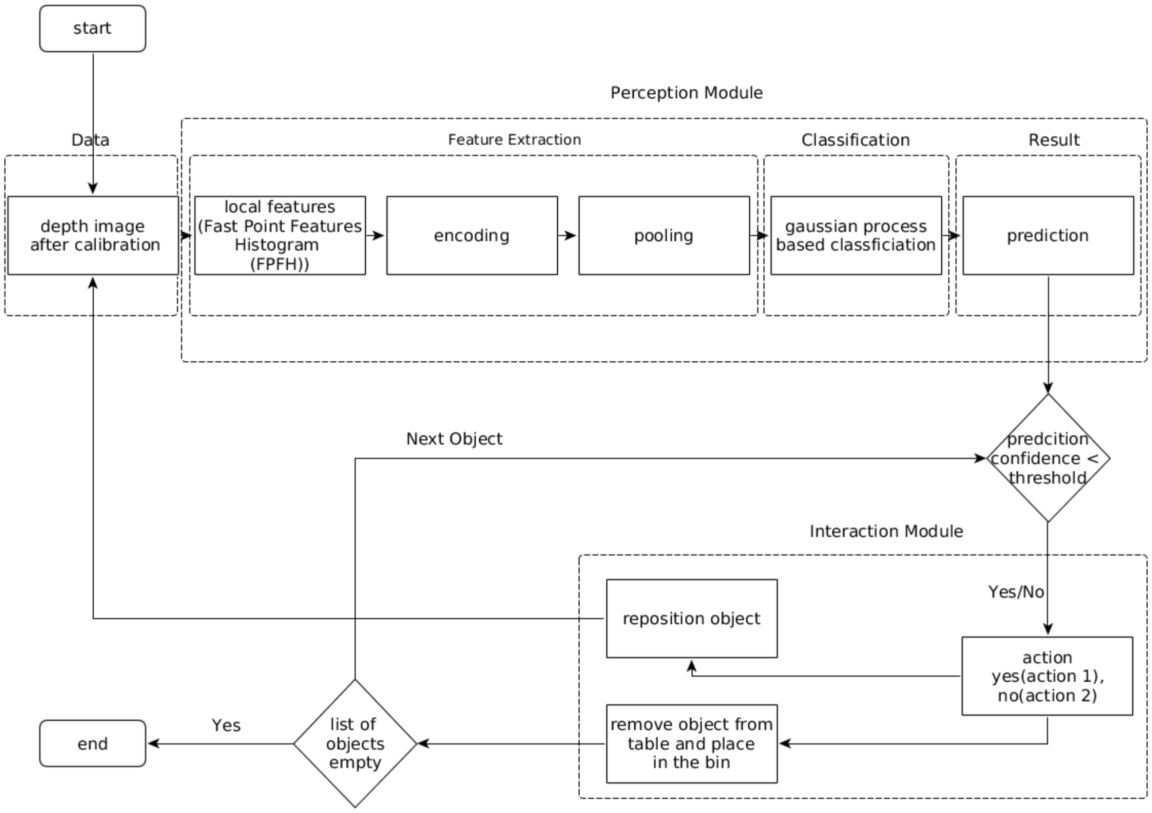


Figure 1.4: Visual Perception Architecture for Scene Understanding by Robotic Interactive Perception.

- A camera and hand-eye calibration methodology are described for integrating an active binocular robot head within a dual-arm robot. For this purpose, the forward kinematic model of our active robot head is derived and describe our methodology for calibrating and integrating the robot head is described. This rigid calibration provides a closed-form hand-to-eye solution. Moreover, an approach for updating dynamically camera external parameters for optimal 3D reconstruction that are the foundation for robotic tasks such as grasping and manipulating rigid and deformable objects. We show from experimental results that our robot head achieves an overall sub-millimetre accuracy of less than 0.3 millimetres while recovering the 3D structure of a scene.

- An interactive perception model based on Gaussian Process (GP) classification that is capable of recognizing objects categories from point cloud data. As an application for the use of the proposed approach, the task of objects sorting according to classes is accomplished. In our approach, besides the traditional object recognition pipeline employed, multi-class Gaussian Process classification is opted to provide a probable estimation of the identity of the object and serves a key role in the interactive perception cycle modeling perception confidence. From results, it is deduced that the proposed GP based interactive sorting approach outperforms random sorting by up to

30% when applied to scenes comprising configurations of household objects.

- The above proposed visual perception approach is extended and integrated with autonomous manipulation skills in a dual-arm robot testbed. A fully autonomous category-based visually-guided household object sorting system is devised.

1.4.1 Impact of this research

Work produced in this thesis has been used in parts in the following publications and project:

A Portable Active Vision System:

- Portable active bincular vision system made contributions to CloPeMa ("Clothes Perception and Manipulation", www.clopema.eu), which was a collaborative EU FP7 project involving University of Glasgow, CRTH[1], CVUT[2] and University of Geneva[3].

Impact of Interactive Perception Model based on Gaussian Process multi-class Classifier (Chapters 5 & 6).

- Base-line objects pick and place pipeline for work supported by InnovateUK under the iSee - Intelligent Vision for Grasping (Project No. 102913).

1.4.2 List of publications

The work described in this thesis has been presented in the following publications:

- Aamir Khan, Gerardo Aragon-Camarasa, Li Sun, Jan Paul Siebert (2016) On the Calibration of Active Binocular and RGB-D Vision Systems for Dual-Arm Robots. IEEE International Conference on Robotics and Biomimetics, DEC. 3 DEC. 7, 2016, Qingdao, China.

- Aamir Khan, Li Sun, Gerardo Aragon-Camarasa, Jan Paul Siebert (2016) Interactive Perception based on Gaussian Process Classification for House-Hold Objects Recognition & Sorting. IEEE International Conference on Robotics and Biomimetics, DEC. 3 DEC. 7, 2016, Qingdao, China

- Khan, A., Aragon-Camarasa, G., and Siebert, J. P. (2016) A Portable Active Binocular Robot Vision Architecture for Scene Exploration. In: 17th Towards Autonomous Robotic Systems (TAROS-16), Sheffield, UK, 28-30 June 2016

[1] Center for Research and Technology Hellas, http://www.certh.gr/root.en.aspx

[2] Ceske Vysoke Uceni Technicke V Praze, https://www.cvut.cz/en?set_language=en

[3] Universita Degli Studi Di Genova, https://unige.it/

- Sun, L., Aragon Camarasa, G., Khan, A., Rogers, S., and Siebert, P. (2016) A precise method for cloth configuration parsing applied to single-arm flattening. International Journal of Advanced Robotic Systems, 13, 70. (doi:10.5772/62513)

- Aamir Khan, Gerardo Aragon-Camarasa, Jan Paul Siebert (2017) Interactive Perception based on Gaussian Process Classification Applied to Household Object Recognition & Sorting. One Page Abstract, IROS 2017, Canada

- Aamir Khan, Gerardo Aragon-Camarasa, Jan Paul Siebert (2017) An Application of Interactive Perception based on Gaussian Process Classification for House-Hold objects Recognition & Sorting. 2nd UK Robotic Manipulation Workshop, Imperial College London (2017), UK

In preparation to submit journal paper:

- Khan, A., Aragon Camarasa, G., and Siebert, P. (2017/18) An Autonomous Household Objects Sorting Visual Perception Architecture based on Interactive Perception, International Journal of Advanced Robotic Systems

1.5 The structure of this thesis

The remainder of this thesis is organized as follows: Chapter 2 provides the background and a comprehensive literature review. The background introduces the CloPeMa robot, the stereo vision system, Baxter Robot and the literature review covers depth sensing, depth data analysis, object recognition, and the state-of-the-art for scene understanding by robotic interactive perception and manipulation. Following, the achievements of this thesis are presented in the following four chapters. Chapter 3 presents a portable active vision system for scene exploration to investigate the effect of an active vision system for scene understanding. In Chapter 4, the methodology for the active robot head calibration and the hand-eye calibration, and, integration of the robot head and RGB-D cameras within a robot's kinematic frame. In Chapter 5, an interactive perception model based on Gaussian Process-based multi-class classifier is presented, to provide visual guidance for scene understanding by robotic interactive perception, in terms of robot's interaction with the object(s) in the scene by performing objects' sorting. Chapter 6 provides a fully autonomous interactive perception model to sort household objects into their designated category based bins without human intervention. Finally, the conclusion and future work are detailed in Chapter 7.

Chapter 2

Background and Literature Review

This chapter presents a comprehensive review of the previously reported theories, methodologies, and applications relevant to the proposed research. For the development of a novel integrated robotic framework for scene understanding by the aid of interaction requires a wide range of research in hardware and software modalities, computer/robotic vision, and robotic skills. Before a detailed literature review is compiled, a brief background is presented with regard to already available existing hardware and software technologies resident at the lab. This chapter can be grouped into the background of the research, computer/robot vision and interactive perception. The background describes different hardware and software platforms used during the research. The background is followed by the review in the field of computer/robot vision relevant to the proposed research. Research areas such as computer/robot vision, object recognition in general and specifically, robotic object recognition are discussed. Interactive perception review provides a supporting literature arguing the benefits of interactive perception for a better understanding of a scene. The chapter ends with the summary of literature review carried out during the course of the proposed research, in terms of the limitation in the current state of the art and discusses, how to advance the state-of-the-art.

The organization of this chapter is as follows: Section 2.1 details the background. Then literature review supporting the proposed research follows. Section 2.2 describes the overview of some of the vision system used in robotic tasks. This section also discusses vision paradigms and the literature review specific to active binocular vision systems. Review of depth sensing for robot manipulation is given in section 2.4. Overview of some of the techniques used to analyze depth data obtained from depth sensors in the context of robotic scene understanding is given in section 2.5. Section 2.6 provides overview of classic object recognition pipeline and different stages constituting the pipeline. Section 2.3 details briefly the related work in the field of camera calibration and hand-eye calibration necessary to achieve robotic manipulations tasks. Section 2.7 provides an overview of literature supporting the use of in-

teractive perception in the context of robotic scene understanding. The chapter ends with summarizing the chapter by stating the limitations in the current state-of-the-art techniques that are able to achieve the task of scene understanding by robotic interactive perception and provides steps, for how to advance by overcoming some of the limitations which are then discussed in detail in the following chapter in section 2.8.

2.1 Background

This section is concerned with hardware and software technologies that have been available for research at the computer vision and autonomous systems laboratory, school of computing science, University of Glasgow, UK. In particular, these hardware and software technologies have been used extensively in producing the research presented in this thesis.

Overview

For the task of scene understanding by robotic interactive perception, a reliable dual-arm robot is useful while exploring the scene with changing viewpoint from vision sensor mounted on one of the arms and manipulating the object in the scene with another. For investigating the benefits of interactive perception in scene understanding, it has been the motivation for use of the dual-arm robots.

2.1.1 Yaskawa Motoman robot

Yaskawa Motoman robot or CloPeMa robot is an off-the-shelf robot that is designed for clothes perception and manipulation. The robot is shown in Fig. 2.1 that has been acquired via the CloPeMa ("Clothes Perception and Manipulation", `www.clopema.eu`), which was a collaborative EU FP7 project involving University of Glasgow, CRTH[1], CVUT[2] and University of Geneva[3]. CloPeMa project aimed at advancing the state of the art in the artificial perception and dexterous manipulation of clothes and other textiles.

During the course of the project (from 2012.2 to 2015.2), tactile sensing, visual sensing, and soft materials manipulation were jointly managed by a goal driven, high-level reasoning module. For further details about some of these real-life autonomous laundering problems are addressed in dual-arm garment folding [Stria et al., 2014], unfolding [Doumanoglou et al., 2014a,b], dual-arm flattening [Sun et al., 2015c], interactive sorting [Sun et al., 2016a]

[1] Center for Research and Technology Hellas, http://www.certh.gr/root.en.aspx

[2] Ceske Vysoke Uceni Technicke V Praze, `https://www.cvut.cz/en?set_language=en`

[3] Universita Degli Studi Di Genova, `https://unige.it/`

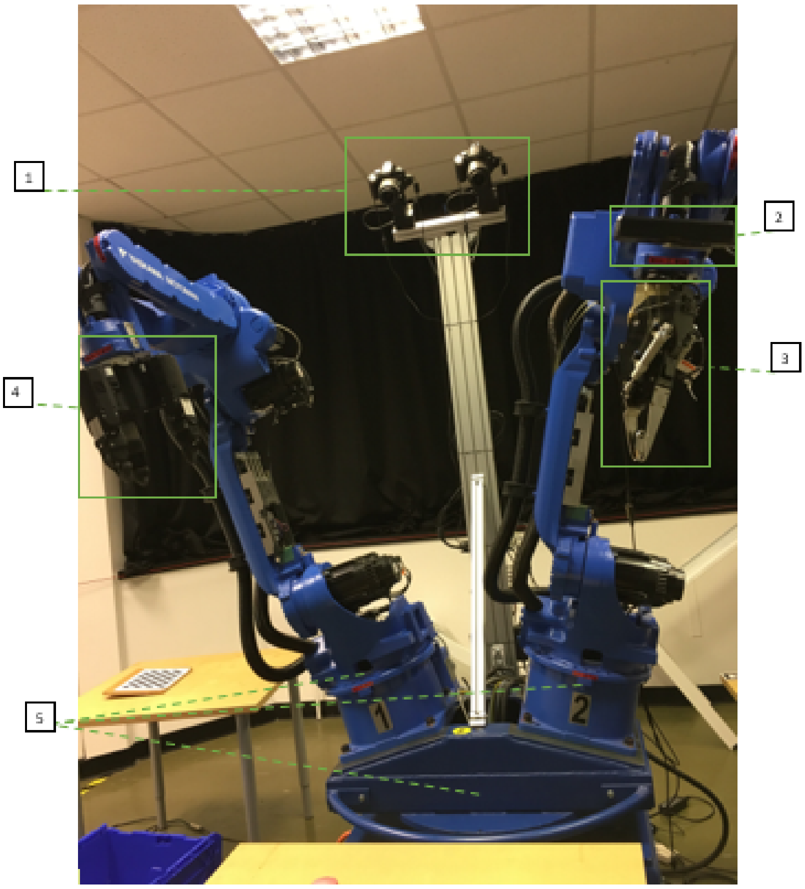


Figure 2.1: The Yaskawa Motoman/CloPeMa robot. 1. Stereo Head, 2. ASUS Xtion Pro, 3. CloPeMa Gripper, 4. The Shadow Smart Grasping System, 5. Dual Arms and Turning base

and a novel gripper design [Thuy-Hong-Loan Le et al., 2013]. The robot is comprised of the main robot body, robot head, robot grippers and other sensors, which are introduced in the following sections.

Robot body

The main robot body is made of the industrial robotic components for welding operation which are supplied by YASKAWA Motoman. As shown in Fig. 2.1, two MA1400 manipulators are used as two robot arms. Each manipulator is of 6 DOF, 4 kg maximal load weight, 1434 mm maximal reaching distance, ±0.08 mm accuracy. They are mounted on rotatable turning tables. The robot arms and turning table are powered and controlled by DX100 controllers.

Robot head

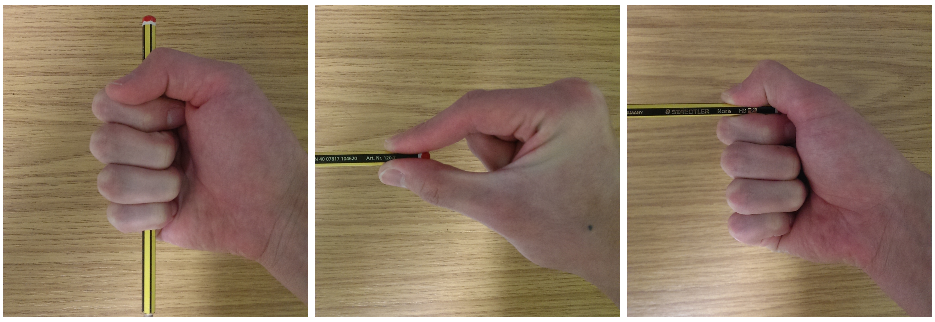
The robot head comprises two Nikon DSLR cameras (D5100) that are able to capture images of 16 megapixels through USB control. Gphoto library[4] is employed to drive the capturing under ubuntu. These are mounted on two pan and tilt units (PTU-D46) with their corresponding controllers. A pre-defined baseline separates the cameras for optimal stereo capturing, yet keeping its field of view to cover the robot workspace.

The aim for designing a robot head was to use relatively inexpensive, commercially available component elements, to build an off-the-shelf robot vision system (binocular head) for image sensing. The goal of this is to offset the limitation of widely-used depth sensors such as Microsoft Kinect with respect to accuracy and resolution.

Robot grippers

Yaskawa Motoman robot is equipped with two different types of end effectors. Details of each of these end effectors, also referred to as grippers, are as follows:

CloPeMa Gripper Inspired by the way humans grip, the CloPeMa gripper is designed mechanically to imitate the human's precision grip. In the book 'Examination of the Hand and Wrist' [Tubiana et al., 1998], human grips have been categorized into power grip (digitopalmar grip), precision grip (thumb-finger pinch) and lateral pinch. The robot's dexterous manipulation of clothes is more likely to imitate the human's precision grip. Hence, the CloPeMa gripper is designed as two thin fingers powered by liquid-pressure and with tactile sensors integrated on the tips.



(a) A demonstration of the power grip.

(b) A demonstration of the thumb-finger pinch.

(c) A demonstration of the lateral pinch.

Figure 2.2: Three types of human grip [Sun, 2016].

The CloPeMa prototype gripper has been developed by University of Genoa (UNIGE) based on Schunk grippers. Furthermore, the gripper has patented variable stiffness actuators for

[4] http://gphoto.sourceforge.net/

adapting to different grasping and steering tasks. However, this gripper is unable to perform grasping tasks for rigid objects.

The Shadow Smart Grasping System the Shadow Smart Grasping System[5], is a grasper, with built-in intelligence as can be seen in Fig. 2.3.

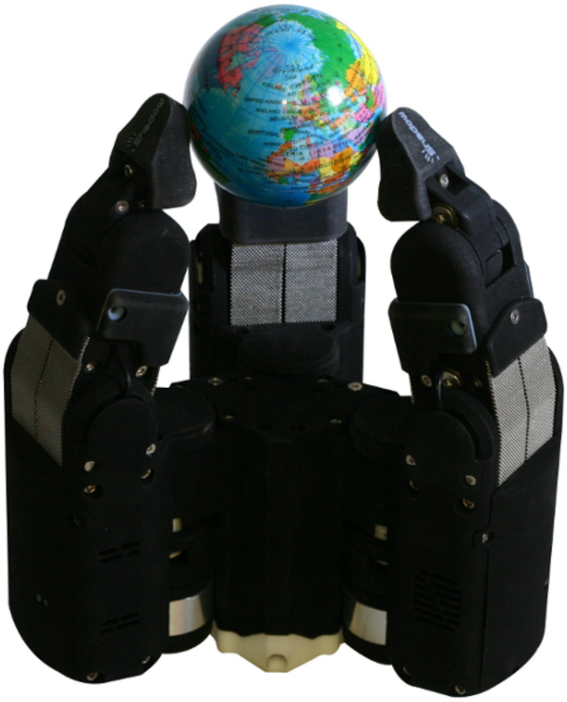


Figure 2.3: The Shadow's Smart Grasping System.

Key features of this smart grasping system are briefly detailed. A library of different grasps, enabling one hand to pick up many types of objects. Torque sensing on each joint, an addition ensuring the hand can make the most accurate and reliable grasp. Manufacturers claim that it is incredibly robust and reliable, reducing the need for maintenance and repair.
Yaskawa Motoman robot recently being equipped with this type of gripper has made this robotic testbed suitable for the proposed research of scene understanding by robotic interactive perception containing rigid objects.

RGB-D sensors

The robot has been fitted with two ASUS Xtion Pro Live RGB-D cameras, that have been installed on the robot, one on each arm, for lightweight range sensing. These are used for depth sensing and play a pivotal role in the work reported in this thesis.

[5] https://www.shadowrobot.com

2.1.2 Robot control

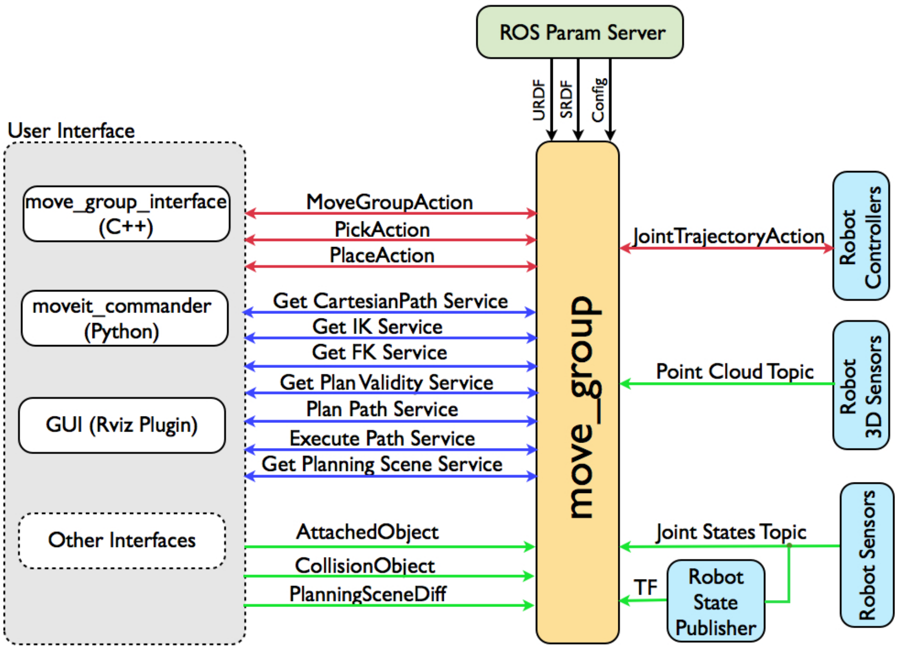


Figure 2.4: The working schema of Moveit.

The Yaskawa robot is fully integrated with Robot Operating System (ROS) through ROS industrial package[6]. In Fig. 2.4, the architecture of the robot control system is presented. The geometric structure of the robot is defined in the Uniform Robot Description Form (URDF). This enables the collision to be detected by robot collision models, and the transforms between robot links can be achieved by TF[7]. MoveIt package enables to achieve the communication between the user interface and robot controllers.

2.1.3 Stereo head calibration

Calibration of the stereo head installed on the dual-arm robot is briefly described. The stereo head calibration has two steps: camera calibration and hand-eye calibration. For the stereo head, Tsai's hand-eye calibration [Tsai and Lenz, 1989] routines are used to estimate rigid geometric transformations between camera to the chess board and chess board to the gripper. Details on camera calibration and hand-eye calibration are discussed in details in 4

[6] http://wiki.ros.org/Industrial

[7] http://wiki.ros.org/tf

2.1.4 Baxter robot

Baxter[8] is a humanoid, anthropomorphic robot, supporting two seven degree-of-freedom arms and state-of-the-art sensing technologies, including force, position, and torque sensing and control at every joint, cameras in support of computer vision applications, integrated user input and output elements such as a head-mounted display, buttons, knobs and more. Further, the Robot components are broken down to on a component by component basis and are briefly described.

Baxter robot body

The robot body[9] is equipped with onboard CPU containing processor of the 3rd generation Intel Core i7-3770 Processor (8MB, 3.4GHz) w/HD4000, graphics memory of 4GB, NON-ECC, 1600MHZ DDR3 and hard drive of 128GB Solid State Drive.

Baxter arms

Baxter's arms[10] are an integral part of "Baxter being Baxter" from the basic actuation premise of Series Elastic Actuators (SEAs) to available control modes, to the overall safety of the system. Baxter has two seven degree-of-freedom (DOF) arms. Seven DOF arms are desirable as they provide a kinematic redundancy greatly improving manipulability and safety.

Baxter grippers

The Rethink Electric Gripper[11] is a parallel jaw gripper meant for lifting payloads up to five pounds. The gripper hardware has a 44mm throw and attachment points for a variety of finger configurations. The gripper software supports a wide range of speeds and grip force levels and is fully upgradeable through Baxter.

Baxter cameras

The Baxter Research Robot has three color cameras[12]; one camera is located in each arm, and a third camera is located on the head, above the display screen. Any two cameras can be operated at a given time; USB system limitations do not allow all three cameras to be operated at once.

[8] http://sdk.rethinkrobotics.com/wiki/Baxter_Overview
[9] http://sdk.rethinkrobotics.com/wiki/Body
[10] http://sdk.rethinkrobotics.com/wiki/Arms
[11] http://sdk.rethinkrobotics.com/wiki/Grippers
[12] http://sdk.rethinkrobotics.com/wiki/Cameras

Baxter head

The Baxter Robot head display[13] is a 1024 x 600 SVGA LCD screen, capable of panning 180 and also nodding to acknowledge user input. Baxter's head has a panning joint and a single "Nod" action for movement. There is one camera, one red LED ring, one green LED ring and 12 individual yellow LEDs along with 12 sonar transducers.

Baxter RSDK

The Baxter Research Robot SDK[14] provides a software interface allowing researchers of all disciplines to develop custom applications to run on the Baxter platform. The SDK interfaces with the Baxter Research Robot via ROS (Robot Operating System)

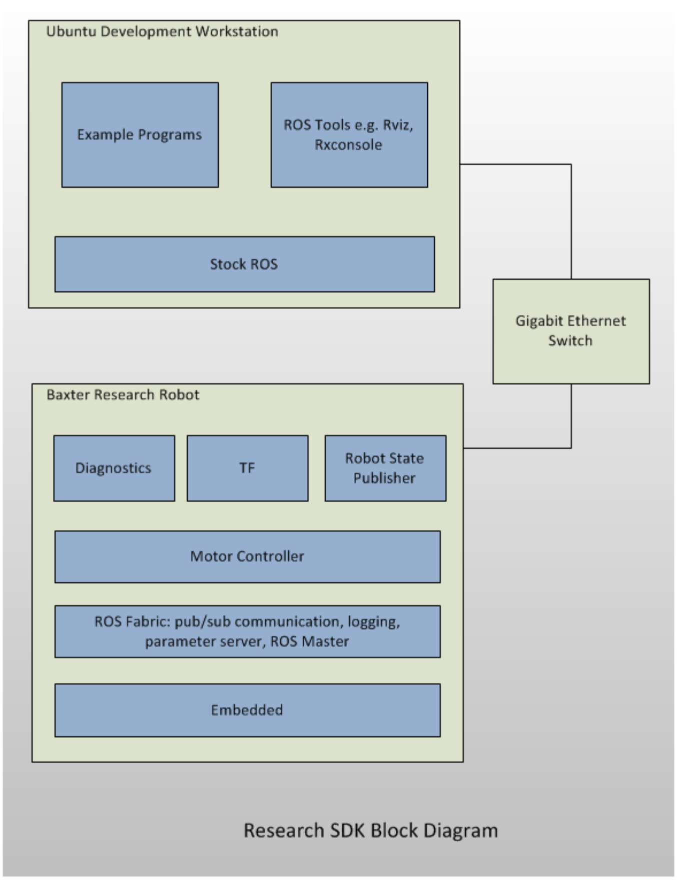

Ubuntu Development Workstation
Example Programs
ROS Tools e.g. Rviz, Rxconsole
Stock ROS
Gigabit Ethernet Switch
Baxter Research Robot
Diagnostics
TF
Robot State Publisher
Motor Controller
ROS Fabric: pub/sub communication, logging, parameter server, ROS Master
Embedded
Research SDK Block Diagram

Figure 2.5: Research SDK Block Diagram.

[13] `http://sdk.rethinkrobotics.com/wiki/Head`

[14] `http://sdk.rethinkrobotics.com/wiki/Baxter_Research_Robot_Software_Developers_Kit_(SDK)`

Literature review

Rest of this chapter is organized to present literature review addressing research areas such as computer/robot vision, object recognition in general and specifically robotic object recognition is discussed.

2.2 Vision systems

In this section, a study of different robot heads, robotic vision paradigms and an active vision system (reported in detail in chapter 3) for robotic tasks are presented.

2.2.1 Robot vision and vision paradigm

Visual Processing in many computer vision approaches is segregated into two complementary stages i.e early vision and cognitive vision. Early vision constitutes the use of image processing algorithms to extract features from given image sequences. Development of sophisticated algorithms in this area has led to the design of a variety of features possessing different invariance qualities. As a second stage, cognitive vision is the processing of high-level visual entities to solve complex tasks such as navigation, planning, and surveillance. The semantic gap between the visual features obtained from the early vision and high-level concepts needed by the cognitive vision is a fundamental challenge for the vision community.

Machines which can perform predefined tasks autonomously or guided are called Robots. Robots are equipped with a variety of sensors to achieve such tasks, such as haptic, positional, ultrasonic range sensors etc. For perception of the surrounding environment, vision is considered to be the most powerful sensor which is responsible for acquiring and extracting visual information from digital images of the observed world to carry out further tasks of acting and interaction within a dynamic world according to nature of the task.

Passive and Active Vision are two paradigms that lay out the foundation for robot vision systems. Passive vision can be termed as the measurement of visible radiation that is already present in the scene and has been a dominant approach as exemplified in the work of Marr [Marr et al., 1983]. Based on passive vision, several prominent approaches are reported namely stereo [Okutomi and Kanade, 1993], structure from motion [Tomasi and Kanade, 1992], shape from shading [Horn, 1970, Horn and Brooks, 1989, L. B. Wolff and Healey], photometric stereo [Woodham, 1980] etc.

In the robotics context, passive vision produces a pictorial view of the world which is captured either by means of a single or sequences of frames and the visual information is processed and analyzed without orientating the sensor. Based on passive vision, Walther et al.

[Walther et al., 2005] developed a system in order to learn visual features of the salient region based on a solely bottom-up approach for robot localization using a fixed time interval by means of a static camera while the robot navigates a pre-defined path.
Active vision system allows the observer to obtain information when needed which is based on the principle that the environment is dynamic and observers should be capable of engaging with it [Findlay and Gilchrist, 2001]. In recent years, [Rasolzadeh et al., 2009] reports an active vision system which is capable of finding, attending and manipulating objects where aspects of top-down and bottom-up attention and in addition with foveated attention are utilized for robotic object grasping.

2.2.2 Visual attention

Human vision is binocular and both eyes move according to vergence mechanism to focus on a fixation point in the real world, in order to maximize the extraction of visual information being extracted after vergence. From the perspective of perception-action cycle, eyes of an observer could be directed to interesting locations of the perceived environment [Posner and Cohen, 1984]. Besides the attentional spotlight metaphor, two additional independent and hierarchical modalities of analysis have been suggested which run parallel and bidirectional with respect to their outputs, namely pre-attentive and attentive. Pre-attentive is a covert attention which operates without changing the gaze while governed by top-down(goal-driven) and bottom-up(stimuli-driven) cueing mechanism [Chun and Wolfe, 2004]. At the attentive stage, one object is the focal point at a given time [Chun and Wolfe, 2004] while inhibition of return keeps the system free from visiting the same object again and again indefinitely.

2.2.3 Active binocular vision system

For the purpose of scene understanding by robotic interactive perception, the research reported in this thesis considered an active vision system for scene exploration, where the vision system directs its gaze to target salient regions and objects in the scene remain static and are not being manipulated, details reported in chapter 3. Chapter 5 and chapter 6 investigate the effect of repositioning the objects in the scene by robotic manipulation while the position of the vision system is constant. This section presents a review of the active vision system as follows. In the robotic vision, active vision can potentially offer a sheer amount of information about the robot's environment. Should a visual task becomes ill-posed, the gaze of a robot can be shifted to perceive the scene from a different viewpoint [Ballard, 1991], and therefore a better understanding of the task. Current research in active robot heads has focused on the *"lost and found"* problem [Meger et al., 2010]. That is, a robot is commanded, to search and locate an object in its working environment for exploration tasks [Aydemir and

Jensfelt, 2012, Collet et al., 2011], manipulation tasks [Rasolzadeh et al., 2010, Sun et al., 2015c] and/or navigation [Meger et al., 2010].

In an effort to replicate the nature of visual search scan paths [Wolfe and Whitney, 2015], researchers have proposed a variety of visual search mechanisms according to the task at hand (e.g. [Ma et al., 2011, Meger et al., 2010, Rasolzadeh et al., 2010]). These heuristic approaches are mainly driven by the outputs of available feature extraction techniques. For example, [Rasolzadeh et al., 2010] used depth to segment the scene according to the distance between a targeted object and the robot as part of a visual object search heuristic. Likewise, [Meger et al., 2010] implemented a saliency map that combines intensity, colour and depth feature to drive attention, biased by a top-down feature detection, based on the MSER feature extractor [Matas et al., 2002] for object recognition and navigation. [Aydemir and Jensfelt, 2012] have recently presented a strong correlation between local 3D structure and object placement in everyday scenes. By exploiting the relationship between local 3D structure and different object classes, the authors are able to localize and recognize complex 3D objects without implementing specialized visual search routines. Finally, [Collet et al., 2011] have proposed an Iterative Clustering Estimation (ICE) algorithm that combines feature clustering along with robust pose estimation. This approach relies on creating sparse 3D models to localize and detect multiple same-class object instances. Advancements in visual search mechanisms have been promising in recent years of which they are not merely restricted to the feature extraction used, and rather powered by cognition. For instance, a notable approach proposed in [Dogar et al., 2014], looks at the problem of a robot searching for an object by reasoning about an object and possible interactions with the object. However, this robot vision system is limited to one single instance per object class in the scene.

The vision architecture detailed in chapter 3, advances the robot vision system described in [Aragon-Camarasa and Siebert, 2009, Aragon-Camarasa et al., 2010]. That is, a previously reported, an active vision system that is capable of binocular vergence, localization, recognition [Aragon-Camarasa and Siebert, 2009, Aragon-Camarasa et al., 2010] and simultaneous identification of multiple target object instances [Aragon-Camarasa and Siebert, 2010]. This architecture was structured as a collection of ad-hoc functions in order to explore autonomously a scene by operating solely with SIFT features. This system was also constrained to the hardware and, therefore, the limitation of its portability remained an issue. Recent developments in robotic middleware (e.g. the Robot Operating System [Quigley et al., 2009]) technologies have made possible the deployment of hardware independent robotic systems. Therefore, there is a need to develop an active binocular robot head architecture that integrates visual behaviors in a parsimonious and generic robot vision architecture based on the Robot Operating System (ROS).

While the visual architecture does not make explicit use of 3D information, an explicit goal was to determine if it could reliably maintain binocular vergence of an actuated stereo-pair of cameras while actively exploring a scene. This converged binocular camera configuration supports the recovery of feature locations in 3D and also provide images for stereo-matching for dense 3D range map extraction. This feature underpins visual competences for other robotic applications as demonstrated in [Sun et al., 2015b] where it presented a dual-arm robot manipulating deformable objects using the binocular system reported in chapter 3.

2.3 Camera and hand eye calibration

For camera systems under dynamic actuation, either in a monocular or stereo configuration, researches have proposed rigid, continuous self-calibration and a combination of both. A summary of current approaches is provided below.

Rigid calibration methods consist of estimating intrinsic and extrinsic camera parameters, and the mechanical relationships between their actuation platforms and camera reference frames ([Kwon et al., 2007, Neubert and Ferrier, 2002, Salvi et al., 2002]). Optimisation routines such as bundle adjustment [Furukawa and Ponce, 2009], are applied to reduce back-projection errors from 2D to 3D measurements and, consequently, arrive at a closed-form and stable solution within a defined camera parameter space. However, there exist two main limitations for rigid approaches. On the one hand, mechanical wear and tear are not considered and the solution obtained depends on the quality of the mechanical parts at the moment, when the calibration was carried out. On the other, the distance and orientation between actuator joints have to be precisely measured. Hence, errors induced by the measuring device and mechanical backlash during their operation are not taken into account.

Self-calibration methods of PTU camera systems have been extensively researched for the last 20 years and is still an active research area. Nevertheless, a general and generic solution has yet to be devised. For instance, self-calibration approaches have constrained the solution to one degree of freedom for each camera [Sapienza et al., 2013], or individual yaw movements per camera plus a joint neck movement for the head [Dang et al., 2009]. Adding degrees of freedom, therefore, results in a lower precision of the reconstructed scene [Brückner et al., 2014]. To overcome constraints on the kinematic structure of the systems, hybrid approaches consist of initializing calibration parameters using rigid solutions and, then, update intrinsic and extrinsic camera parameters while interacting with the environment. These approaches are based on particle filtering techniques to update the camera and external orientation of the cameras with respect to a world reference frame [Mueller and Wuensche, 2016]. As our robot also handles highly-deformable objects, these approaches fall short in terms of accuracy and precision performance since they attain an overall precision within 0.5cm and 1cm [Hansen et al., 2012].

In order to overcome the above limitations, this thesis proposes a hybrid approach that handles dynamic content in the observed scene (detailed in Chapter 4). The approach consists of simple yet robust techniques. It must be noted that it is not an attempt to solve the general problem of self-calibration but provide a robust and general purpose approach to handle both deformable and non-deformable objects. It is therefore assumed that the vision system does not change its focus and the focal length is adjusted, aperture and shutter settings for each camera to obtain the desired focus range.

2.4 Depth sensing for robot manipulation

From here onwards, the remaining literature review is aimed towards the investigation of scene understanding by robotic interactive perception in terms of manipulating the object in the scene while the vision system remains constant.

2.4.1 Stereo-Matching based binocular cameras

Before Kinect-like cameras were available, stereo-vision based depth sensing had been widely investigated. It has been one of the goals in the community of computer vision to describe the world that we see based on the information obtained from one or more images and to restructure its properties, such as its illumination, shape, and color distributions. Stereo vision is the reconstruction of the three-dimensional coordinates of points for depth estimation. A stereo vision system consists of a stereo camera, namely, two cameras placed horizontally (i.e., one on the left and the other on the right). The stereo 3D reconstruction pipeline consists of: (1) capturing two images simultaneously by these cameras, (2) finding correspondences between pixels from the pair images in order to obtain disparity maps (i.e. stereo matching) and (3) reconstructing 3D scene from the disparities maps using calibrated camera parameters of the stereo system. The challenge lies in determining the method to best approximating the differences between the views shown in the two images to map (i.e., plot) the correspondence (i.e., disparity) of the environment.

Stereo vision disparity map algorithms can be classified into two categories, local or global approaches. Local approaches are also called as area-based or window based approaches.

Local methods generate the pixel correspondences by measuring the correspondence and similarity between image regions. A notably effective implementation is reported in [Tombari et al., 2008]. After calculating each candidate disparity value individually, winner takes all strategy is applied to achieve the assignment of disparity values. By averaging or summation over a support region, the matching cost function is aggregated. As a result, the disparity value associated with the minimum cost for each pixel is assigned to that pixel. [Mattoccia et al., 2010] reported an algorithm based on an efficient cost aggregation strategy and a refinement to this approach has been reported in [Psota et al., 2011] and [Yang, 2012].

Global approaches formulate the stereo matching problem as the minimization of the energy function incorporating the DSI error term and a smoothness term. Belief propagation (BP) and graph cut (GC) algorithms are best-known approaches in this category reported in [Prez and Snchez, 2011]. Applied to their 3D telepresence systems, they developed a real-time, high-definition algorithm by implementing two BP algorithms. The first BP instance performs classification of the pixels into areas designated as reliable, containing occlusion errors and texture-less to reduce the numbers of memory accesses required for these three groups of pixels. The second BP process decreases memory traffic by generating the final disparity map.

Based on particle filtering, [Ploumpis et al., 2015] developed an accurate stereo matching approach, for accurate stereo matching. The proposed method consists of three parts. First, multiple disparity maps are utilized in order to acquire a very distinctive set of features called control landmarks. It is followed by segmentation as a grouping technique. As a second step, to estimate the best disparity value, scan line particle filtering is applied using the corresponding landmarks as a virtual sensor data. Finally, the computational redundancy of particle filtering is reduced in stereo correspondence with a Markov chain model, given previous scan line values. Using this method, high-quality disparity maps can be produced.

It is worth noting that the active binocular vision system was suitable to achieve the task of scene exploration by actively interacting with the scene by changing its gaze to salient regions as evident by the results discussed in the chapter 3. However, to investigate the effect of interactive perception in terms of repositioning the object in a given scene by robotic manipulation and due to the limitation of low-quality depth data acquired for a rigid object being less suitable from the active binocular vision system, off the shelf Kinect-like cameras have been used.

2.4.2 Kinect-Like cameras

The most famous off-the-shelf depth sensors are Kinect-like cameras (including Mircosoft Kinect, Kinect one, Intel ZR300 and Asus Xtion Pro, Xtion 2), which have been widely used in RGB-D based recognition. The sensing range of Kinect-like cameras is 0.8m to 3.5m and sensing precision is approximately 5mm. Kinect achieves real-time depth sensing (30FPS), but for each image frame, the quality of depth map is limited. Recently ZED[15] stereo cameras have received attention for robotics application intended for outdoor use. Afterward, for achieving high-accuracy 3D reconstruction of static objects or indoor scenes, Kinect-fusion

[15] https://www.stereolabs.com/

[Izadi et al., 2011, Newcombe et al., 2011b] is proposed by merging multiple frames from a moving Kinect, in which the coarse-to-fine iterative closest point (ICP) algorithm is used to map dense point clouds in order to estimate the pose of the Kinect camera. A finer approach to incorporate a 3D model of a slow-moving object is produced in their subsequent work with dynamic fusion [Schmidt et al., 2015].

2.4.3 Discussion

Depending on the robotic application, both Kinect-like cameras and binocular-cameras have been widely-used in mobile robots and manipulation robots. As the performance of depth, sensing is determined by their stereo matching algorithms, the resident C3D correlation based algorithm [Siebert and Urquhart, 1995] is focused to be used for depth sensing of the clothes visual perception research(poor matches at the boundaries) with an active binocular vision system. While, rigid object manipulation scenes usually are of discontinuous depth, and hence, the depth changes drastically on the object boundaries. Hence, for the purpose of obtaining accurate 3D information of the rigid objects (to be repositioned) in the scene, Kinect-like cameras are suitable after a comprehensive literature investigation and experimental observation.

2.5 Depth data analysis

Visual perception is one of the most important components of an autonomous system. In this thesis, one of the objectives is to investigate the visual perception model that reasons and provides information for the interaction stage to interact with the scene for better understanding over one shot recognition. For such visually-guided manipulation tasks, RGB-D data (depth map or point cloud) is more robust than RGB data in terms of representing the shape, size, and landmarks of objects. Hence therefore a short and relevant survey on 2.5D/3D data techniques is provided.

2.5.1 Depth Image/Point Cloud Registration

Depth image/Point cloud registration is a task in computer and robotic vision, used to match two or more depth images/point clouds taken, for example, at different times, from different sensors, or from different viewpoints. This serves as an early stage in computer vision virtually for all large systems which evaluate these data. Also, methods based on depth image/point cloud registration are widely used to fuse multiple view depth maps to obtain a solid 3D scene/object. Basically depth map/point cloud registration techniques are used to estimate the rigid transforms as an early stage computer vision.

Principal component analysis (PCA) is often used in classification and compression techniques, to project data on a new orthonormal basis in the direction of the largest variance [Khan and Hasan, 2011, Yambor et al., 2002]. The direction of the largest variance corresponds to the largest eigenvector whereas the magnitude of this variance corresponds to eigenvalue. A rough registration can be obtained through alignment of the eigenvectors of their covariance matrices, provided that the covariance matrix of two point clouds differs from the identity matrix.

PCA based depth image/point cloud registration is sensitive to outliers as it does not minimize the Euclidean distance between corresponding points of the datasets. However, when point correspondences between the two point clouds are available then the minimization of sum of the Euclidean distances between these points corresponds to a linear least-squares problem that can be solved robustly using the singular value decomposition method [Marden and Guivant, 2012]. However, this technique requires perfect data. [Besl and McKay, 1992] proposed an iterative closest point (ICP) algorithm to register a scene point cloud to a model point cloud. For the ICP algorithm, a source point cloud and a target point cloud both serve as input. The algorithm starts with an initial registration by finding the point correspondences between these point clouds based on the nearest neighbour approach or a more sophisticated scheme e.g. geometrical features, color information etc.. Afterwards, at each iteration the transform estimated from the latest registration is applied on a scene point cloud, and the nearest point pairs are found using k-d trees [Friedman et al., 1977], and a new transform is estimated. This procedure continues until the registration error is lower than a pre-defined threshold. This approach works successfully provided that the rotation between the scene and model point clouds is reasonably small. It is worth noting that the approach reported by [Besl and McKay, 1992], is able to handle the registration problem of line segment sets, implicit curves, parametric curves, triangle sets, implicit surfaces and parametric surfaces.

In the following research, various kinds of 2.5D/3D local descriptors [Johnson and Hebert, 1999, Lo and Siebert, 2009, Rusu et al., 2009b, Steder et al., 2010, Tang et al., 2012] are proposed in order to acquire a better matching performance between two range images and point clouds. More details are introduced in Section 2.6.2.

2.5.2 Visual SLAM-Based 3D reconstruction

In recent years, Simultaneous Localization and Mapping (SLAM) using only cameras, has been actively discussed for the reasons that the sensor configuration is simple and the degree of technical difficulties is simpler than other sensors. Since the input is visual information only, such techniques are specifically known as visual SLAM. These approaches can further be grouped into monocular-SLAM and RGB-D SLAM, depending on the input data types.

First monocular visual SLAM, called MonoSLAM was reported in [Davison et al., 2007, Davison, 2003]. It is a filter-based SLAM algorithm where the camera motion and 3D structure of an unknown environment are simultaneously estimated using an extended Kalman filter (EKF). Visual features are used to estimate the camera frame transform, and camera states are modeled by 13 parameters, including 3D position, quaternion, motion, linear velocity and angular velocity. Limitation of this method is computational cost, that increases in proportion to the size of an environment, which is addressed in Parallel Tracking and Mapping (PTAM) [Klein and Murray, 2007] by splitting the tracking and the mapping into different threads on CPU.

In terms of monocular-SLAM, the transform between the current camera frame and keyframe is estimated, then a depth map is generated by triangulation and afterward merged with the global map. But during the pose tracking, errors are accumulated, which is the most difficult problem SLAM needs to resolve. For example, in Davison's classic monocular SLAM [Davison, 2003], the visual features are used to estimate the camera frame transform, and camera states are modelled by 13 parameters, including 3D position, quaternion, motion, linear velocity and angular velocity; Extended Kalman Filter (EKF) [Welch and Bishop, 1995] is used to reduce the transformation noise. [Engel et al., 2014, Newcombe et al., 2011a] are fully direct methods. In [Newcombe et al., 2011a], proposed direct tracking and mapping (DTAM), in which the tracking is done by comparing the input image with synthetic-view images generated from the reconstructed map. LSD-SLAM [Engel et al., 2014] is the state-of-the-art dense-mapping (feature-less) monocular SLAM, in which the camera pose transform is estimated by minimizing the photometric errors among dense pixels and Gauss-Newton method is employed to solve this optimization problem.

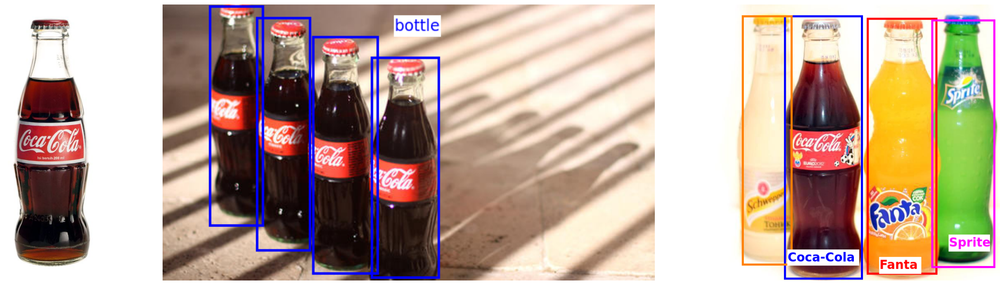
SLAM-based 3D reconstruction approaches are able to offset the drawbacks appearing in single-shot stereo vision, e.g. occlusions, while their prerequisite is that the image frames have to be captured from distinctive views.

2.6 The object recognition pipeline

Object recognition/identification is an extremely important problem in computer vision. Since this thesis presents object recognition pipeline for robotic interactive perception, in the context of scene understandings, the literature review about the object recognition pipeline is reviewed here.

2.6.1 Image classification, object detection and object recognition

Image classification usually is the task to find one major object in an image/scene shown in Fig. 2.6(a). Object detection is the task to localize the interested object(s) from a wider scope shown in Fig. 2.6(b). Sliding-window (exhaustive searching) [Dalal and Triggs, 2005b] is a classic approach for object detection, in which the small detecting window slides through the image over different scales, and for each position and scale, the window is classified whether it has an interesting object(s). The object recognition (identification) usually tends to recognize multi-class objects as shown in Fig. 2.6(c), the bottles can be identified by their brands. where the class information is 'finer' than that in object detection (e.g. children, adults, cars, vans, etc.).



(a) An image of a glass bottle.

(b) The detection/localisation of glass bottles.

(c) The recognition/identification of glass bottles.

Figure 2.6: The difference between image classification, object detection and object recognition [Sun, 2016].

This thesis presents more advanced component-based object recognition or pixel-level semantic scene labelling in the proposed visual architecture (details in Chapter 5 and 6). Section 2.6.2 introduces the image classification approaches from the early-stage to the state-of-the-art.

2.6.2 Image classification

In this section, some of the most widely used, Global and Local feature based approaches are reviewed. Review of other components such as encoding, pooling, classification, of the image classification pipeline is followed. For completeness, a brief overview of the recent advances in deep learning for image classification/recognition for robotic tasks is discussed.

Global Feature-Based approach

Global features have been widely used in many object recognition systems. These features are a very compact representation of images, making such features, attractive. Approaches

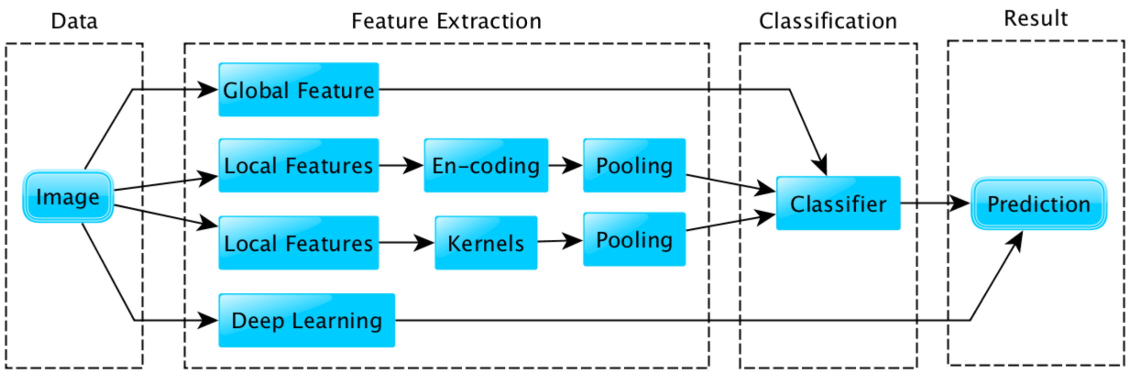


Figure 2.7: The image classification pipeline [Sun, 2016].

for global features representing each image corresponds to a point in a high dimensional feature space. Therefore, any standard classifier can be used. For example, Local Binary Patterns (LBP) descriptor is generated by comparing the values between a central pixel and its neighbors (usually 3×3 regions). These patterns can be counted into a histogram to obtain a global representation. Multi-scale LBP descriptor achieves outstanding performance on texture classification for small-scale datasets [Ojala et al., 2002]. Global features are sensitive to clutter, noise, and occlusion, and are not suitable for the work presented in this thesis.

Local Feature-based approach

Image feature extraction techniques reduces the dimensionality of the data in image and provide set of features which are used further for the desired task. A lot of research has matured in this area over past decades. Feature extraction has enabled the vision systems to identify different cues in a given image or sequence of images such as edges, corners, interest points, blobs, contours, motion cues etc.

Edge and corner detectors

Variety of edge detection techniques are developed among which Prewitt and Sobel were first developed which is based on computing the gradient to notice the changes in intensities where occur. Marr-Hildreth Edge detector [Marr and Hildreth, 1980] records the edges in an image by finding the Laplacian of Gaussian(LoG) and zero crossings. It has better resistance to noise and provides good localization. This method may result in too many responses. Canny edge detector [Canny, 1986] imposes single response constraint for each edge point by employing non-maximum suppression and hysteresis threshold.

An interest point is an expressive texture, where the direction of the boundary of an object changes abruptly or the point of intersection of two or more edge segments as found by the Harris corner detector [Harris and Stephens, 1988].

Histogram of oriented gradients (HOG)

Dalal and Triggs [Dalal and Triggs, 2005a] showed that with the use of grids of the histogram of oriented gradients outperformed previous feature sets used for human detection. The algorithm steps are:

- Compute centered horizontal and vertical gradients with no smoothing. Compute gradient orientation and magnitudes.
- For a color image, pick the color channel with the highest gradient magnitude for each pixel.
- For a 64x128 image,
- Divide the image into 16x16 blocks of 50 percent overlap. 7x15=105 blocks in total
- Each block should consist of 2x2 cells with size 8x8.
- Quantize the gradient orientation into 9 bins
- The vote is the gradient magnitude
- Interpolate votes between neighboring bin center.
- The vote can also be weighted with Gaussian to down-weight the pixels near the edges of the block.
- Concatenate histograms (Feature dimension: 105x4x9 = 3,780)

Visualization of the HOG for two input images is shown below on the right of their respective input image.

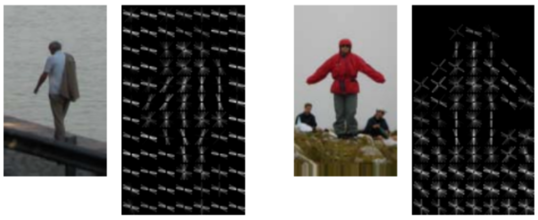


Figure 2.8: Visualization of the HOG for two input images

Scale Invariant Feature Transform (SIFT)

Feature matching across different images is usually a common problem in computer vision. If the images remain of the same scale then simple corner detection algorithms may be enough but for images with different scales, a scale-invariant feature transform is a possible solution reported in [Lowe, 1999] and the details and improved version in [Lowe, 2004a]. Histogram of Oriented Gradient is an optimal solution when features intended are of global need while SIFT performs better as local features because of computing the descriptor for localized keypoints and robust to scale and rotation variance and possibly to geometric transformations. Outline of the SIFT method is shortly discussed below.

- In order for the features of an image to be scale invariant, a scale space is generated.

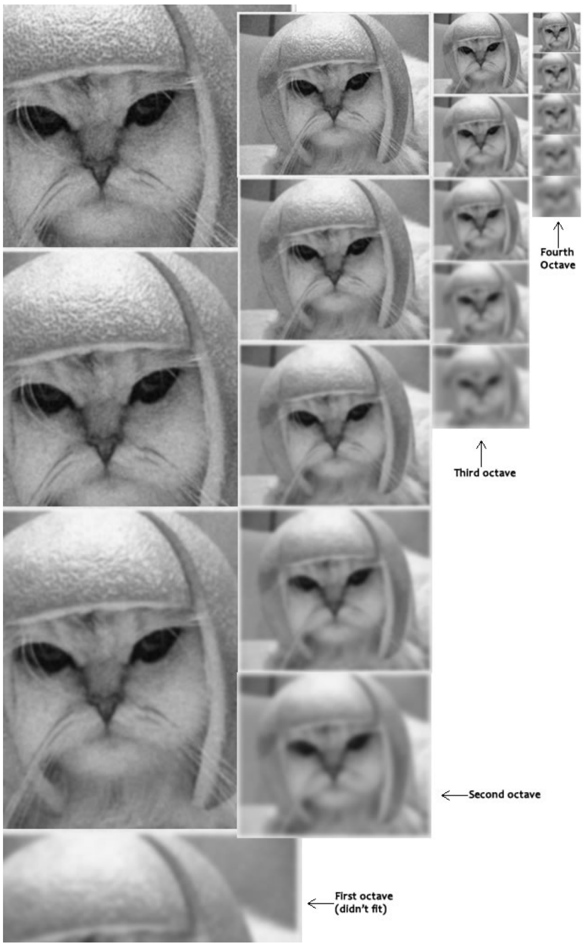


Figure 2.9: SIFT-Octaves

- Next step is to do the Laplacian of Gaussian (LOG) Approximation which is a repeated process of subtracting two consecutive images in octave and results in the approximation of scale-invariant LOG.
- After approximation of LOG keypoints are found which are many of these and some of them are removed which are along the edges or at low contrast.

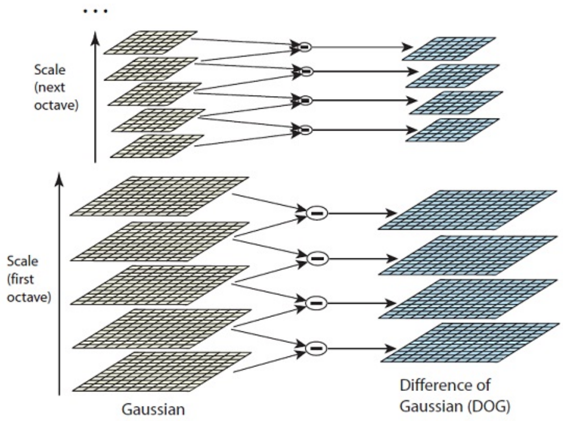


Figure 2.10: SIFT-Octaves

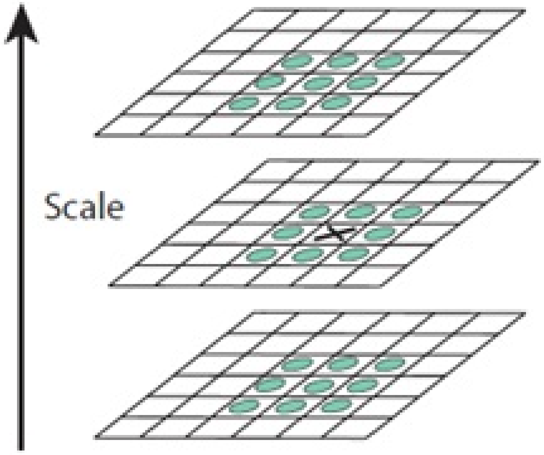


Figure 2.11: sift-maxima-idea

- Gradient magnitude and orientation for each key point are found to assign an orientation to each key point which makes them rotation invariant.

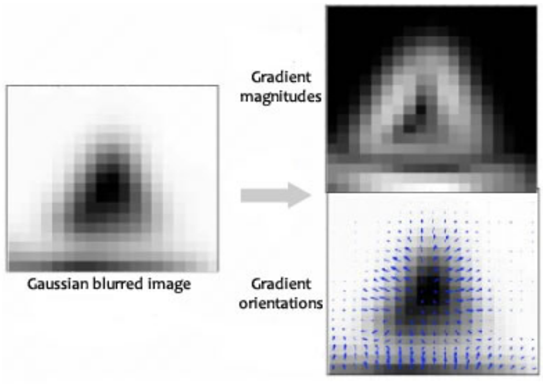


Figure 2.12: sift-orientation-window

- To generate the SIFT descriptor, a 16x16 window of in-between pixels for each key point is split into sixteen 4x4 windows. Next, from each 4x4 window, a histogram of 8 bins is generated corresponding to 0-44 degrees, 45-89 degrees, and so on. Gradient orientations from the 4x4 are put into these bins and finally, normalize the 128 values.

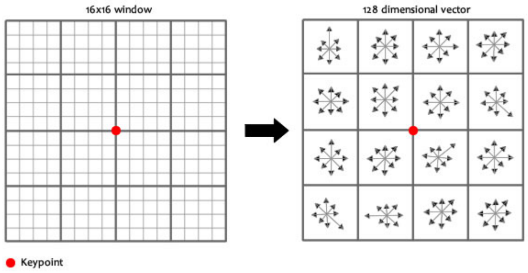


Figure 2.13: sift-fingerprint

Since the environment is dynamic and complex, visual competency needs to strengthen both its nature of being distinctive and computationally affordable. SIFT which has already been explained briefly above and Speeded Up Robust Features (SURF) [Bay et al., 2008a] are based on histograms of gradients. Computations of the gradients of each pixel in the patch cost time, though, SURF lesser than SIFT.
Binary string, encoding information of a path using only the comparison of intensity images and the use of hamming distance as a distance measure between two binary strings make computation very fast. Which is the rationale behind binary descriptors such as BRIEF: Binary Robust Independent Elementary Features [Calonder et al., 2010], ORB: an efficient alternative to SIFT or SURF and based on BRIEF [Rublee et al., 2011], BRISK: Binary Robust Invariant Scalable Keypoints [Leutenegger et al., 2011], Freak: Fast retina keypoint [Alahi et al., 2012].
The active binocular vision system (details in Chapter 3) has been tested with SIFT, SURF, and KAZE Features and established that it works reasonably good with these representations, however, SIFT and KAZE outperform SURF clearly. In robotics context, however, it is observed that information related to the depth of the object(s) is an important cue which needs to be addressed and as proved from various studies presented in the following sections that having 3D based matching and pose estimation gives better results. Typically, a range image can be termed in three types, namely a depth image, a point cloud or a polygonal mesh. The goal of 3D object recognition, given a range image, is to correctly identify objects present in the range image, and determine their poses i.e. position and orientation [Bariya and Nishino, 2010]. The following 3D local descriptors can also be used to describe the local components of 3D objects essential for 3D object recognition.

[Mokhtarian et al., 2001] used the Gaussian and mean curvatures to detect key-points. They declared a point p as a key-point if the curvature value of a point p was larger than the curvature values of its 1-ring neighbours (k-ring neighbours are defined as the points which are distant from p by k edges).

[Novatnack and Nishino, 2007] presented one of the first works to exploit geometric scale space on range image. In their approach, a surface is represented using its normals and parametrized the surface on a 2D plane to obtain a dense and regular 2D representation. Then by successively convolving the 2D normal map with geodesic Gaussian kernels of increasing standard deviations to construct a scale-space of the normal field. Key-points along their corresponding scales were detected by identifying the points in the scale-space.

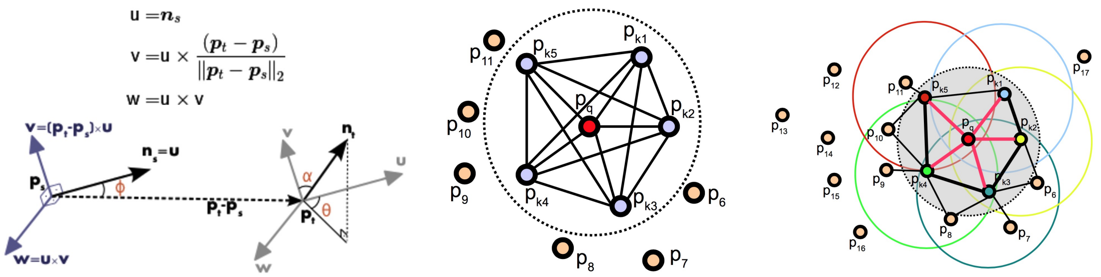
[Johnson and Hebert, 1999] reported a rotational invariant function representing Spin Image for point cloud or 3D mapping and 3D object recognition. The spin image is extracted at an original point, and each 3D point is projected within τ, υ where τ and υ provide the projected distance between the point and the origin on the tangent plane and the surface normal. The tau and $upsilon$ values are collected in uniform bins to create the 'Spin Image '.

In addition, [Lo and Siebert, 2009] extended the 2D SIFT [Lowe, 2004c] descriptor to a depth map. [Steder et al., 2010] presented a sparse local function NARF (Normal Aligned Radial Feature) on a depth map. NARF usually appears on the edges of the object where the shape of the object surface is unstable. [Tang et al., 2012] proposed the HONV (Histogram of Oriented Normal Vectors) descriptor by accumulating surface normal vectors in spherical coordinates.

Point Feature Histogram (PFH) [Rusu, 2010], proposed by Rusu in 2009, is on the basis of robust surface normal estimation via PCA and fast point cloud localisation by tree structures [Rusu and Cousins, 2011]. As shown in Fig. 2.14(b), for each query point, its neighbours located in a local sphere space are retrieved; then for each unrepeated pair of the points within this sphere, the α, β, θ angular values shown in Fig 2.14(a) can be calculated from the estimated surface normals. The final descriptor can be formed by quantifying these angular values into three-dimensional histograms. Later on, PFH was accelerated by only connecting query points with neighbours, as opposed to connecting every pair in the sphere. This yields Fast Point Feature Histogram (FPFH) [Rusu et al., 2009b], shown in Fig. 2.14.

Encoding

Encoding provides higher-level representation of local descriptors extracted from the image by transforming the local descriptors into a vocabulary-space. One of the most classic coding methods is bag-of-features or bag-of-words (BoF) in which the local feature descriptors are treated as 'visual words' and unsupervised learning techniques are employed to train the 'codebook' (or dictionary) such as k-means, Gaussian Mixture Model (GMM) and mean-shift. Typically during the coding procedure, each 'word' (local descriptor) in the image is



(a) The α, β, θ angles.

(b) The Point Feature Histogram (PFH).

(c) The Fast Point Feature Histogram (FPFH).

Figure 2.14: The Point Feature Histogram (PFH) and Fast Feature Histogram (FPFH) features. The red point is the query point. In PFH, the points are fully connected in the sphere, while in FPFH, points are only connected with the query points, and features in neighbouring spheres are grouped together.

matched to its nearest codebook entry, and a global representation is obtained by matching each ‘word’ (local descriptor) in the image is matched to its nearest codebook entry. The accumulation of each codebook pattern is used as global representation (this process is also called sum-pooling [Lazebnik et al., 2006]). Vector Quantization (VQ) [Deng and Manjunath, 2001] follows the same process but uses filtered or vectorized image patch instead of local descriptors. Hence VQ is more popular in texture recognition.

Pooling

The task of integrating local representation into an image-level global representation (codes), pooling is required. Sum-pooling [Lazebnik et al., 2006] and max-pooling [Yang et al., 2009] are the two widely-used pooling methods. Sum-pooling accumulates the corresponding values (value of the code) among all the codes with respect to each codebook base, while max-pooling uses the largest value among all the codes with respect to each codebook base as the global representation.

Classification

As a final step to conclude the image classification pipeline is “classification” – identifying to which of a set of categories an unknown example belongs. In this section, the state-of-the-art discriminant classification approaches are briefly reviewed. The goal is to fit the classifier model for training data and the learned model should be able to predict unknown testing data.

In the early literature, k-Nearest-Neighbour (kNN) [Cover and Hart, 1967] and decision tree [Safavian and Landgrebe, 1991] are the most widely-cited discriminant algorithms. kNN is one of the simplest, non-parametric, lazy classifier, in which a simple majority voting strategy is adapted to predict the unknown examples. It is still highly desirable on a range of pattern recognition problems. Decision tree [Quinlan, 1986, Safavian and Landgrebe, 1991] is a hierarchical tree structure classifier, A tree can be ”learned” by splitting each tree node splits the examples into two sub-sets through thresholding the values in a specific dimension, and the dimension for splitting is chosen by calculating the information gain. Some other famous ensemble methods proposed are: bagging [Gregorski et al., 1994], Random Forest [Ho, 1998], AdaBoosting [Rätsch et al., 2001].

Support Vector Machine (SVM) is a widely-cited discriminative classifier, benefiting from max-margin, soft-boundary mechanism and kernel function. In the training procedure, the objective function is in maximizing the margin of examples from positive and negative classes, while the decision boundary can be ‘soft-boundary’ with tolerance for misclassified examples. SVM also supports various kernels and compared to linear kernels, non-linear kernels usually are able to enhance the classification performance but increase the complexity of computation.

SVM classifiers are popular due to their often excellent empirical performance and their use of kernel functions to map data into complex feature spaces in which linear classifiers can be built. However, there are drawbacks to SVMs. In particular, SVMs do not provide probabilistic confidences in their classifications. Although the outputs can be post-processed into probability values [Platt, 1999] this is known to be sub-optimal for small datasets where learning the parameters of the additional probabilistic model is biased by the high proportion of training points that have outputs ± 1 (the support vectors). SVMs are also inherently binary and solving multi-class problems involves combining the outputs of several binary SVMs. Although in some applications that can give high classification accuracy, it is not clear how one should combine the probability values computed for each classifier into a single distribution across the classes. For example, for bio-imaging based medical applications, the classification accruracy may not be high which involves techniques such as high-content screening, virtual screening [Samardzhieva and Khan, 2018]. Forest-like classifiers [Doumanoglou et al., 2014b, Willimon et al., 2013] can generate approximate probabilities via a voting scheme, but the reliability of such estimates is limited by the number of trees and has no formal probabilistic basis.

In light of these drawbacks of the commonly-reported classification methods, this thesis employs a multi-class Gaussian Process classification to fully model the distributions within the object category prediction problem. Gaussian Process (GP) [Rasmussen] is a non-parametric model for regression and classification problems. In GP regression problems, the conditional probability of latent variables w.r.t testing examples given training examples, testing examples, latent variables w.r.t training examples is modeled as a multiple-variant Gaussian distribution. For the classification problems, the key problem is to estimate the posterior of latent variables given training data. Several inference methods such as Laplace Approximation

[Rasmussen], Expectation Propagation [Rasmussen], are proposed to estimate this distribution. GP is also a kernel-based method since the covariance matrices are calculated by kernels that determine the estimation of distribution. In essence, the posterior distribution of GP latent variables over training examples are modeled as a multi-variant Gaussian distribution, and the Laplace approximation is employed to estimate it. Moreover, the hyper-parameters within the GP kernel are automatically optimized by marginal likelihood maximization. This type of classification allows demonstrating that the confidence provided through the conditional probabilities in a probabilistic classifier is a sensible halting criterion for interactive perception.

Although, deep learning techniques haven't been explored in producing this thesis, for completeness, a short review of such techniques are presented in the following section.

Deep Learning

Notable works such as [Jarrett et al., 2009, Krizhevsky et al., 2012, Simonyan and Zisserman, 2014] have used the most commonly used Convolutional Neural Network (CNN) Architecture. For image classification task, CNN was applied to the ImageNet challenge, which was able to classify images of more than 1000 categories trained from 1.2 million images [Krizhevsky et al., 2012], Ciresan et. al [Ciresan et al., 2012] proposed a multiple column deep CNN for hand-craft character recognition problems achieving substantial improvement. Recently, [Rana et al., 2016] proposed a dual-tree complex wavelet feature based CNN application to classify real and reconstructed stereo 3D video for consumer valuation.

For general object classification problem, [Donahue et al., 2013, Jia et al., 2014], pre-trained CNN is used to produce visual features.

Data-driven, Learning-based approaches such as [Kehoe et al., 2013, Viereck et al., 2017], have provided solutions for grasping objects covering a wide range of object configurations, shapes, and scenarios. For instance, [Levine et al., 0] devised a deep learning architecture, which is trained to predict the probability that task-space motion of the gripper that maximizes a predicted probability of quality grasp solutions. For this purpose, the network needs to observe the spatial relationship between the gripper and objects in the scene, and in turn, learning hand-eye coordination. They used a cluster of robots that collectively learned successful grasps to mitigate the need for a large dataset.

Another notable example is from [Mahler et al., 2017a,b], who have developed benchmark datasets (i.e. Dexnet 1.0, 2.0 and 3.0) where grasping candidates are learned using annotated datasets consisting of 150 3D models of which 1M rendered poses are extracted. Techniques like these are good motivation for incorporating better grasping strategies into the proposed visual architecture and in turn, can extend the effect of this architecture on a wider range of applications.

The key difficulty of training CNN is to set the network architecture. Zeiler et al. proposed a deconvolutional neural network [Zeiler et al., 2011] to adjust the neural network architecture structure and tune parameters of each layer, which is able to reconstruct the input in multiple layers by learning a sparse code of a specific image (or its feature maps) from learned filters common to all images.

The challenge and prerequisite of deep learning are that the training requires large amounts of data. Take CIFAR[16] and ImageNet[17] for example: the former has 80 million images for 100 categories and the latter has 14 million images for more than 30,000 categories. For the specific object recognition with manipulation tasks, where the training data is collected manually in the lab, the collection of large-scale datasets becomes extremely difficult.

2.6.3 Discussion

The proposed framework reported in this thesis for scene understanding by robotic interactive perception aims to provide a general solution. All the stages of a classic object recognition pipeline are employed in the proposed framework and each stage of the pipeline is interchangeable with a solution on a need basis. However, for classification stage, this thesis exploits the benefits of Gaussian Process (GP) based multi-class classifier. For the classification problem, the key problem is to estimate the posterior of latent variables given training data. Several inference methods such as Laplace Approximation [Rasmussen], Expectation Propagation [Rasmussen], are proposed to estimate this distribution. Using the associated probability along with the predicted class serves as a decision strategy for interaction with an object in the scene i.e whether the confidence of a class label is high enough to regard it as a true class or needs repositioning for further inspection if the confidence is lower than some pre-defined threshold. More details in chapter 5 and chapter 6.

2.7 The State-of-the-Art for scene understanding by interactive perception

[Gibson, 1979, Noe, 1979, O'Regan and No, 2001] make a compelling case that perception in humans and animals is an active and exploratory process. Another notable report in [Held and Hein, 1963], experimented to analyze the development of visually guided behavior in kittens. Their findings are notable that this development critically depends on the opportunity to learn the relationship between self-produced movement and concurrent visual feedback. As discussed in section 2.2.3, [Aloimonos et al., 1988] show how challenging vision problems,

[16] available at: $http://groups.csail.mit.edu/vision/TinyImages/$

[17] available at: $http://www.image-net.org$

such as shape from shading or structure from motion, are easier to solve with an active than a passive observer. However, in the remaining of this thesis, the visual sensor is not active i.e, the camera is fixed.

A variety of techniques for scene understanding aided by interaction, also known as interactive perception techniques have been gaining attention recently. Few of the notable works in the recent past are discussed here.

Segmentation through interaction has been proved to improve performances in [Fitzpatrick, 2003, Li, 2011, Schiebener et al.]. Another notable work in the field of interactive object sorting/recognition systems has been reported to achieve viable levels of performance [Gemignani et al., 2016, Gupta et al., 2015]. Their improved version in [Gupta et al., 2015], reports an investigation on sorting objects in clutter. In particular, small objects are sorted on a tabletop by segmenting the scene into regions: uncluttered, cluttered and pile. For the uncluttered case, every object is picked and placed in its respective bin to clear the table. In the cluttered case, objects are first separated and then picked and placed in bins. For the piled scene, the pile is knocked over by the manipulator in order to decompose it so that the objects lie directly on the table. However, this framework only works for small objects of single and homogeneous color. similarly, in [Tho et al., 2016], a solution to fruit sorting has been proposed. However, the number of fruit classes are limited to two and hence lacks generalization.

[Sinapov and Stoytchev, 2013, Sinapov et al., 2014a,b] investigated the interactive perception of object recognition by letting a robot interact with a set of objects. These objects are characterized by different attributes, such as rigid or deformable, heavy or light, and slippery or not. Features computed on the different sensor modalities serve as the basis to learn object similarity. The authors from their work showed that this task is eased when the learning process is conditioned on joint torques and the different interaction behaviors.

Motivated by the human infant development, a perceptual system is proposed by [Lyubova et al.] to allow a robot to learn about physical entities in its surrounding workspace in two stages. In the first stage, a human partner moves a workspace element and the robot learns the object's appearance of this moving element. In a second stage, the robot interacts with the objects to learn its appearance. Although this system requires only very limited prior knowledge, it has to have the knowledge of parts of its own body, parts of a human partner, and what constitutes a manipulable object. [Katz and Brock, 2008] reports an interactive perception approach that by focussing on object function rather than object appearance, is able to manipulate unknown objects which possess inherent degrees of freedom, such as scissors, pliers, but also door handles, drawers, etc., without requiring a priori model. This interactive perception system is able to manipulate articulated objects successfully by acquiring a model of the objects kinematic structure.

[Gupta et al., 2015] report an investigation on sorting objects in clutter. In particular, small objects are sorted on a tabletop by segmenting the scene into regions: uncluttered, cluttered and pile. For the uncluttered case, every object is picked and placed in its respective bin to clear the table. In the cluttered case, objects are first separated and then picked and placed in bins. For the piled scene, the pile is knocked over by the manipulator in order to decompose it so that the objects lie directly on the table. However, this framework only works for small objects of single and homogeneous color. Similarly, in [Tho et al., 2016], a solution to fruit sorting has been proposed. However, the number of fruit classes are limited to two and hence lacks generalization.

[Kaipa et al., 2016] presents an interactive perception-based approach that addresses perception uncertainty in order to reduce failure rates in a robotic bin-picking task. Human intervention is required when the uncertainty in the part detection leads to perception failure. The automated perception system provides the partial match to estimate the object's detection confidence. Thereafter, a sensor-less fine-positioning planner is used to correct the part placement errors.

In [Jang et al., 2017], Jang et al. considered the task of semantic robotic grasping, demonstrating the capability of a robot to pick up an object of a user-specified class using only monocular images. The proposed system is able to learn object detection, classification and grasping strategy. However, this work focuses on better grasping strategies and additionally, in order to achieve both successful grasping and object localization, a huge amount of training is involved. Also for novel objects, localization of the object is in question.

Overall, current techniques compromise over general object manipulation tasks such as general household objects under challenging real-world scenes i.e variable lighting, people walking in the background.

2.8 Discussion

In this section, a summary of limitations of the state-of-the-art for scene understanding by robotic interactive perception is described and how to advance it. In Section 2.8.1, the limitations of the state-of-the-art are depicted from four aspects: limitation in Active binocular vision systems, in Camera Calibration, and in Hand-Eye Calibration, in Visually Guided robotic systems for service tasks such as House Hold Object Sorting System and in Robotic Interactive Perception for Scene understanding. Then these limitations are summarised in Section 2.8.2, and Section 2.8.3 presents how this thesis advances these limitations.

2.8.1 limitations of the state-of-the-art for scene understanding by robotic interactive perception

Limitations in active binocular vision systems

As stated in section 2.2.3, an active vision system is reported in [Aragon-Camarasa and Siebert, 2009, Aragon-Camarasa et al., 2010], that is capable of binocular vergence, localisation, recognition [Aragon-Camarasa and Siebert, 2009, Aragon-Camarasa et al., 2010] and simultaneous identification of multiple target object instances [Aragon-Camarasa and Siebert, 2010]. This architecture is structured as a collection of ad-hoc functions in order to explore autonomously a scene by operating solely with SIFT features. Moreover, this system is constrained to hardware and, therefore, the limitation of its portability remained an issue. Recent developments in robotic middleware (e.g. the Robot Operating System [Quigley et al., 2009]) technologies have made possible the deployment of hardware independent robotic systems. Thus, there is a need to develop an active binocular robot head architecture that integrates visual behaviors in a parsimonious and generic robot vision architecture.

Limitations in camera calibration and Hand-Eye calibration

From investigation in section 2.3, rigid calibration methods consist of estimating intrinsic and extrinsic camera parameters, and the mechanical relationships between their actuation platforms and camera reference frames ([Kwon et al., 2007, Neubert and Ferrier, 2002, Salvi et al., 2002]). The approach proposed in [Furukawa and Ponce, 2009] suffers from mechanical wear and tear and the solution obtained depends on the quality of the mechanical parts at the moment the calibration was carried out. In addition, the distance and orientation between actuator joints have to be precisely measured. Hence, errors induced by the measuring device and mechanical backlash during their operation are not taken into account.

Furthermore, for PTU camera systems, self-calibration method capable of a general and generic solution has yet to be devised. Moreover, hybrid approaches lime [Mueller and Wuensche, 2016] fall short in terms of accuracy and precision performance since they attain an overall precision within 0.5cm and 1cm [Hansen et al., 2012].

Limitations in visually guided household object sorting system

From investigation carried in section 2.7, the approaches such as [Katz and Brock, 2008, Lyubova et al.], [Gupta et al., 2015, Kaipa et al., 2016] are limited in their ability to deal with general object manipulation tasks such as an autonomous sorting system involving interaction for general household objects. The perception stage devised for such cases need to provide additional information (e.g. probabilistic confidence measure) along with recognition information for further reasoning, essential for the interaction stage as to what can be done to the object in question, to improve on its recognition accuracy. Also for service robot to be widely accessible, it requires portable services programmed into them. Hence, portability and time sensitivity is essential but challenging.

Limitations in robotic interactive perception for scene understanding

Additionally, from section 2.7, it can be deduced that in robotic context, a scene can be studied and understood in more detail by visually guided object recognition systems when the perception stage is backed by an action stage. An action stage could involve an interaction of the active vision system with the scene or manipulate the object(s) in the scene e.g. reposition the object such that view of the object is not occluded and simple from clutter after interaction with it. Although, scene understanding for robotics has been explored by interactive perception with the use of active vision systems evident from the literature in 2.2.3. However, scene understanding by robotic interactive perception in terms of object manipulation investigations have been limited in their generality. In addition, a general framework for scene understanding by robotic interactive perception is needed where the observer and the object observed could be manipulated to simplify the scene and understand it thoroughly which would extend its use to numerous applications. More specifically, investigating to capture the right pose of an object in the scene can be advantageous both for recognition tasks and additionally, learning task for object appearances.

2.8.2 Summary

Overall, from the literature review, the conclusion can be stated as, that the current state-of-the-art in scene understanding by robotic interactive perception usually focuses on simple and/or controlled scene rather than the basis of understanding generic scene configurations. The key limitations can be summarized as:

- Existing methods including [Aragon-Camarasa and Siebert, 2009, Aragon-Camarasa et al., 2010] made contributions to eliminate the limitations discussed in section 2.2.3 and extended the scope of potential applications. However, for time sensitive and platform agnosticism robotic applications, the portability and platform agnosticism of the active vision system remains a challenge.

- A generic solution has yet to be devised as rigid and hybrid approaches such as [Mueller and Wuensche, 2016] fall short in terms of accuracy and precision performance since they attain an overall precision within 0.5cm and 1cm [Hansen et al., 2012].

- In predominantly reported scene understanding recognition pipelines by robotic interactive perception, the classifiers e.g. SVM, RF, kNN are non-probabilistic discriminative classifiers and cannot provide the confidence of prediction which discussed in 2.7, can play a key role in providing the information needed for reasoning whether or not an object needs to be investigated more.

- The majority of the reported scene understanding in robotic context are limited over general object manipulation tasks such as an autonomous sorting system involving interaction for general household objects. Some of these robotic systems do not have manipulation solutions followed by perception-manipulation cycles. As a consequence, the robot is unlikely to be able to recover the identity of the object from ill-posed configurations.

2.8.3 How to advance

Corresponding to these limitations, this thesis provides the following solutions to advance the state-of-the-art:

- This thesis validates the portability of an active binocular vision system with a functional accuracy of system reported in [Aragon-Camarasa and Siebert, 2009, Aragon-Camarasa et al., 2010] and a cross-platform, time-sensitive and resilient active vision system of environment changes for the task of scene exploration is developed.

- In this thesis, a camera and hand-eye calibration methodology for integrating an active binocular robot head within a dual-arm robot are described. In addition, a comparative study between current RGB-D cameras and our active stereo head within two dual-arm robotic testbeds is reported that demonstrates the accuracy and portability of our proposed methodology.

- A semi-autonomous visual perception model by interaction with the scene, for an application of objects sorting based on Gaussian Process (GP) classification is devised. The proposed model is capable of recognizing objects categories from point cloud data. Gaussian Process multi-class classification is adapted for predicting the category of unknown object in the scene and additionally provides with predictive probability. This thesis provides the first example of research to adapt non-parametric multi-class probabilistic classification (via Gaussian Processes) to the household object recognition problem by performing objects sorting task. The predictive probabilities generated by GP are adapted into a novel interactive sorting pipeline which leads to substantial improvements in the recognition performance.

- A complete autonomous visual perception architecture applied to the task of object sorting based on Gaussian Process (GP) classification is proposed, that is capable of recognizing objects categories from point cloud data. Interactive perception consists of two main stages: a perception stage and an interactive stage working in a loop, following multiple perception-action cycles. Where the output of the perception stage is fed into the interaction stage, and the feed may be received back from the interaction stage on a requirement, if the object needs further inspection to confirm the category of the object, to accomplish the task of accurate sorting of household objects placed on a table top. The perception stage is responsible for object detection/segmentation, object recognition, classification and the interaction stage verifies object class labels by interacting with the object, removing the object from the table top and dropping it in the bin designated for each class.

Chapter 3

A Portable Active Binocular Robot Vision Architecture for Scene Exploration

This chapter presents a portable active binocular robot vision architecture that puts together a number of visual behaviors. This active vision architecture has the abilities of vergence, localization, recognition and simultaneous identification of same-class object instances. The portability and functional accuracy of the vision architecture is demonstrated by carrying out a qualitative and comparative analysis under different hardware robotic settings, feature extraction techniques and viewpoints.

3.1 Introduction

Active robot vision systems are dynamic observers to perform actions and fulfill tasks by exploiting recovered information from the imaged scene [Ballard, 1991]. Active robot vision systems are mainly comprised of hard-wired, ad-hoc visual functions and the intended purpose is to have the capability of robustly exploring a scene and finding objects contained in a database of pre-trained object examples [Chen et al., 2011, Collet et al., 2011]. However, current systems are confined either by limitation in their visual capabilities and/or their software modules are crafted according to the robot's specific geometric configuration and hardware components. These limitations, in turn, constrain the scope of potential applications for such vision systems.

A portable active binocular robot head architecture is devised that is able to execute *vergence, localization, recognition and simultaneous identification of same-class object instances*. Here the focus is on the development of a portable architecture while preserving visual behaviors previously reported in [Aragon-Camarasa and Siebert, 2009, Aragon-Camarasa et al., 2010]. The Sensor Fusion Effects (SFX) architecture [Murphy and Mali, 1997] has been chosen as the foundation for the portable robot head (Fig. 3.1). It must be pointed out that *this robot architecture is not an attempt to model the mammalian visual pathway itself*, but it is a functional system that robustly carries out the specific high-level task of *autonomous scene exploration*. To demonstrate the portability and functional accuracy of the proposed system, experiments have been conducted considering three important variables for any active scene exploration tasks, namely; the hardware used, visual representation and view(s) of the scene. Hence, this chapter presents experiments with three different state-of-the-art feature extraction techniques, namely SIFT [Lowe, 2004b], SURF [Bay et al., 2008b] and KAZE [Alcantarilla et al., 2012] and, different hardware and scene settings.

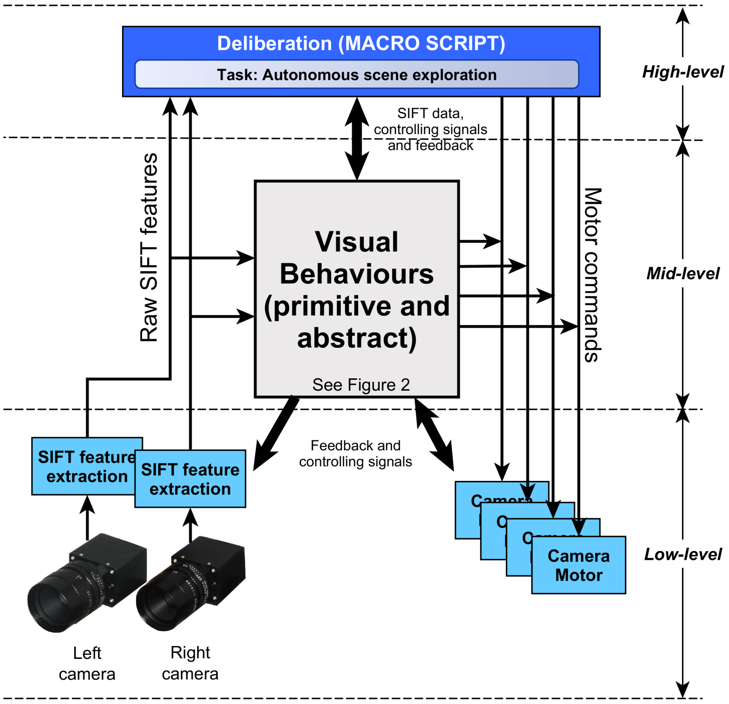

Deliberation (MACRO SCRIPT)
Task: Autonomous scene exploration
High-level
SIFT data, controlling signals and feedback
Raw SIFT features
Visual Behaviours (primitive and abstract)
See Figure 2
Motor commands
Mid-level
SIFT feature extraction
SIFT feature extraction
Feedback and controlling signals
Camera Motor
Low-level
Left camera
Right camera

Figure 3.1: Active binocular robot vision architecture [Aragon-Camarasa and Siebert, 2009, Aragon-Camarasa et al., 2010].

This chapter is organized as follows: Section 3.1 presents a brief overview of the active vision systems and the portable active vision system devised in the course of this thesis. Section 3.2 explains the motivation and objectives. Section 3.3 and 3.4 presents the robot vision architecture devised. Finally, details of the experimental validation of the system and concluding remarks of this chapter are given in Section 3.5 and Section 3.6 respectively.

3.2 Motivation and objectives

Visual behaviours previously reported in [Aragon-Camarasa and Siebert, 2009, Aragon-Camarasa et al., 2010] made contributions to eliminate the limitations discussed in 3.1 and extended the scope of potential applications. However, for time sensitive and platform agnostic robotic applications, the portability of the active vision system remained a challenge. Also, to increase the impact of an active vision system, there is a need to integrate the system from a platform dependent to a cross-platform vision system i.e integration of the existing system into Robot Operating System (ROS).

The objective of this chapter is to validate the functional accuracy of system reported in [Aragon-Camarasa and Siebert, 2009, Aragon-Camarasa et al., 2010] and develop an active vision system, that is cross-platform, time-sensitive and resilient to environmental changes, for the task of scene exploration.

3.3 Robot vision architecture

As stated before, the active vision system is based on the hybrid deliberative/reactive *Sensor Fusion Effector* architecture (SFX, Murphy and Mali [1997]). Specifically, the SFX architecture, as implemented, relates how deliberative and reactive modules are interconnected with sensor and actuator functions. Visual behaviors in this architecture implement the configuration of the visual streams in the mid-level of the SFX architecture. This arrangement exploits sensed visual information in order to explore the environment without further reasoning (i.e. the mid-layer *senses and acts* accordingly) while the deliberative layer manages visual behaviors and, consequently, orchestrates the required set of commands to carry out a *high-level* visual task; for instance, manipulation/interaction tasks Sun et al. [2015b].

Specifically, Fig. 3.1 shows this architecture. The processing levels are classified in terms of their functionality (i.e. low-level, mid-level and high-level). The corresponding low-level and mid-level functions consist of simple yet effective behaviors that sub-serve upper-level goals, whilst the high-level functions relate to the intelligence, deliberation, and reasoning (out of the scope in this thesis).

High-level functions (as observed in Fig. 3.1) specify visual tasks and goals. This layer is cast as scripted meta-behaviors (Section 4.5) that orchestrate the sequential activation of visual behaviors in order to fulfill the task of autonomous visual object exploration.

Low-level and mid-level (Figures 3.1 and 3.2) integrate a number of *primitive* and *abstract behaviours*. On the one hand, primitive behaviors comprise monolithic methods that only serve a single purpose; i.e. they are simple stimulus-response mappings that transform a collection of sensed information into data structures. On the other hand, abstract behaviors

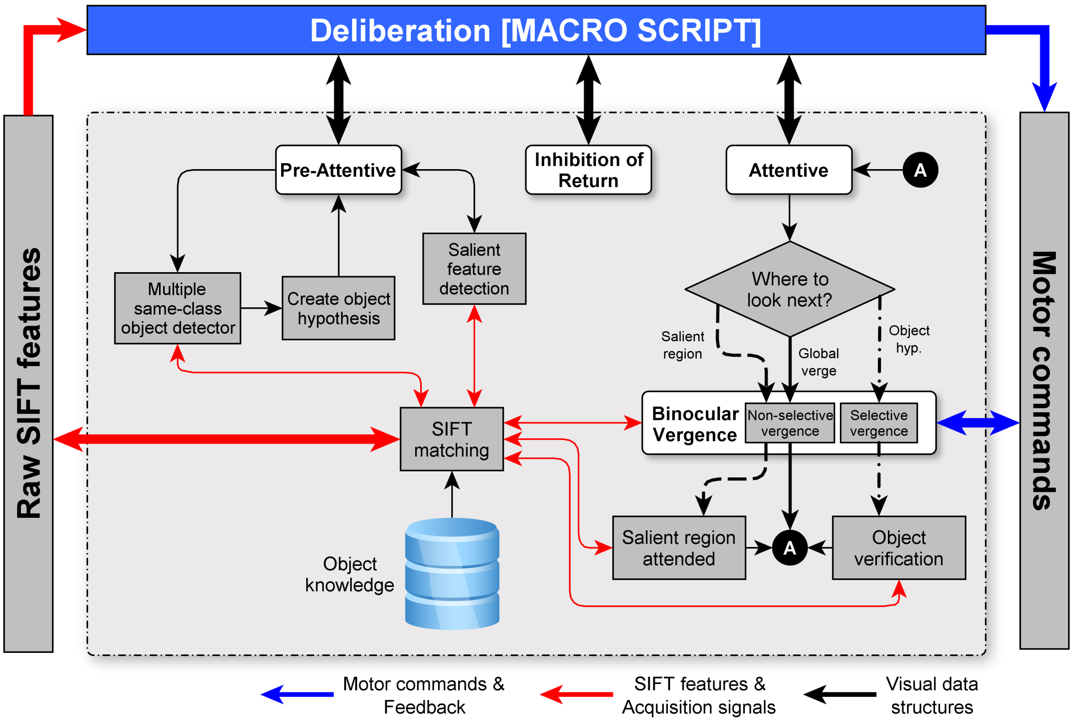


Figure 3.2: Internal representation of visual behaviours (Fig. 3.1). White boxes denote abstract behaviours, whereas grey boxes represent primitive behaviours [Aragon-Camarasa and Siebert, 2009, Aragon-Camarasa et al., 2010].

comprise a collection of primitive or other abstract behaviors. Fig. 3.2 illustrates the **mid-level** processing architecture that comprises *pre-attentive*, *attentive*, *inhibition of return* and *binocular vergence* visual behaviours previously reported in [Aragon-Camarasa and Siebert, 2009, Aragon-Camarasa et al., 2010] and for completeness, this thesis details these in the following sections. Sensor and motor behaviors are decoupled from the mid- and high-level layers. This configuration allows us to maintain visual behaviors that are not constrained to the chosen feature extraction technique and hardware components.

To achieve generic and preserve a modular arrangement within the architecture, an egocentric coordinate system is devised which are not related to the real-world units of the observed environment. The egocentric coordinate map is defined as a relative pixel-based map where the frame of reference is established with respect to a *“home”* position of the robot head.

Mid-level processing architecture that comprises *pre-attentive*, *attentive*, *inhibition of return* and *binocular vergence* visual behaviours are detailed in following sections, but first, binocular vergence is briefly described.

3.3.1 Binocular vergence

The vergence behavior is presented in Aragon-Camarasa et al. [2010], which has modeled different vergence modes based on the hierarchical paradigm and it is defined as *Global non-selective vergence* and *Attentive, selective vergence*. The implemented vergence behavior is a closed-loop function based on extracting and matching local features (in our case SIFT) from each of the stereo-pair images captured and result in calculating disparity values. These values are classified in a histogram and the highest peak of the distribution cues the most dense/compact feature cluster and this highest peak is then used to estimate the actuator movement required to rotate the gaze angle of each camera to centre this peak at zero disparity within a capture-verge loop and the loop halts when the highest peak has been shifted close to zero disparity. Further implementation details can be found in Aragon-Camarasa et al. [2010].

3.3.2 Pre-attentive

In the vision architecture, the pre-attentive behavior is divided into three different visual processing behaviors.

Multiple same class object instance detection

As scenes may contain both same-class known and unknown objects (i.e. objects may be self-occluded and occluded by other different-class objects), the implementation of a *"Multiple Same-Class Object Instance"* (MSCOI) detector affords the ability to localize multiple same class object instances pre-attentively in order to generate object hypotheses within the observed field of view.

Hypotheses generation

In order to provide a single set of new object hypotheses ($\mathcal{H}^{P_{new}}$) when multiple same-class object instances are in the scene, the pre-attentive behavior must be able to locate correspondences between the object instances found in $\mathcal{H}^{L}$ and $\mathcal{H}^{R}$. The working assumption is that both cameras are currently verged on, and they roughly observe, the same portion of the scene (within a disparity tolerance. This assumption is independent of the particular type of vergence behavior that is currently active. Hence, an object hypothesis is considered to be the same in terms of its spatial location in both observed images if it belongs to the same object class ($H^{\mathcal{I}}$, and the Euclidean distance between the camera fixation coordinates located on each camera image plane is close to zero. The egocentric pixel map stores the absolute coordinate position of the observed object instance fixation point with respect to the recorded home position of the robotic system. Thus, it is not necessary to create and store this map in working memory explicitly. This egocentric map further maintains a record of all fixations in the scene that is later employed in the *"inhibition of return"* behavior.

Saliency detection

As the cameras are only driven to look at objects or salient locations, salient locations are only registered if they appear in the field of view of the dominant eye (the left camera) when the cameras are either targeting an object or salient location. Salient locations are those SIFT feature locations in the dominant camera (left camera in our system) that do not match with any feature in the current view and exhibit a saliency score above a threshold value. The heuristic formulation of the saliency score is based on the degree of visual eccentricity at which observed salient items are located in the current view, in order to bias the system towards peripheral cues. The output of the "*salient feature detection*" behavior,

3.3.3 Attentive

According to the attentional "spotlight", the critical assumption adopted in this report for the visual search heuristic is that those putative object and salient locations that appear in the field of view of both cameras in the binocular configuration should be attended. An overview of the algorithmic implementation is depicted in Fig. 3.3.

The attentive behavior employs the information provided by the pre-attentive and inhibition of return behaviors to direct the attentional spotlight, to perform recognition decisions and to bring the cameras into convergence. As shown in Fig. 3.3, the attentive mechanism operates under the following three cases (each case is enumerated in Fig. 3.3 accordingly):

1. If $\mathcal{H}^P$ is not empty, the object hypothesis with the highest confidence score is then targeted, verged on (selective vergence) and verified.

2. If $\mathcal{H}^P$ is empty but $\mathcal{H}^S$ is not, the cameras are targeted and verged on the salient feature with the highest score (selective vergence).

3. If $\mathcal{H}^P$ and $\mathcal{H}^S$ are empty, the cameras are maintained in convergence under the non-selective vergence case without a pre-selected target.

The third point is a special case; this occurs: (a) when the scene has been fully explored, or (b) when the robot vision system is initializing. It must be noted that camera actuation is only invoked while targeting object hypotheses or salient locations that contribute to the overall

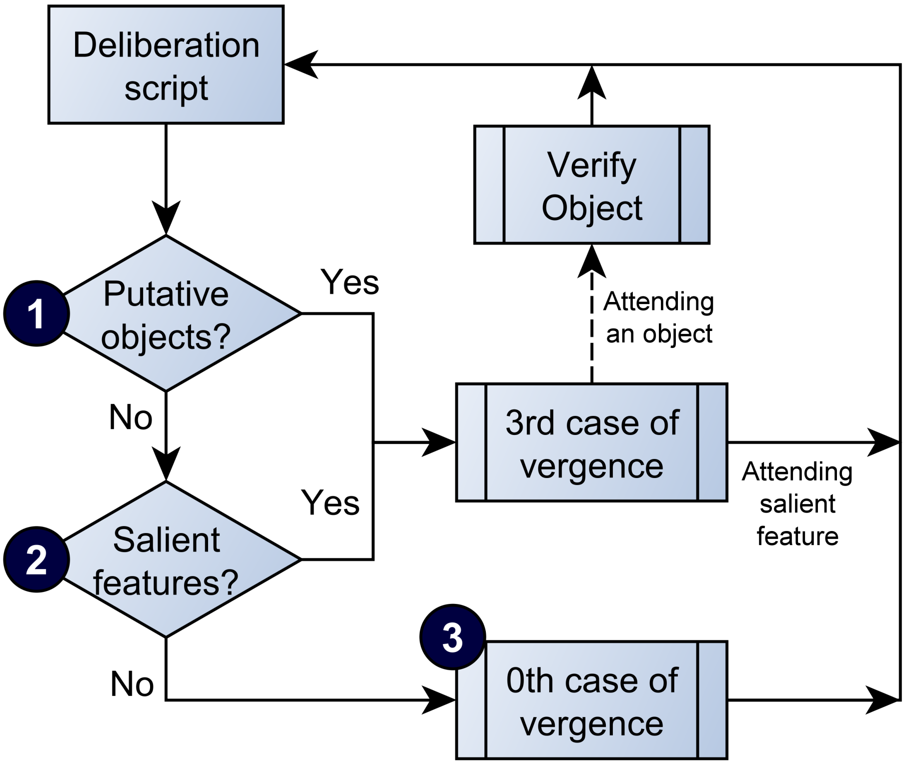


Figure 3.3: Flow diagram of the implemented attentive behaviour while executing a visual task. The three operational cases are marked accordingly [Aragon-Camarasa and Siebert, 2009, Aragon-Camarasa et al., 2010].

execution of the visual task, therefore motor control signals are explicitly commanded by the vergence behaviors in our architecture. As observed in Fig. 3.2(b), the *non-selective vergence* and *selective vergence* behaviour modes are integrated within the attentive behaviour.

Saccadic targeting

The saccadic targeting (denoted as *"Where to look next?"* in Fig. 3.2(b)) consists of two behavioural modes: *object based attention* ($\mathcal{H}^P$) and *salient based attention* ($\mathcal{H}^S$). As discussed in the previous section, the attentive behaviour prioritises $\mathcal{H}^P$ as the most important visual information to be observed.

Object verification

This behavior is solely concerned with corroborating the attended object hypothesis. As the system is capable of localizing multiple same-class object instances, the object attended could be close to or occluded by, another instance of the same class. The bounding box coordinates are thus employed (B^L and B^R to segment the region of interest *symbolically* in the current view and only those features inside both segmented regions are matched in order to verify the identity of the object. The system is thereby biased to consider only the object of interest since possible false positive matches of other same-class instances in the current view are excluded. Fig. 3.4 illustrates a segmented region of interest while the robot investigates a “juice box” object.

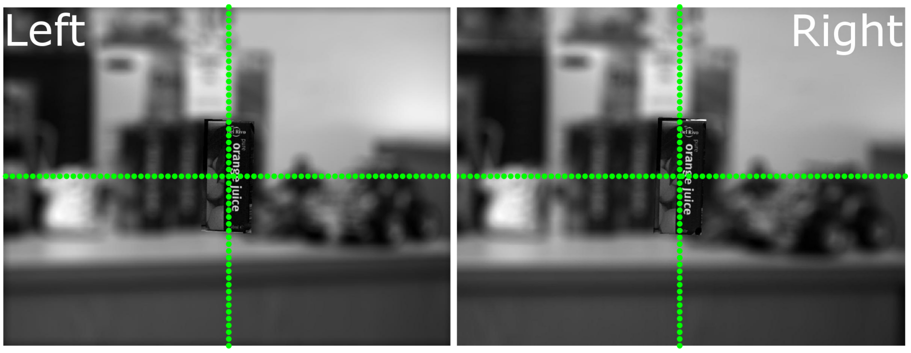


Figure 3.4: Demonstration of the selective vergence case and segmented region of interest while verifying an object [Aragon-Camarasa and Siebert, 2009, Aragon-Camarasa et al., 2010].

3.3.4 Inhibition of return

While exploring a scene, the binocular robot head might have detected other object hypotheses during previous pre-attentive cycles. The *inhibition of Return* (IOR) behavior must therefore determine whether detected object instances are: *identified objects that have been attended and verified*, $\mathcal{H}^A$; or, *previous pre-attentively localised objects that have not been attended yet*, $\mathcal{H}^{P_{new}}$.

Following each saccade, to inhibit multiple instances of each object class of the visual search task, the Inhibition of Return (IOR) behavior determines whether detected instances have been either: identified objects, attended and verified, $\mathcal{H}^A$; or pre-attentively localized objects (the set of putative objects of previous saccades, $\mathcal{H}^P$, that have not been attended yet).

A confidence ellipsoid test is used to define this behaviour, where the critical assumption hypothesis is that an object in $\mathcal{H}^P_{new}$ is inhibited if and only if its fixation coordinate, y^L (represented by points inside the drawn ellipse in 3.5), falls inside the region confidence interval of either an attended object ($\mathcal{H}^A$) or an object hypothesis ($\mathcal{H}^P$) in the dominant eye (left camera). Fig. 3.5 depicts an example of the inhibition process between $\mathcal{H}^P_{new}$ and $\mathcal{H}^A$. Although, the inhibition process between $\mathcal{H}^A_{new}$ and $\mathcal{H}^P$ is described (current and previous putative objects). However, the exact steps are also valid while inhibiting between $\mathcal{H}^P$ and $\mathcal{H}^A$. Thus, the algorithmic steps of this behavior are explicitly modeled in the spatiotemporal pixel map domain by means of the statistical confidence ellipse test as follows (this test is only applied when the object belongs to the same object class):

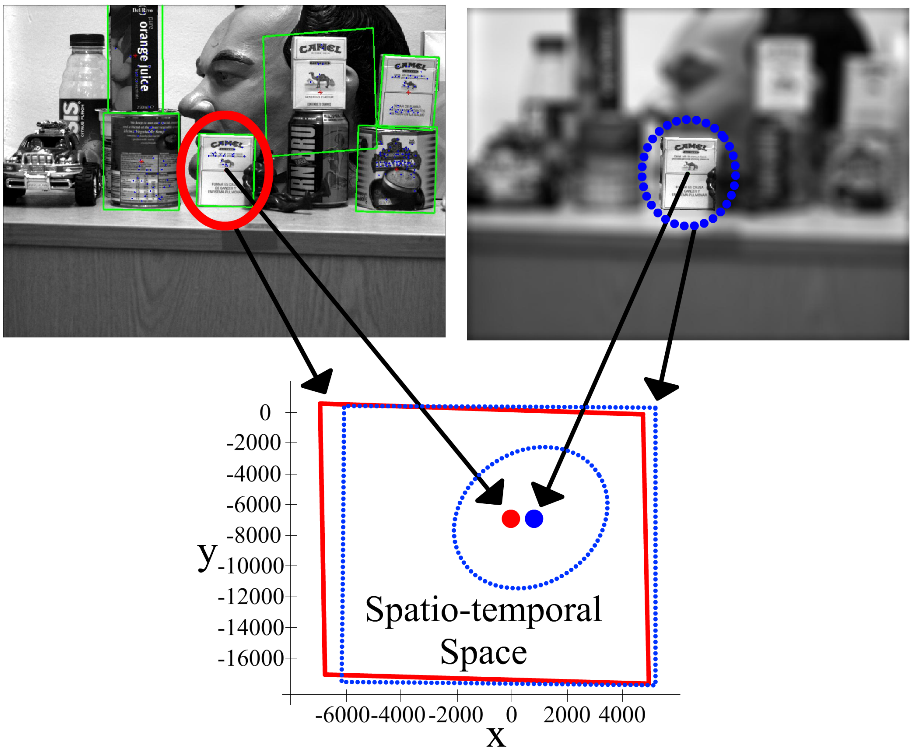


Figure 3.5: Example of the inhibition of return behaviour applied to $\mathcal{H}^{P_{new}}$ and $\mathcal{H}^P$. X and Y axes depict the egocentric map of the system described and these are expressed in pixel units with respect to the home position of the cameras [Aragon-Camarasa and Siebert, 2009, Aragon-Camarasa et al., 2010].

$$y = (\mathcal{Y}_i^{\mathcal{L}})^{\mathcal{P}}_{new} - (\mathcal{Y}_j^{\mathcal{L}})^{\mathcal{P}} \tag{3.1}$$

$$\mathcal{K}_{ij} = yC^{-1}y^T : \mathcal{I}_i = \mathcal{I}_j \forall \mathcal{I}_i^L \in \mathcal{H}^P_{new} \tag{3.2}$$

where $\mathcal{K}_{ij}$ is a confidence factor determined by the $\mathcal{X}^2$ distribution of the i^{th} and j^{th} object elements in $\mathcal{H}^P_{new}$ and $\mathcal{H}^P$; and, C, the covariance matrix between bounding boxes (BL) of i^{th} and j^{th} object elements in $\mathcal{H}^P_{new}$ and $\mathcal{H}^P$; p and q denote the population size of $\mathcal{H}^P_{new}$ and $\mathcal{H}^P$, respectively. The null hypothesis is defined as the probability that the ith object in $\mathcal{H}^P_{new}$ appears in the interior ellipsoid of j^{th} attended object (defined by the Equation 3.2) is equal to $P_{\mathcal{X}^2}(\mathcal{K}, d)$. Each result of the null hypothesis is thereby stored in an array, $G = [g_{ij}]^{pq}_{i=1,j=1}$ as follows:

$$G = \begin{cases} 1 & 1 - P_{\mathcal{X}^2}([\mathcal{K}_{ij}]^{pq}_{i=1,j=1}, d) > 0.1 \\ 0 & Otherwise \end{cases} \tag{3.3}$$

where $P_{\mathcal{X}^2}$ is the probability of the $\mathcal{X}^2$ distribution; 0.1, the 90% of the significance level of being the null hypothesis true; and d, the degrees of freedom which in this case is 2 as the visual coordinates are in the two-dimensional image plane. Thereafter, G is then reduced to a column vector in order to determine the inhibited object of the i^{th} element in $\mathcal{H}^P_{new}$, hence;

$$\mathbf{g}'_i = [g_{i1}]^p_{i=1} \vee \ldots \vee [g_{iq}]^p_{i=1} \tag{3.4}$$

In case, a pre-attentively observed object in the current saccade might be detected with a higher confidence value than the one from previous pre-attentive cycles; therefore, the new object hypothesis presenting the highest confidence replaces the previous hypothesis. This special case is only valid while inhibiting objects in $\mathcal{H}^P_{new}$ and $\mathcal{H}^P$. This is a useful utility, by allowing the system to correct detections and to suppress visual object information that might not contribute to the overall visual search task.

The final result of inhibited sets thus consist of appending to $\mathcal{H}^P_{new}$ and $\mathcal{H}^S_{new}$ in $\mathcal{H}^P_{new}$ and $\mathcal{H}^S$, respectively. Therefore, $\mathcal{H}^P$ and $\mathcal{H}^S$ are the output of this behavior.

3.4 Visual search task definition

The high-level layer is defined as a macro-script that specifies the visual search task, controls, and schedules behavioral resources in lower layers (ref. Aragon-Camarasa et al. [2010]), and monitors the progress of the task. It is worth noted here, that, a *pre-attentive-inhibition of return-attentive cycle* is defined, in order to allow the active vision system to perform autonomous scene exploration (Table 4.2). That is, the robot acts according to the sensed visual information and reports recognized object classes stored in a database.

3.5 Experimental validation

In this section, experimental validation on portability and functional accuracy of the portable active binocular vision system is discussed in detail.

3.5.1 Robot head hardware and software interface

These experiments are designed to validate the portability of the active robot vision architecture in two different scene settings, hardware components, and different visual features extraction techniques. The first active binocular robot head (Fig. 3.6) comprise two *Prosilica* cameras (*GC2450C* and *GC2450*; colour and mono, respectively) at 5 Megapixels of resolution fitted with *Gigabit Ethernet* interfaces and 4 high-accuracy stepper-motors and

Table 3.1: Pseudo-code of macro script in Fig. 3.1 and 3.2.

```
Inputs: None
Outputs: List of objects recognised and attended to.

Generate database
Verge cameras and extract features from the image pair
 (binocular arrangement)
Obtain pre-attentive object and salient hypotheses
Set the saccade number to 1
Loop until possible object or salient hypotheses are not empty
or no. of saccades is less than a user-defined number
   Select an object from the possible obj. hypotheses that has
   the maximum recognition score (see Aragon-Camarasa et al. [2010])
   Verge and attend (attentive behaviour) to the selected object and return features
   from both cameras after verging and the lists of the
   remaining object and salient hypotheses
   Update pre-attentive object and salient hypotheses
   Inhibit (inhibition of return) new pre-attentively found
   objects w.r.t previous possible object and salient hyps
   Saccade no. increments 1
Report objects stored
```

motor-controllers (Physik Instrumente). The robot vision architecture is arranged as follows for the latter robot head. Low-level components, namely, image acquisition and motor control modules (Fig. 3.1); are interfaced to a Pentium 4 computer with 2 GB in RAM running under Windows XP and MATLAB R2008a. Whilst, image feature extraction, mid-level and high-level components (Fig. 3.1) are interfaced to a 4-core Intel Xeon (model E5502) with a CPU clock speed of 2 GHz, with 24 GB in RAM running under Windows 7 and MATLAB R2009b. Both computers are interconnected through the local network by means of a collection of network socket functions for MATLAB[1].

The second active binocular robot head (Fig.) consists of two Nikon DSLR cameras (D5100) at 16 Megapixels of resolution. Cameras are mounted on two pan and tilt units (PTU-D46) with their corresponding controllers. This robot head is mounted on a dual-arm robot with anthropomorphic features. Low-level functions where implemented as ROS nodes and interfaced with Matlab 2014a with *pymatlab*[2]. The hardware is interfaced to an Intel Core i7-3930K computer at 3.20 GHz with 32GB of RAM running Ubuntu 12.04 and ROS.

[1] `http://code.google.com/p/msocket/` (verified on 4 March 2016)

[2] `https://pypi.python.org/pypi/pymatlab` (verified on 4 March, 2016)

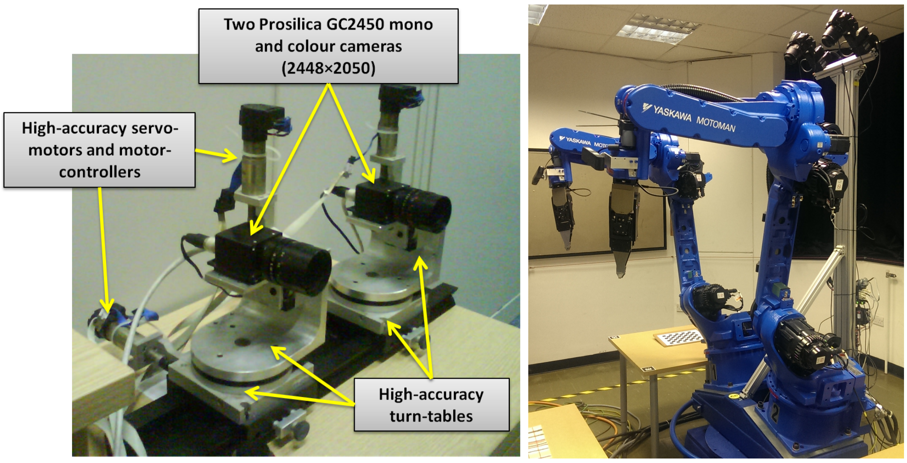

Two Prosilica GC2450 mono and colour cameras (2448×2050)
High-accuracy servo-motors and motor-controllers
High-accuracy turn-tables
YASKAWA MOTOMAN

Figure 3.6: Left: The Prosiclica robot head exploring the scene. Right: An image of the dual-arm robot featuring the Nikon robot head on top. Additionally, this robot features grippers specifically designed for manipulating clothing.

3.5.2 Methodology

In order to test the robustness and repeatability of this active vision architecture, for both binocular robot heads, 3 visual exploration tasks have been performed for each scene, each visual task with a random initial home position. It must be noted that the visual search task is terminated if the robot's pre-attentive behavior does not find an object within 5 consecutive saccades; i.e. the system is only targeting salient features. This halting criterion has been implemented in order to reduce the execution time while conducting these experiments. There are three possible outcomes while actively exploring a scene:

- *True positives* are all correctly detected and identified object hypotheses where the system is able to center the hypothesized object in the field of view.
- *False positives* are when the system localizes an object hypothesis, but without being able to center the object in the field of view of both cameras during the attentive cycle or, similarly, an attended object hypothesis does not correspond to the object class in the scene.
- *Not found* is the case that translates to the system's failures. i.e an object instance is not detected in the visual search task at all.

For each robot head, scenes have been designed comprising of a mix of several multiple same-class and different-class object instances, arranged in different poses. A scene complexity is defined according to the number of similar unknown objects in the scene (i.e. a typical source of potential outliers) and by the degree of background clutter present (i.e surfaces, the objects are placed on, and at the background e.g wall). Training of the background clutter is not required for our active vision architecture. This makes the tasks object detection and recognition of objects, challenging in general. Experimental methodology is detailed below.

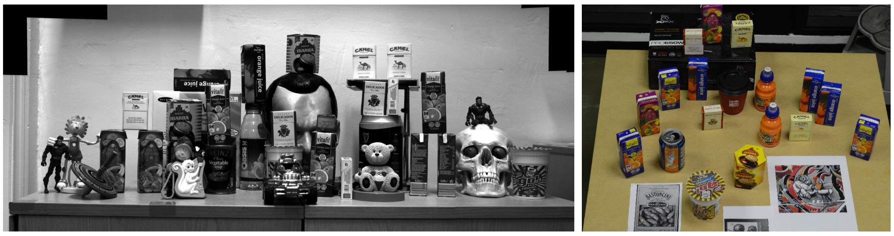


Figure 3.7: Left: View from the Prosilica robot head's left camera exploring a scene. Right: View of the Nikon-based robot head as viewed from the left camera [Khan et al., 2016c].

Prosilica robot head.

7 different scenes have been arranged [3] of differing complexity, based on combinations of 20 known object instances, of 10 different object classes. Fig. 3.7 shows an example of a scene. Objects were placed in arbitrary poses and locations. A database has been created of the 10 known objects by capturing stereo-pair images of an object at angular intervals of 45° and 60°. These captured images are then manually segmented in order to contain only the object of interest. This segmentation of objects for database creation is made autonomous and reported in the vision pipeline developed for robotic interactive perception (details in Chapter 5). We have considered two databases in order to measure the recognition performance of our system with different visual knowledge.

Nikon binocular robot head.

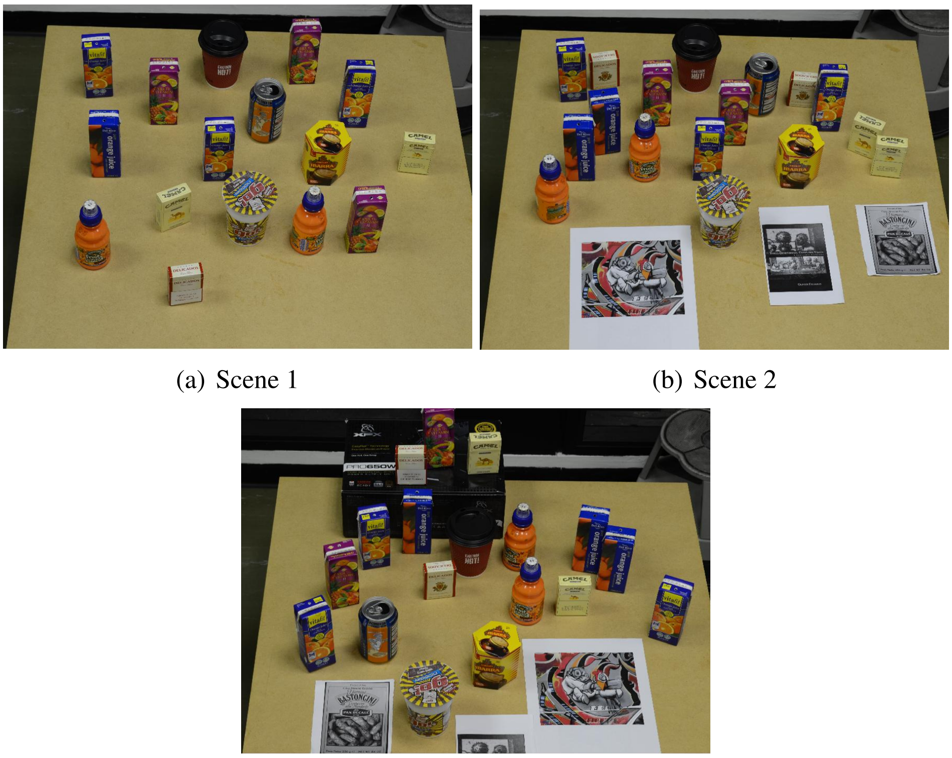
Scenes for these experiments consist of objects placed on top of a table. The goal is to investigate the response of our active vision architecture to different viewpoints, different feature extraction techniques and hardware components for the sake of portability. With this robot head, we are also able to investigate the effects of having an anthropomorphic robot configuration as opposed to a front-parallel configuration as above. Fig. 3.8 shows examples of the scenes we created. Object databases used in these experiments include stereo-pair images of object instances sampled randomly in order to cover the objects' view-sphere by placing the object in isolation on the working table. Each object instance stored in the database is manually segmented.

Therefore, 3 different scenes[4] are arranged of variable complexity. Each scene is a composition of 14 known object instances observing arbitrary poses and locations, of 9 different

[3] All 7 scenes can be accessed at `http://www.gerardoaragon.com/taros2016.html`

[4] All 3 scenes can be accessed at `http://www.gerardoaragon.com/taros2016.html`

object classes. Scene 1 is considered to be the simplest while scene 3, the most complex (Fig. 3.8). It must be noted that Scene 2 and Scene 3 include flat objects and objects with 3D structure while Scene 1 only comprises objects having a 3D structure. In order to effectively understand the response of the system to different feature extraction techniques, each of the three scenes were explored by our system with SIFT, KAZE, and SURF features.



(a) Scene 1

(b) Scene 2

(c) Scene 3

Figure 3.8: Scenes used for the Nikon robot head. a) Scene 1 depicts less complexity. b) Scene 2, medium complexity. c) Scene 3, most complex scene of the last two [Khan et al., 2016c].

3.5.3 Analysis and discussion

Investigating all experiments and three randomly starting position for each scene, it can be deduced that the portable active robot vision architecture presents stochastic behaviors. Accordingly, neither robot vision head follows a pre-defined visual scan path but it adapts according to the contents of the scene while exploring the scene. Table 3.2 and Table 3.3 show the true positives, false positives, objects not found and the active vision system to recover from failure for different scenes, for Prosilica robot head and Nikon robot head respectively. Summary of the outcomes for each robot head is presented as follows.

Prosilica robot head.

Table 3.2 illustrates the system's recognition rates for all experiments. False positives emerged due to the object feature descriptors matching with unknown objects and, in consequence, these matches were not consistent with the reference object center in the database while generating object hypotheses pre-attentively (as previously reported in [Aragon-Camarasa and Siebert, 2009]). However, the system recovered from false positives. These results further support the active vision paradigm, since the robot vision architecture is able to recover from these failures while investigating the scene from different views. Thus, the robot is able to locate the object instances, despite not identifying every object instance during the current pre-attentive cycle.

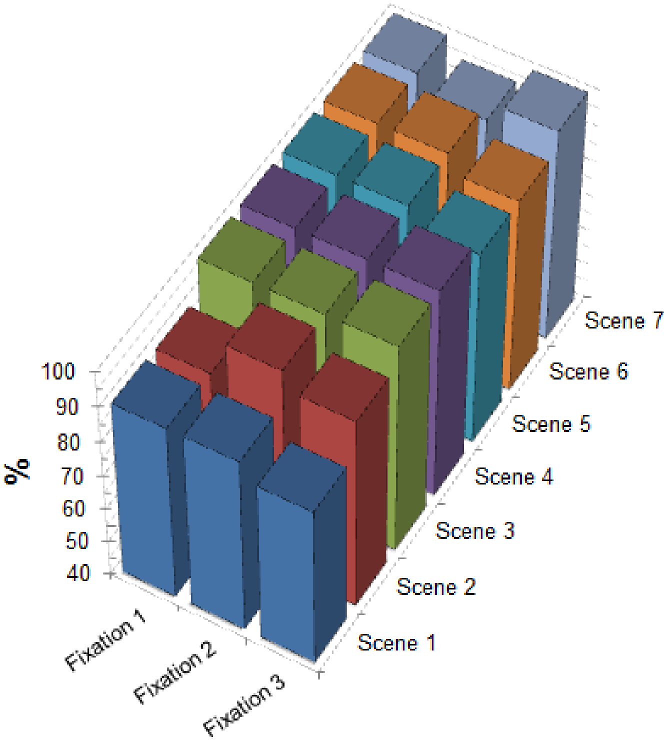


Figure 3.9: Overall recognition rate for the visual tasks for the Prosilica robot head.

Table 3.2: Outcomes for the Prosilica robot head.

Scene no.	True positives	False positives	Not found	Recover from failures	Performance (%)
1	56	5	7	0	82
2	57	1	3	0	93
3	60	2	0	2	97
4	60	2	0	2	97
5	59	5	1	4	91
6	59	0	1	1	98
7	59	1	1	0	97
Total	**410**	**16**	**13**	**9**	**93.5 (average)**

Table 3.3: Outcomes for the Nikon robot head.

Type of Feature Descriptor	Scene no.	True positives	False positives	Not found	Recover from failures	Performance (%)
SURF	1	16	5	14	54	53
	2	26	4	13	65	66.6
	3	25	2	17	64	59.5
Total		**67**	**11**	**44**	**184**	**60 (average)**
SIFT	1	24	0	6	30	91
	2	29	0	10	54	98
	3	32	0	10	59	97
Total		**85**	**0**	**26**	**143**	**77 (average)**
KAZE	1	30	0	0	30	100
	2	30	0	9	39	76.9
	3	30	0	12	29	71.4
Total		**90**	**0**	**21**	**98**	**83 (average)**

Nikon robot head.

From table 3.3, it can be observed that the recognition performance is linked to the feature extraction techniques used. Average recognition rates for SURF, SIFT, and KAZE are 60%, 77%, and 83% percentage, respectively. SIFT and KAZE, in these experiments, achieved better recognition rates than SURF due to the inherent properties of being ”almost” invariant to perspective transformations. It is also worth noting that both SIFT and KAZE techniques are less prone to false positives as opposed to SURF. As described above, this portable active vision architecture is tested using an anthropomorphic configuration where objects are not in similar 2D planes as it is the case from the Proscilica robot head experiments. By comparing Table 3.3 with Table 3.2, a decrease in the performance is observed. That is, 3D structures from an anthropomorphic configuration are more difficult to recognize and, therefore, the robustness of feature descriptions decrease. There has been an observation of more *recoveries from failures* in these set of experiments. It can be deduced that this particular configuration introduces challenging geometric transformations that state-of-the-art feature descriptions are still not able to cope with. Hence, the chosen feature extraction has a key role in the overall recognition performance. Nevertheless, our active robot head is able to explore a scene regardless of hardware configuration, different viewpoint while maintaining acceptable recognition rates.

3.6 Conclusions

This chapter presented a successful demonstration of a portable active binocular robot head that integrates visual behaviors in a unified and parsimonious architecture that is capable of autonomous scene exploration. That is, the portable active robot vision architecture can identify and localize multiple same-class and different-class object instances while maintaining vergence and directing the system’s gaze towards scene regions and objects.

This portable robot vision architecture has been validated over challenging scenes and realistic scenarios in order to investigate and study the performance of the visual behaviors as an integrated architecture. By carrying out a qualitative comparison with current robot vision systems whose performance has been reported in the literature, it is argued that our architecture clearly advances the reported state-of-the-art [Aragon-Camarasa et al., 2010, Arbib et al., 2008, Ma et al., 2011, Meger et al., 2010, Rasolzadeh et al., 2010] in terms of our system's innate visual capabilities and portability to different environment settings, e.g. multiple same-class object identification and tolerated degree of visual scene complexity. Our architecture is therefore portable in order to be adapted to different hardware configuration, feature description, and viewpoints.

For any vision architecture to benefit from interaction with the scene i.e object manipulations, requires the vision system to be integrated within the autonomous system's kinematic frame chain. This required an approach for camera-hand eye calibration. The Chapter 4 details the proposed methodology for camera-hand eye calibration required for the vision system to be integrated into a dual-arm robot's kinematic frame chain.

Chapter 4

On the Calibration of Active Binocular and RGBD Vision Systems for Dual-Arm Robots

In this chapter, a camera and hand-eye calibration methodology for integrating an active binocular robot head within a dual-arm robot is described. For this purpose, the forward kinematic model of the portable active robot head is derived and methodology for calibrating and integrating the robot head is described in detail. The rigid calibration provides a closed-form hand-to-eye solution. Furthermore, this chapter presents an approach for updating dynamically camera external parameters for optimal 3D reconstruction that is the foundation for robotic tasks such as grasping and manipulating rigid and deformable objects. It is shown from experimental results that our robot head achieves an overall sub-millimeter accuracy of less than 0.3 millimetres while recovering the 3D structure of a scene. In addition, a comparative study between current RGBD cameras and an active stereo head within two dual-arm robotic testbeds is reported that demonstrates the accuracy and portability of the proposed methodology.

4.1 Introduction

Camera calibration or more precisely geometric camera calibration is the process of estimating the parameters of a lens and image sensor of an image/video camera. Once estimated, these parameters are used to correct for lens distortion, making it viable to measure the size of an object in world units in the scene. Therefore, the importance of camera calibration is realized in robotics, for navigation systems, 3-D scene reconstruction etc.

RGBD camera sensors (i.e. Kinect-like cameras) have had an impact on robotics and robot vision research as they have provided a low-cost, ready-to-use and off-the-shelf sensor to accommodate different robotic configurations, settings, and tasks. However, the accuracy of RGBD camera sensors varies according to the distance between the object to be imaged and the sensor [Khoshelham and Elberink, 2012]. That is, RGBD sensors limit the perceptual capabilities of robots since they provide low-resolution depth maps and usually suffer from image noise. Likewise, their rigid configuration does not allow the robot to adjust the cameras' physical configuration in order to image objects at different distances from the camera with high accuracy. These limitations make characterizing the 3D structure of a given object challenging.

Photogrammetric vision systems provide the required accuracy but at a high cost. To mitigate these costs without compromising accuracy, an active binocular robot head is designed with off-the-shelf components for an industrial dual-arm Yakasawa robot (Fig. 4.1) under the FP7 CloPeMa project[1]. This robot head is capable of changing its gaze under computer control. Our robot head has been used successfully for clothes perception and manipulation research [Sun et al., 2015a, 2016a,b] because of its ability to provide high-resolution imaging for 3D mapping and range sensing. Due to the ability of the robot head to target different parts of a scene, it is required to maintain accurate calibration of its intrinsic and extrinsic parameters with respect to the robot's reference frame.

This chapter presents the methodology used to calibrate the active robot head, and the hand-eye calibration by integrating our robot head, and RGBD cameras within a robot's kinematic frame.

This chapter is organized as follows: Section 4.2 provides the motivation and objectives. Section 4.3.3 details the materials related to the investigation. Section 4.4 describes the methodology opted for camera calibration and hand-eye calibration. Section 4.5 details the experimental results of the system and concluding remarks and future extension of this method are given in Section 4.6.

4.2 Motivation and objectives

One of the limitations of the off-the-shelf RGBD sensors is that these are rigidly configured which, in turn, does not allow the robot to adjust the cameras' physical configuration, in order to image objects at different distances from the camera with high accuracy. These limitations make characterizing the 3D structure of a given object challenging. This leads to developing a vision system that could image objects located at different distances with high accuracy such as the vision system discussed in previous chapter 3.

[1] `http://www.clopema.eu/`

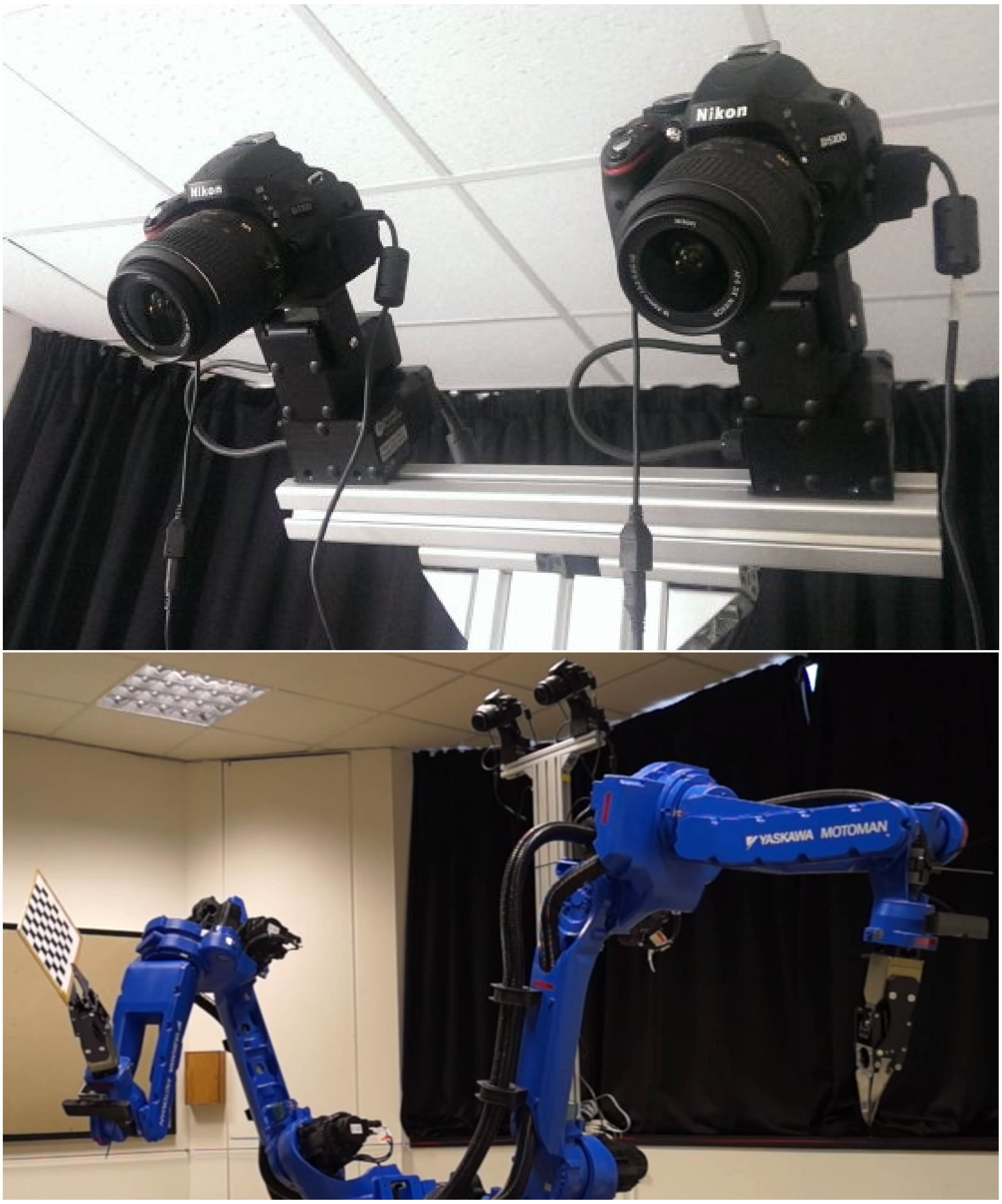


Figure 4.1: Top: The active binocular robot head. Bottom: Dual-arm Yakasawa robot with our robot head integrated.

However, with the ability to target different parts of a scene, the active robot vision system is required to maintain accurate calibration of its intrinsic and extrinsic parameters with respect to the robot's reference frame.

This chapter describes the methodology to calibrate the active robot head and describes the solution developed to dynamically update cameras' extrinsic parameters, in order to achieve geometric compatibility with respect to the robot's kinematic chain. Furthermore, this chapter provides a comparative evaluation of 3D reconstruction of the scene, between our robot head and RGBD cameras by integrating them within a robot's kinematic frame to demonstrate the operational validity of the proposed method.

4.3 Materials

This section starts by introducing the hardware facilities used to demonstrate the validity of the proposed approach. It is followed by the methodology opted for Camera-Hand eye calibration.

4.3.1 RGB-D cameras

This section briefly states specifications of the two of the most famous RGB-D cameras.

ASUS Xtion Pro

The ASUS Xtion Pro is an RGB-D camera that uses depth-sensing technology. It is based on structured light and has a depth range from 0.8 to 3.5 m, a 3D point cloud resolution of 640×480, an RGB image resolution of 1280×1024, a frame rate of 30 fps and a latency of 1.5 frames. The depth error of this sensor decreases according to the increasing distance between the sensor and the object. After calibrating the sensor, it achieves a depth accuracy at best of 4.7 mm at a distance of 0.96 m and drops to 38.6mm at a distance of 3.6m [Karan, 2015].

Kinect V2

Kinect V2 is also an RGB-D camera from Microsoft based on time of flight technology. It has a depth range of 0.5 to 4.5 m, 3D resolution of 512×424, RGB resolution of 1920×1080, a frame rate of 30 fps and latency of 20 ms. Similar to the ASUS Xtion Pro, the depth error of the Kinect V2 sensor decreases according to the distance between the sensor and the object. It is evident that the distance measurements delivered by this sensor are much more precise than the ASUS Xtion Pro. However, the Kinect V2 reconstruction error is approximately 20mm at a distance of 3m [Pagliari and Pinto, 2015].

4.3.2 Active binocular robot head

For the developed active binocular robot head, we employed relatively inexpensive and commercially available components in order to allow us to capture high-quality 3D depth maps and dense point clouds for deformable object recognition [Sun et al., 2016a] and manipulation [Sun et al., 2015a]. Hence, our robot head comprises two off-the-shelf Nikon DSLR cameras (D5100) that capture 16 MegaPixels images (MP) every 400ms. Each camera is mounted on two degrees of freedom pan and tilt platforms (PTU-D46). Cameras are rigidly separated by a pre-defined baseline for optimal stereo capturing. We interface our active robot head to an Intel Core i7-3930K computer with 32GB of RAM running Ubuntu and ROS.

Fig. 4.1 depicts the robot head as mounted on our dual-arm robot. The active robot head's visual capabilities include: autonomous gaze control and visual search based on SIFT features [Khan et al., 2016c]. To achieve real-time performance, we have implemented a GPU variant of SIFT [Wu, 2007]. GPU-SIFT features are used for verging the cameras and for tracking features in the scene for our dynamic calibration routine (Section 4.4.4). Likewise, a GPU version of the stereo matcher [Cockshott et al., 2012] is used, to compute horizontal and vertical disparities of two images captured by our robot head. Thus, our robot head can produce 16MP depth maps and point clouds in 0.2 fps.

4.3.3 Robots

In this paper, an evaluation is provided for the developed calibration methods over two different robotic testbeds. The first testbed is a dual-arm industrial robot manufactured by YASKAWA Motoman, as shown in Fig. 4.1. This robotic testbed comprises two MA1400 manipulators and a custom-made turntable. Each arm has 6 DOF and features 4 kg maximal load weight, 1.4 meters of maximal reaching distance, and ±0.08 mm accuracy. The dual-arm robot is powered and controlled by the DX100 controller. Our active binocular robot head is rigidly mounted on the turntable and in between both robot arms.

The second testbed is the Rethink Baxter robot (Fig. 4.2). Baxter is a humanoid, anthropomorphic robot with two seven degree-of-freedom arms and state-of-the-art sensing technologies. Its key purpose is to be able to operate continuously within humans environments and run for longer periods of time. Baxter's positional accuracy is $\pm 5mm$ with a maximal reaching distance of 1.2 meters.

4.4 Methodology

To obtain metrically accurate depth maps under dynamic camera motion, in this chapter, a hybrid approach is adopted. That is, first rigid camera calibration routines are employed to obtain intrinsic and extrinsic camera parameters and then the rigid Euclidean transformations between the robot and calibration target is found, and for the robot head and robot (Section

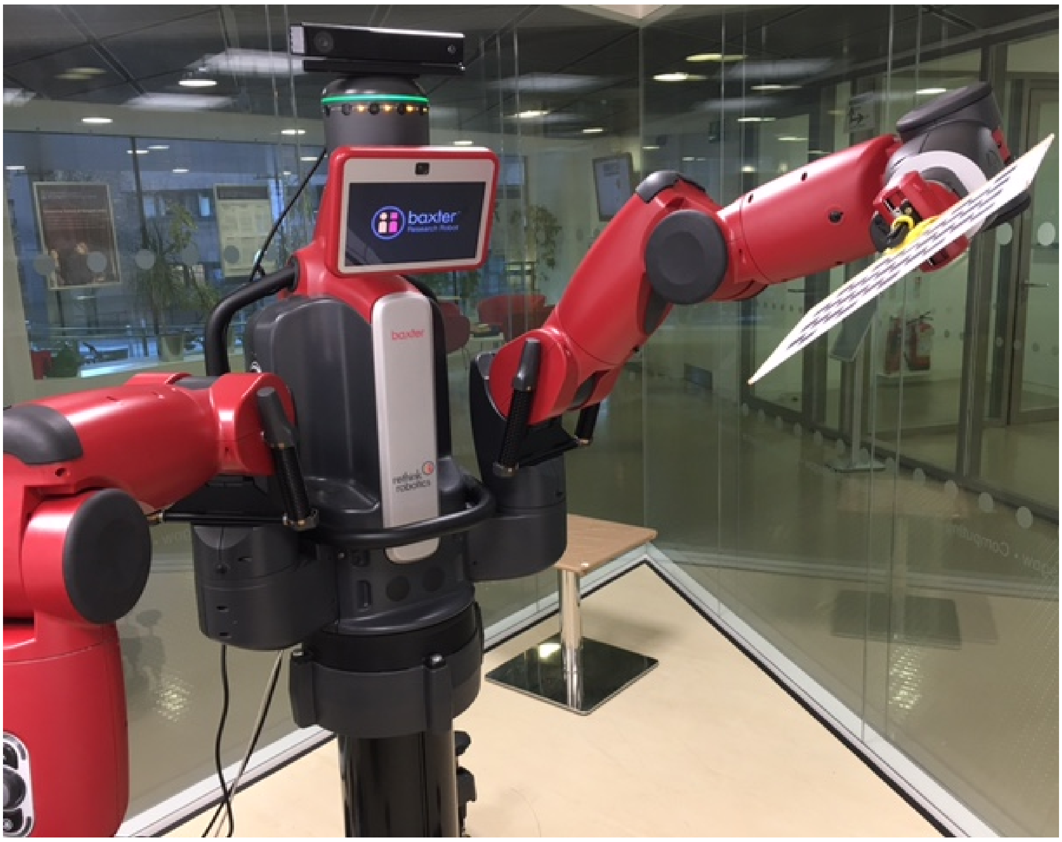


Figure 4.2: Rethink Baxter robot, holding the calibration target. In our experiments, the RGBD camera (Kinect V2 in Fig.) was mounted at the top of the robot's head.

4.4.2 and 4.4.3, respectively). Dynamic, online, calibration, in turn, consists of tracking known 3D positions from stable interest point features observed from previous camera poses (Section 4.4.4). Tracked features are then used to update camera extrinsic parameters accordingly.

4.4.1 Forward Kinematics Derivation

In order to find the geometric transformations to integrate the active binocular robot head within the robot, firstly there is a need to deduce the forward kinematic model of the active binocular robot head. Fig. 4.3 depicts the coordinates frames in the "home" position of the robot head (H subscript in Fig.). We use Sharkey et. al. [Sharkey et al., 1997] notation to express our forward kinematic model. Table 4.1 defines the coordinate frames. It is assumed that the world reference frame, $\{W\}$, lies on the base, $\{B\}$, of the robot head, and $y_{CS_0} = z_{CS_0} = \theta_C = 0$. Sharkey et. al. [Sharkey et al., 1997] established that if the cameras are targeting the same point X in $\{W\}$, then both open kinematic chains for each camera will be closed around X. Therefore, the forward kinematics for the open kinematic chain

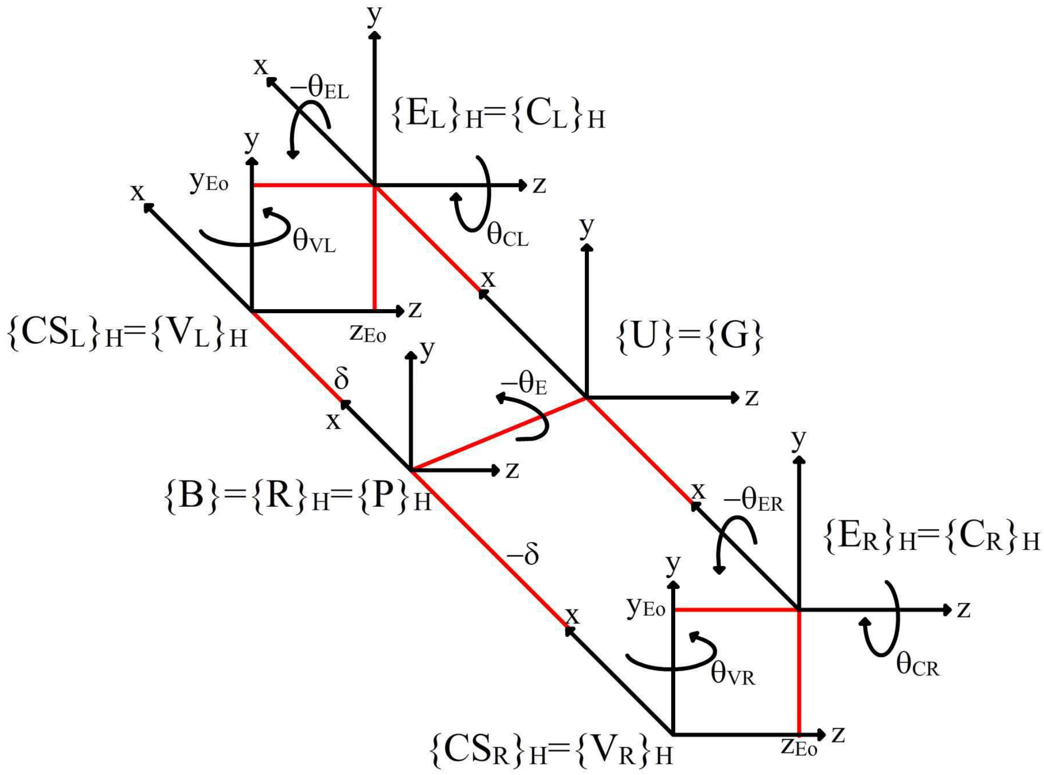


Figure 4.3: Coordinate frames in the HOME position of the CLoPeMa robot head [Khan et al., 2016a].

Table 4.1: Coordinate frames definition.

{World}	World
{Base}	Base
{Roll}	Roll (Not considered)
{Pan}	Pan (Not considered)
{CS$_{R/L}$}	Camera separation (left/right) Offsets: $(0, y_{cs0}, z_{cs0})$; and from {B}, it is given by + / - δ
{E$_{R/L}$}	Independently elevated (left/right)
$\{\theta\}$	Angle of rotation ($\theta_R; \theta_P; \theta_{EL}; \theta_{ER}; \theta_{VL}; \theta_{VR}; \theta_{CL};$ and θ_{CR}
{V$_{L/R}$}	Vergence (left/right) Offsets w.r.t {E} : $(0, y_{v0}, z_{v0})$
{C$_{L/R}$}	Cyclotorsion (left/right) (Not considered)
{O$_{L/R}$}	Optical (left/right) Offsets w.r.t {C} or {V} (for systems without {C} : (α, β, γ) and (u, v, w)
{T}	Target object. For simplicity, both camera represent a single point in {W} Offsets w.r.t $z_{OL/OR}$: d_{TL} and d_{TR}

from the base to the target position is found as:

$$
{}^{B}X = H_T^W \times \left[\begin{array}{cccc} 0 & 0 & 0 & 1 \end{array} \right]^T = \\
\left[\begin{array}{c} s\delta + d_T Cos(\theta_E) Sin(\theta_V) + z_{E_o} Sin(\theta_V) \\ d_T Sin(\theta_E) + y_{E_0} \\ d_T Cos(\theta_E) Cos(\theta_V) + z_{E_o} Cos(\theta_V) \\ 1 \end{array} \right] \tag{4.1}
$$

where ${}^{B}X$ is the point w.r.t. the base of the robot head, H_T^W is the transformation between world and target reference frames, s indicates if the camera separation is for the left, $s = 1$, or right, $s = -1$, camera. Thus, Equation 4.1 defines both forward kinematic chain for the left and right camera.

In practice, the above forward kinematic chain is not closed as it is assumed that the tilt reference frame is aligned to the optical image plane. In our robot head, it is required to find the geometric transformation from the tilt reference frame to the principal point of the camera's image plane, ($\{E\}$ and $\{C\}$ as defined in Table 4.1, respectively). To find H_C^E, first, it is needed, to carry out camera calibration to find the intrinsic geometry of the camera and hence the principal point of the camera's image plane as described on what follows.

4.4.2 Camera calibration

For camera calibration, OpenCV camera calibration method determines the distortion matrix and afterward, the required camera matrix. The distortion matrix is composed of radial and tangential distortion coefficients.

$$x_{corrected} = x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \tag{4.2}$$

$$y_{corrected} = y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \tag{4.3}$$

Where $x_{corrected}$ and $y_{corrected}$ are the corrected pixel values of the corresponding x and y pixel values in the original image. When images are taken with lenses not perfectly parallel to the imaging plane, tangential distortion occurs which can be corrected as follows:

$$x_{corrected} = x + [2p_1 xy + p_2(r^2 + 2x^2)] \tag{4.4}$$

$$y_{corrected} = y + [p_1(r^2 + 2y^2) + 2p_2 xy] \tag{4.5}$$

Solution to these equations (4.2, 4.3, 4.4 and 4.5) result in distortion matrix expressed as:

$$Distortion_{coefficients} = \begin{pmatrix} k_1 & k_2 & p_1 & p_2 & k_3 \end{pmatrix} \tag{4.6}$$

Finally camera matrix is given as:

$$\begin{bmatrix} x \\ y \\ w \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

Here, using homography coordinate system, w = Z. To find these intrinsic camera parameters, the cameras' parameter spaces are sampled using different positions and orientations of the OpenCV's check-board target. The calibration target is attached to the robot gripper and used the dual-arm robot to automate the target's sampling process. The robot head is converged to a fixed point in the robot's space and the robot head remains fixed at this point during camera calibration. Thus calibration for each camera in isolation is performed by presenting the calibration target at different positions and orientations in order to sample the camera parameter space adequately.

For each sampled target position, OpenCV returns the cameras' pose w.r.t. the target. However, it is found that by optimizing these poses using sparse bundle adjustment[2], it is possible to obtain a more accurate estimation of the camera's pose in comparison to the one obtained through OpenCV's stereo calibration routines. Hence, the stereo geometric relationship between cameras can be found by triangulating kinematic transformation across the calibration target as:

$$H_{O_R}^{O_L} = H_T^{O_L} \left(H_T^{O_R}\right)^{-1} \tag{4.7}$$

4.4.3 Hand-Eye calibration

Typically for object manipulation tasks in robotics, an object recognition system determines the position and orientation of the object with respect to the sensor. This location (position and orientation) of the object is mapped from the sensor frame to the gripper of the robot, to perform the manipulation task in hand. This mapping of position and orientation from sensor frame to the gripper of the robot is coined as hand-eye calibration. For this purpose, the classical method for solving the homogeneous transformation equation is employed.

$$AX = XB \tag{4.8}$$

[2] https://sourceforge.net/projects/cvsba/

Where the robot gripper's motion A, and the corresponding camera motion is B, the two motions are conjugated by the hand-eye transformation X. Decomposing the above equation into two equations: A matrix equation depending on rotation and a vector equation depending both on rotation and translation:

$$R_A R_X = R_X R_B \tag{4.9}$$

and

$$(R_A - I)t_X = R_X t_B - t_A \tag{4.10}$$

where I is 3x3 identity matrix. Taking advantage of the particular algebraic and geometric properties of rotation (orthogonal) matrices. Eq. 4.9 can be written as:

$$R_A = R_X R_B R_X^T \tag{4.11}$$

which is a similarity transformation since R_X is an orthogonal matrix. Hence, matrices R_A and R_B have the same eigenvalues. As we know property of a rotation matrix is that, it has one of its eigenvalues equal to 1. Let n_B be the eigenvector of R_B associated with this eigenvalue. By multiplying eq. 4.9 with n_B we obtain:

$$R_A R_X n_B = R_X R_B n_B = R_X n_B \tag{4.12}$$

and it can be concluded that $R_X n_B$ is equal to n_A , the eigenvector of R_A associated with the unit eigenvalue:

$$n_A = R_X n_B \tag{4.13}$$

To conclude, solving for $AX = XB$ is equivalent to solving for eq. 4.13 and for eq. 4.10. According to [Tsai and Lenz, 1989], at least three positions are necessary in order to uniquely determine X , i.e., R_X and t_X .

Specifically, to close the kinematic chain of the robot head and, consequently, the robot integrating the robot head, it is required to find:

- The transformation between the robot gripper and the calibration target – $H^T_{gripper}$; and,
- The transformation from tilt (E) and the camera frame (O) for the left and right camera – H^E_O.

To find the above transformations, Tsai's hand-eye calibration is implemented [Tsai and Lenz, 1989] as a ROS node. Thus, for $H^T_{gripper}$, the hand-eye calibration routine is fed with the left camera poses and the forward kinematic chain of the robot for each sampled target position as described in Section 4.4.2. Note that the kinematic chain of the robot is defined using ROS Unified Robot Description Format (URDF) (`http://wiki.ros.org/urdf`) and transformations are retrieved using TF (`http://wiki.ros.org/tf`).

By obtaining the pose of the camera with respect to the robot base for both the left and right cameras, it is now possible to be able to estimate H_O^E. The strategy adopted consists of sampling random PTU pan and tilt positions while capturing images of the calibration target that is fixed. For each camera movement, the forward kinematic chain is computed for each camera using the kinematic model described in Section 4.4.1. Finally, an implementation of ROS node is used for each camera in order to obtain the geometric transformation that relates $\{E\}$ and $\{C\}$.

It must be noted that each actuated pan and tilt unit (PTU) is treated separately during hand-eye calibration. It is assumed that the world reference frame lies on the base of each PTU of the robot head. If the cameras are targeting the same point in the world, then both open kinematic chains for each camera will be closed around this world point.

Tsai's hand-eye calibration algorithm provides us with an estimation of the kinematic transformation from the base of the robot to the gripper holding the calibration target and from the gripper to the calibration target. It is then possible to find the forward kinematic transformation from the robot's world reference frame to the left camera in the robot head's "HOME" position by triangulating transformations. For the right camera, the transformation of the stereo configuration is known; therefore, it can be concluded that the kinematic chains for each camera will be closed around the robot base. The complete camera-hand eye calibration routine is depicted in Fig. 4.4.

4.4.4 Dynamic tracking of extrinsic camera parameters

The estimated forward kinematic chain in the previous section holds true only when the cameras do not change their gazing point and remain in the "HOME" position. This is because the mechanical information of the PTUs is inaccurate and, in consequence, Tsai's hand-eye calibration routine finds an optimal solution for the sampled poses. Thus, in order to maintain dynamic calibration, the Euclidean transformation is computed every time the cameras move by using known visual information from previous camera positions. Hence, it is necessary to update the transformation from the world reference frame of the robot to the left camera and the left camera to the right camera. The latter is achieved by tracking previously observed and stable 3D points in the scene.

Tracking consists of stereo triangulating SIFT features coordinates from each camera. 3D projection of SIFT coordinates are expressed in terms of the world reference frame of the robot – these features are then used to initialize the dynamic updating routine. A PnP algorithm [Ferraz et al., 2014] is used on matched SIFT features between previous and current

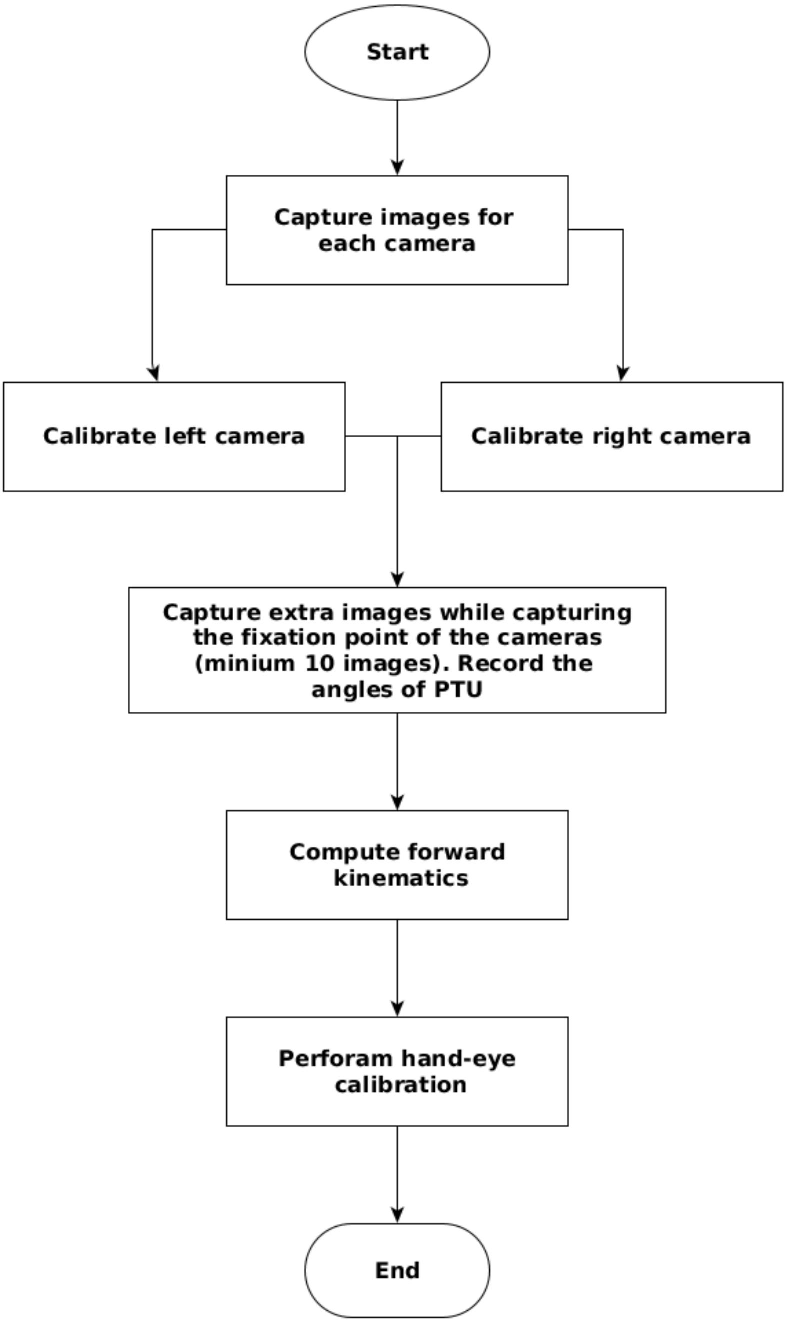

Start
Capture images for each camera
Calibrate left camera
Calibrate right camera
Capture extra images while capturing the fixation point of the cameras (minium 10 images). Record the angles of PTU
Compute forward kinematics
Perforam hand-eye calibration
End

Figure 4.4: Flow diagram of the camera-hand eye calibration pipeline [Khan et al., 2016a].

observations in order to recover the Euclidean transformation from 3D coordinates to the left camera. As outliers are likely to affect the performance of the PnP algorithm, SIFT feature matches are filtered by adopting a RANSAC homography fitting strategy from the current image to the previous image. Finally, the stereo relationship between the cameras is computed by computing the missing link from the forward kinematic chains of each camera and the camera projection matrices are updated accordingly for optimal 3D reconstruction.

4.5 Experiments

In this section, details of experiments performed are presented.

4.5.1 Active binocular robot head

The ability of the system to maintain dynamic calibration is expressed in terms of the accuracy of 3D reconstruction. In this case, the cameras can be positioned in any angle range in the tilt and pan axes (within the hardware range limits of the PTU); therefore, the extrinsic properties of the stereo configuration need to be updated as explained in section 4.4.4. Fig. 4.5 shows a simplified example of the coordinate frames defined for describing the kinematic structure of the robot head after hand-eye calibration. The objective of this experiment is to test the accuracy of our binocular robot head, to recover and reconstruct an accurate point cloud regardless of camera motion. For this purpose, an object model with known 3D structure is employed. The object model used in these experiments is the check-board calibration target. This planar calibration target allows us to:

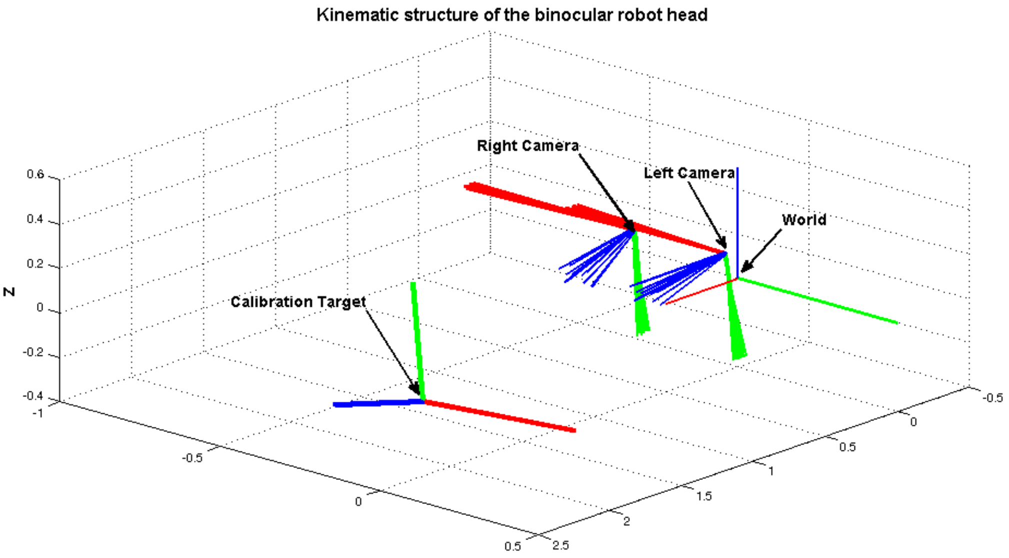


Figure 4.5: Simplified coordinate frames of the stereo head. This consists of the world, camera and calibration target coordinate systems.

1. compare the reconstruction residuals between a calibrated stereo system and the updated extrinsic camera parameters; and,
2. verify the accuracy of the reconstructed geometry by measuring the RMSE between a known 3D (ground truth) measurement on the real object and the 3D reconstructed model.

To measure the accuracy of the system, the calibration target is sampled over 10 different poses, by setting the robot head, and robot in a random position. The 3D Euclidean distance from one point of the calibration target to next in line is measured (Table 4.2). The ground truth distance between squares on the checkboard is 24mm. The RMSE between reconstructed points of the check-board while using the dynamic calibration parameters is therefore compared with respect to the ground truth. As observed in Table 4.2, the reconstruction accuracy for X, Y and Z is less than 0.3 mm. Therefore it can be stated that our binocular robot head in combination with the proposed rigid and dynamic calibration is capable of reconstructing the scene within sub-millimeter accuracy. Likewise, Table 4.3 shows the mean and standard deviation of RMSE (in millimetres) values in Table 4.2. From the results, it can, therefore, be concluded that the system is able to recover the 3D geometry with sub-millimeter accuracy – our dual-arm robot has an accuracy of $\pm 0.08mm$. Hence, the estimated geometric transformations between robot head's and robot's frame are within optimal limits for the required accuracy for practical manipulation tasks.

Table 4.2: Residual errors (in millimetres) between optimal stereo calibration and dynamic calibration of the extrinsic camera parameters. 3D coordinates are expressed with respect to the world reference frame of the robot head.

Pose #	**X**	**Y**	**Z**
1	0.17	-1.03E-02	-3.02E-02
2	1.50E-02	6.37E-02	-4.68E-02
3	2.04E-02	5.14E-02	3.76E-02
4	0.67	-6.69E-02	-0.201
5	0.79	-8.60E-02	-0.111
6	7.82E-02	1.85E-02	-3.47E-02
7	0.42	-3.67E-02	-0.119
8	6.95E-02	1.16E-02	-6.94E-03
9	0.38	-3.22E-02	-6.90E-02
10	0.32	-2.60E-02	1.50E-02
Mean	0.29	-0.011	-0.056
1 Std	0.27	0.025	0.06

4.5.2 Xtion Pro and Kinect V2

From the above assertion, system's ability to maintain dynamic calibration is expressed in terms of the accuracy of the 3D reconstruction. For consistency and completeness of our hand-eye calibration approach, the experiments discussed above for two RGB-D sensors i.e ASUS Xtion Pro and Kinect V2, mounted on Baxter's head are repeated. The aim is to test the accuracy of these RGB-D sensors while being integrated within the kinematic's chain of the robot.

Table 4.3: RMSE (in millimeters) between estimated 3D reconstructed points and ground truth for the active binocular robot head.

Pose #	**Mean**	**1 Std**
1	0.10	0.077
2	0.09	0.034
3	0.12	0.067
4	0.31	0.072
5	0.05	0.060
6	0.12	0.067
7	0.22	0.077
8	0.22	0.077
9	0.05	0.057
10	0.34	0.069
Overall	***0.16***	***0.105***

It must be noted that similar rigid calibration steps are followed as for our robot head (Fig. 4.4). Hence, both RGBD sensors are calibrated by employing calibration methods for the Xtion PRO available at `http://wiki.ros.org/camera_calibration`; and, for Kinect V2 from `https://github.com/code-iai/iai_kinect2/tree/master/kinect2_calibration`. Then the geometric transformation from robot's gripper and calibration target is obtained using our hand-eye calibration routine. By having knowledge of this transformation, then the RGBD camera's position w.r.t. the robot's kinematic chain is computed. The accuracy of the RGBD cameras is shown in table 4.4 for ASUS Xtion Pro and table 4.5 for Kinect V2.

From the errors reported in the table 4.4 and table 4.5, it can be observed that the system is able to recover the 3D geometry with an overall RMSE of 12.4mm with ASUS Xtion Pro sensor and 11.6mm with Kinect V2 sensor. As observed, the larger accuracy errors are due to the intrinsic properties of the RGBD sensors (ref. Section 4.3.1) and the positional accuracy of Baxter's robot (ref. Section 4.3.3). Hence, it can be concluded that these RGBD cameras are able to recover the 3D geometry of objects and scenes within millimeters accuracy. Note that the latter statement depends on the positional accuracy of the robot test-bed and depth accuracies of RGBD sensors.

4.6 Conclusion

This chapter described a calibration methodology to calibrate an active binocular robot head system integrated within an industrial dual-arm Yakasawa robot and described a comparative experiment on the integration of RGB cameras into the Rethink Baxter robot. This chapter also presented a simple, yet effective solution to dynamically update cameras' ex-

Table 4.4: RMSE (in millimetres) between estimated 3D reconstructed points and ground truth for ASUS Xtion Pro in Baxter.

Pose #	Mean	1 Std
1	13.63	9.35
2	14.04	5.18
3	10.55	5.24
4	11.85	6.15
5	14.63	6.66
6	14.88	5.85
7	15.54	8.56
8	8.83	4.40
9	9.38	3.63
10	11.59	6.40
Overall	***12.49***	***6.14***

Table 4.5: RMSE (in millimeters) between estimated 3D reconstructed points and ground truth for Kinect V2 in Baxter.

Pose #	Mean	1 Std
1	13.68	7.401
2	13.42	4.24
3	12.84	4.70
4	12.88	5.58
5	8.09	3.45
6	11.43	5.09
7	9.08	3.97
8	8.44	2.16
9	9.85	3.30
10	16.55	8.38
Overall	***11.62***	***4.83***

trinsic parameters in order to achieve geometric compatibility with respect to the robot's kinematic chain. By comparing the binocular robot head with consumer RGBD cameras, it can be concluded that this robot head provides an off-the-shelf depth sensing solution capable of reconstructing the observed 3D scene within sub-millimeter accuracy. Likewise, it is shown that the implemented calibration routines in this chapter can provide reliable results that allow reconstruction with relative 3D errors of less than 0.3 millimeters. An open-source ROS package that implements our calibration methods can be found at `https://github.com/gerac83/glasgow_calibration`. A video demonstration of the calibration of our active binocular robot head can be accessed at `https://youtu.be/9OYy9Q_bN2w`.

Having solved the problem of camera-hand eye calibration by the proposed methodology in this chapter, it is desirable for the visual architecture to extend its capability of interaction with the scene, by object manipulation. An initial attempt towards achieving this goal is the subject of Chapter 5.

Chapter 5

Interactive Perception based on Gaussian Process Classification for House-Hold objects Recognition and Sorting

This chapter presents an interactive perception model for the task of category-wise objects sorting, based on Gaussian Process (GP) classification that is capable of recognizing objects categories from point cloud data. In this approach, Fast Point Feature Histogram (FPFH) features are extracted from point clouds to describe the local 3D shape of objects, and a Bag-of-Words coding method is used to obtain an object-level vocabulary representation. Multi-class Gaussian Process classification is employed to provide a probable estimation of the identity of the object and serves a key role in the interactive perception cycle - modeling the perception confidence. This serves as reasoning information useful for the interaction stage, that is responsible to invoke action skills on a need basis to confirm the identity of the object with high confidence in multiple perception-action cycles. For this purpose, first, results are obtained by simulation of input data on both SVM and GP based multi-class classifiers to validate the recognition accuracy of the proposed perception model and suitability of GP based classification. Results obtained during this investigation demonstrate that by using a GP-based classifier, true positive classification rates are achieved of up to 80% for the hand-crafted dataset of general household objects. Second, experiments are designed and results are obtained by demonstrating the task of semi-autonomous objects sorting system to show that the proposed GP based interactive sorting approach outperforms random sorting by up to 30% when applied to scenes comprising configurations of household objects.

5.1 Introduction

It is essential for service robots to have the ability to recognize objects in their immediate vicinity when working in dynamically evolving human environments. Ideally, these robots should be capable of detecting, recognizing and classifying objects within their environment, and then interacting with these objects without the need for supervision. In this chapter, an interactive perception model is presented to prove its effectiveness by demonstrating with an application of sorting everyday household objects into their respective categories through direct visual observation (typically when objects are not occluded), and then by means of active object manipulation where objects are occluded or ”difficult to recognise”. The proposed visual architecture also termed as GP based Interactive Perception Model (GP-IPM) does not require prior knowledge about the environment or scene. This visually assisted objects sorting system is capable of segmenting a set of household objects lying directly on the robot's workspace tabletop, and categorizing these objects into their respective object classes (e.g. juices bottles, mugs, etc.). The system has been pre-trained on a subset of these object instances, while a proportion of the objects investigated have not been used to pre-train the system. We revisit the high-level model of the proposed visual architecture from Chapter 1.

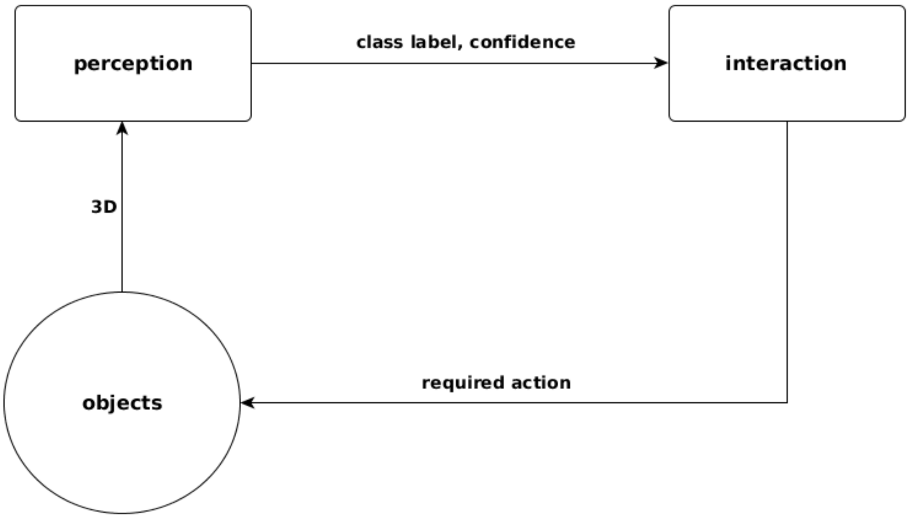


Figure 5.1: Gaussian Process Classification based Interactive Perception Model (GP-IPM).

The proposed visual architecture is portable, invariant to 6 Degree Of Freedom (DOF) pose changes and operates close to real-time. The pipeline consists of the following stages: object segmentation, visual representation, classification, semantic visualization, and finally a human operator, that is responsible for removing the object from the scene following its correct classification by the system. For the interactive perception model, it is pivotal to obtain a measure of confidence for a class label in the proposed research, which also serves as the decision strategy towards the interaction stage. In terms of classification accuracy, a Gaussian

Process-based multi-class classifier has been cross-validated by comparing the classification results with an SVM multi-class classifier for a similar dataset.
The operating scenario adopted comprises a visual identification of objects lying directly on the table, such as a bottle(s) or mug(s), potentially partially occluded. After one shot recognition, a list of identified and recognized objects is maintained along with their respective probabilistic confidences. Based on this scenario, the proposed visual architecture is validated by conducting experiments, to sort household objects, on the basis of the class predictions obtained from the GP based multi-class classifier with the highest probability. By opting this strategy, the scene becomes simpler after each recognized object with high confidence is removed. The visual architecture proposed in this chapter claims to make the following contributions to the state-of-the-art in visually guided object sorting system:

1. the first example of research to adapt non-parametric multi-class probabilistic classification (via Gaussian Processes) to the household object recognition problem by performing household objects sorting task.

2. a demonstration of the proposed GP-IPM approach applied to a semi-autonomous sorting task yields substantially improved performance over non-interactive alternatives.

This chapter is organized as follows: Section 5.2 presents motivation and objectives of the investigation. Section 5.3 presents the proposed GP classification based interactive perception model. Finally, Section 5.4 and Section 5.5 describe the preliminary experimental validation of the system and concluding remarks of this chapter, respectively.

5.2 Motivation and objectives

In an ideal scenario, robots should be capable of detecting, recognizing and classifying objects within their environment, and then depending on the task at hand, interacting with these objects without the need for supervision. When understanding challenging scenes, conventional single shot recognition either fails or have low success rates. This makes the task of object manipulation for robots difficult.

In [Gibson, 1966], the authors argued that physical interaction further augments perceptual processing beyond what it can be achieved by invoking deliberate pose changes. Building on this premise, this chapter argues and demonstrates that the interaction with the scene and objects can significantly improve the success rate of detecting object classes in front of a service robot. This is because interactive perception allows the perception-behavior to acquire more information about the object(s) that is present in the environment. Furthermore, It has the potential to reduce the complexity of the observed scene, in this case, encoded within a heuristic that directs the robot to grasp an object under investigation and separate it from other objects prior to making an attempt at re-classifying it. This, in turn, requires, development of a visual model for scene understanding in the context of robotic interactive perception, that is capable of providing information in addition to an object category, i.e prediction confidence. Such knowledge extends the capability of a visual architecture for accurate scene understanding by interaction with the scene, and hence, enabling it to reason over a given object class, i.e to accept it as a true class or investigate further before it is confirmed as a true class. This chapter describes an overall visual architecture for scene understanding by interaction for an autonomous system. Specifically, it describes the details of the perception stage of the proposed visual architecture for scene understanding by robotic interactive perception and demonstrates its application by carrying out the task of the semi-autonomous objects sorting system.

5.3 Methodology

This section describes the proposed approach in detail which consists of interleaving five stages:

- Segmentation of an input point cloud captured for a scene containing household objects of different shape, size, color, and texture, into individual objects.
- Visual representation of each segmented object is created.
- Category classification based on Gaussian Process multi-class classifier is carried out to provide the class label for each object along with respective prediction confidence.
- A semantic visualization is used for meaningful visualization of the scene, where objects are color-coded according to their respective categories.
- For this initial attempt at performing the task of objects sorting, a human operator is needed to pick the object from the tabletop and place it into a bin designated for the object's class.

The proposed visual system for category-based objects sorting is implemented, and cross-validation is performed by means of simulating GP based classification against SVM based classification for classification accuracy. Furthermore, a demonstration is carried out on two dual-arm industrial robots, 1. Yaskawa Motoman industrial robot, that is equipped with stereo cameras, a pair of Xtion pro cameras and tactile sensing gripper and a smart grasping system, and 2. Baxter robot. The proposed system utilizes the depth images from one of the Xtion pro cameras mounted on each arm of the robot which serves as an input and triggers the rest of the pipeline of the system. The visual system pins out the category of the objects lying on the table directly, and the system recognizes the category of the object and in addition, produces a color-coded semantic representation of the scene contents. For implementation purpose, the visual system is implemented using the Point Cloud Library (PCL) [1] and integrated into ROS. In the following sections, the components of the pipeline are described briefly.

5.3.1 Segmentation

An input point cloud is first processed for points within a defined range by a pass-through filter [2]. This filter enables the visual system to define a range in order to disregard the points that are not of interest. This allows the proposed visual system to process only the 3D points of the scene which belong to objects on the table. Initially, segmentation is carried out by detecting the table plane solely from the depth information. The operating table is segmented by using Random Consensus Sampling (RANSAC) [Fischler and Bolles, 1981] which estimates all the points in a point cloud for a model plane. Afterward, objects lying on the table are segmented by computing a convex hull from the plane coefficients, and it extrudes a certain height to create a prism, and give back all points that are lying inside. This process of segmentation for an example scene (Fig. 5.4) is shown in Figures 5.3, 5.5 and 5.6.

5.3.2 Visual representation

Local statistical 3D shape features are extracted for each of the object obtained from segmentation. These features are then encoded to create a dictionary containing codes for all the trained objects. For each extracted object, a visual representation is acquired by means of the Bag-Of-Words model. For low-level features, local 3D curvature features are computed by means of the Fast Point Feature Histogram (FPFH) [Rusu et al., 2009a]. For each query point, p_q and its neighbors, a set of tuples α, ϕ, θ are computed that results in Simplified Point Feature Histograms (SPFH). Subsequently, for each point, its k neighbors are re-evaluated, and the neighboring SPFH features are used to weight the final histogram p_q (called FPFH) as follows:

$$FPFH(P_q) = SPFH(P_q)\frac{1}{k}\sum_{i=1}^{k}\frac{1}{w_k}.SPFH(P_k)$$

where the weight ω_k represents a distance between the query point p_q and a neighbor point p_k in some given metric space, thus scoring the (p_q, p_k) pair, but could just as well be selected

[1] www.pointclouds.org

[2] http://pointclouds.org/documentation/tutorials/passthrough.php

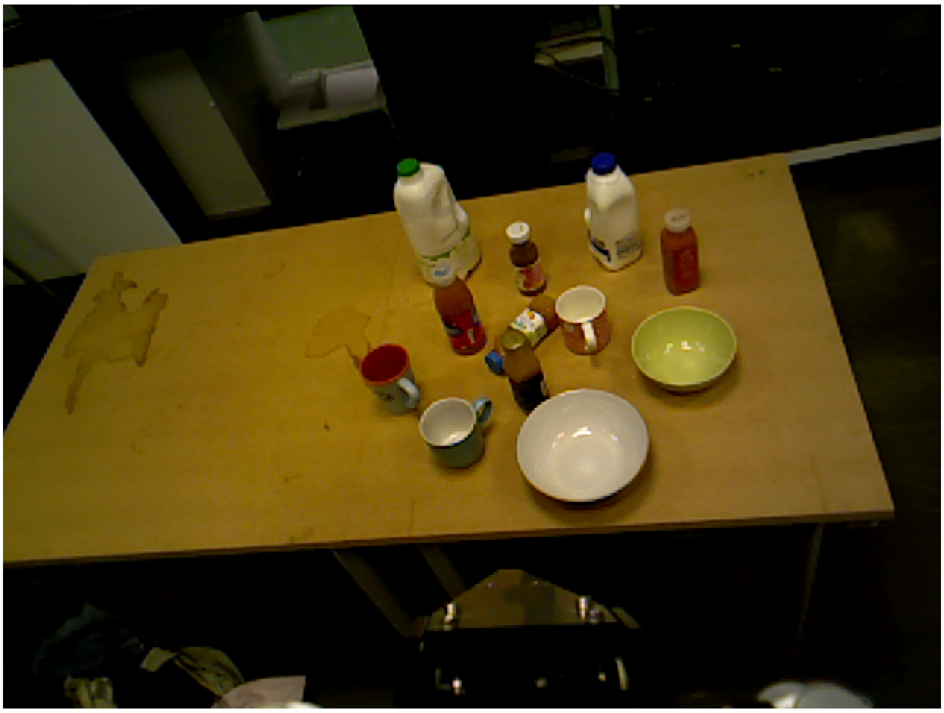


Figure 5.2: An RGB image of an example scene.

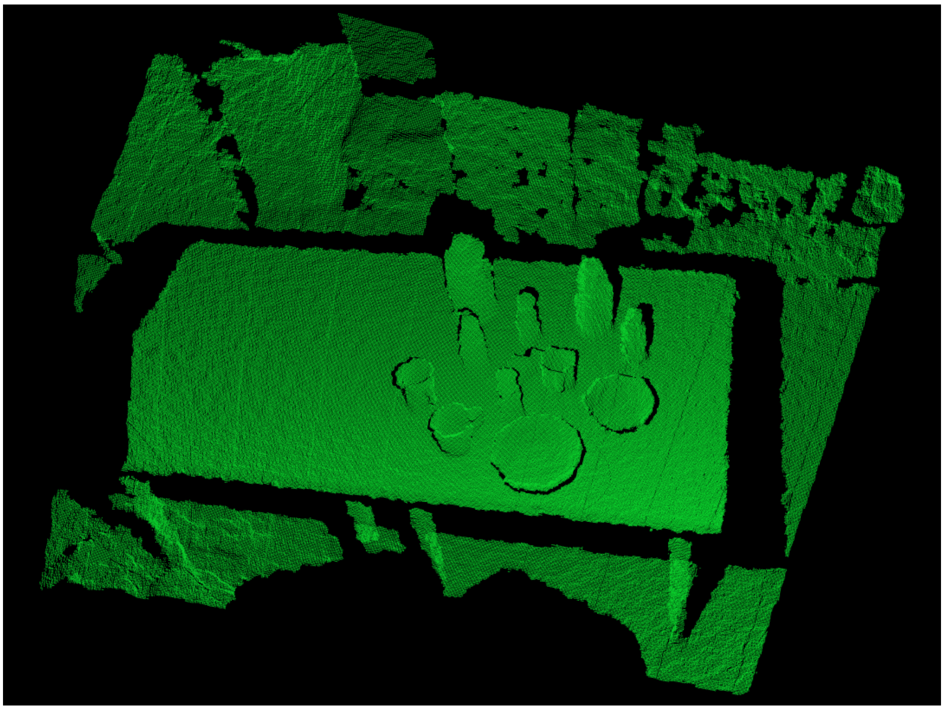


Figure 5.3: Point Cloud of the above example scene.

as a different measure if necessary. To understand the importance of this weighting scheme, the figure below presents the influence region diagram for a k-neighborhood set centered at p_q.

At the training stage, after features are extracted for all the views and for all the objects,

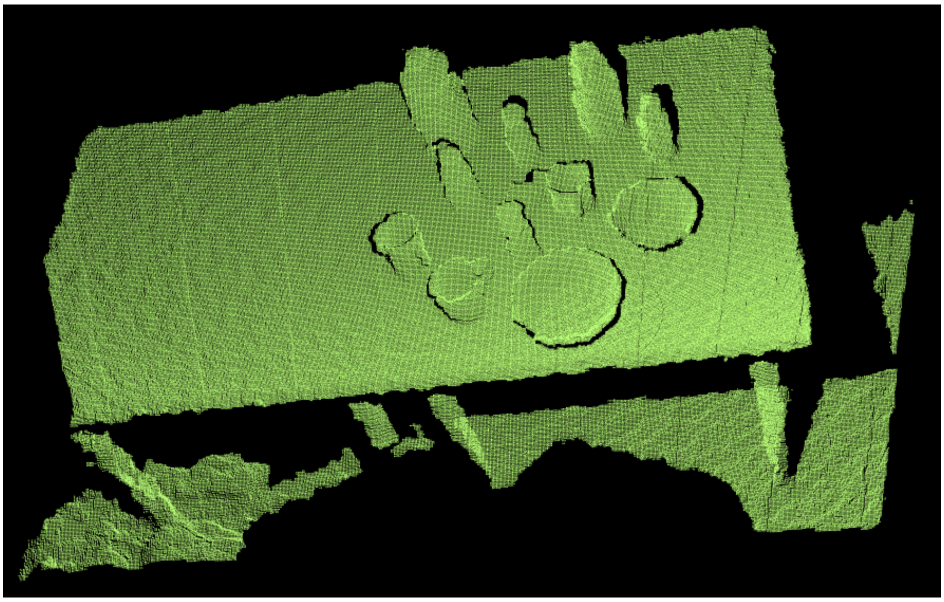


Figure 5.4: Result of pass-through filter of an example scene.

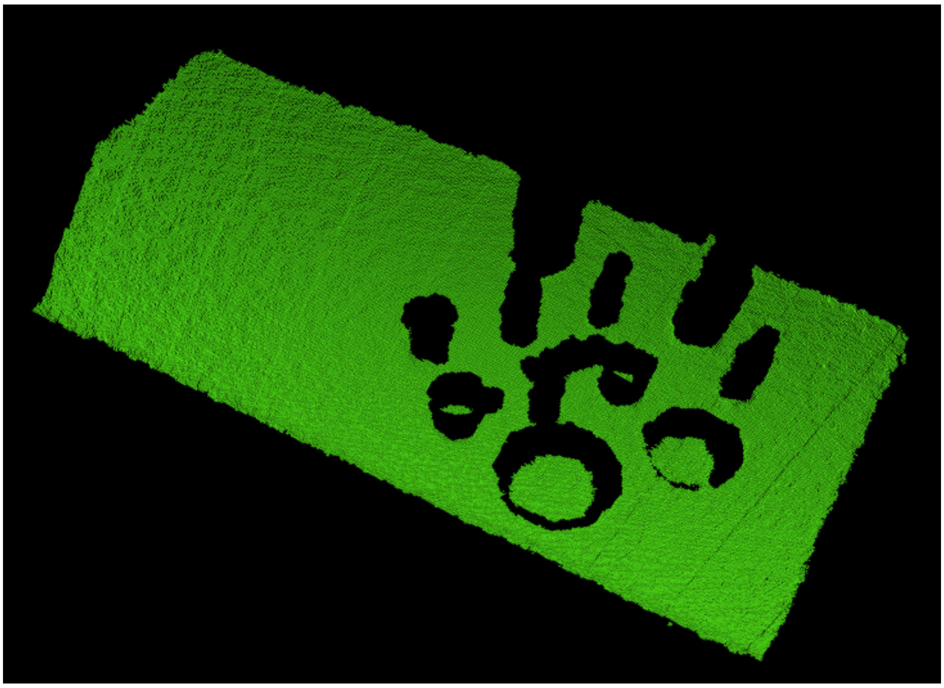


Figure 5.5: Point Cloud of the segmented plane of the above example scene.

we perform K-means clustering over all the vectors to obtain feature codes from the centers of the computed clusters. We set 256 K clusters as the size of our codebook. A putative visual representation for each object is then obtained by comparing all the features in the codebook generated by finding the minimum Euclidean distance, max pooling, and finally l2 normalization, resulting in 256 dimensional long vectors for each pose of an object. At the test phase, features extracted for all the objects are encoded using the codebook generated and fed into the classifier.

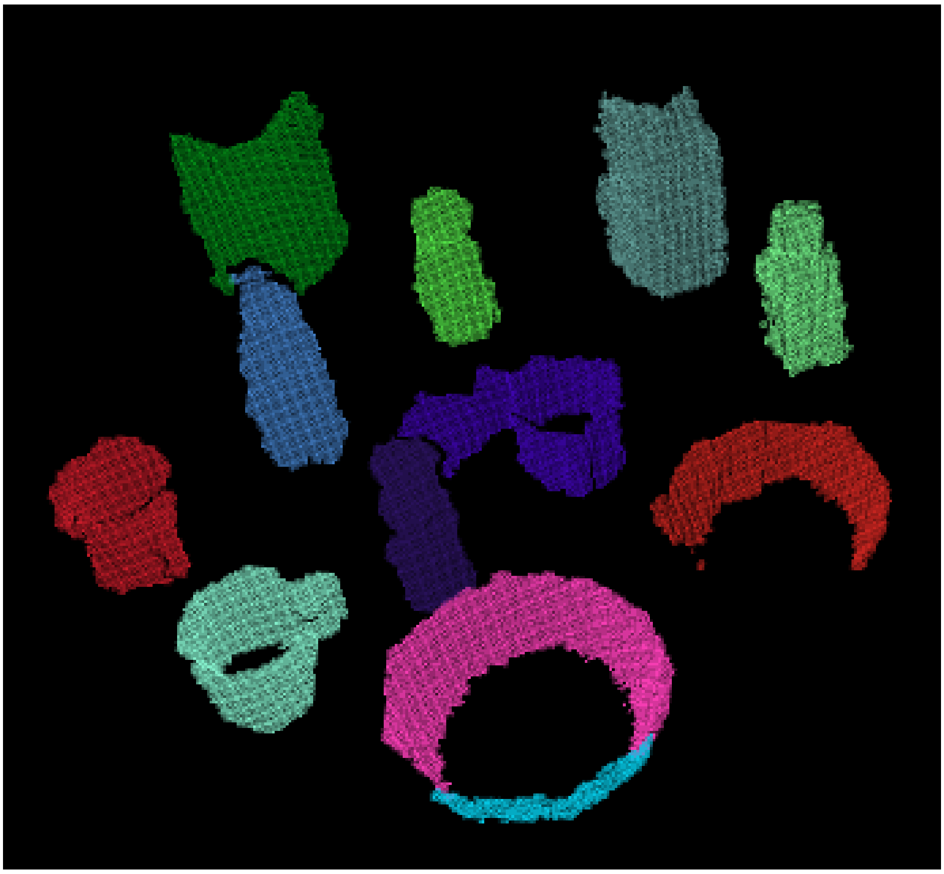


Figure 5.6: Point Cloud of the segmented objects of the above example scene.

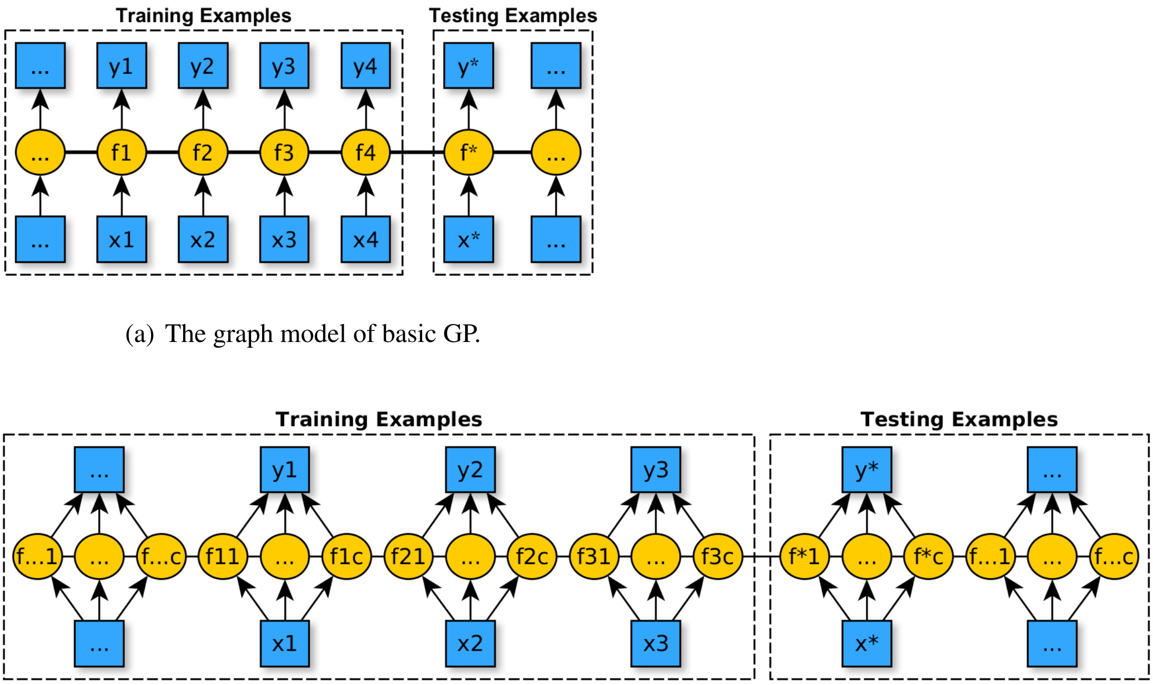
5.3.3 Classification

Since SVM-like classifiers cannot generate true probabilistic confidence through one-vs-one or one-vs-all voting for the multi-class classification problem, in order to model the distribution of probabilistic confidence among all object categories, multi-class Gaussian Process Classification [Rasmussen] is employed.

In this approach, RBF kernel is used as the kernel function, the posterior of the Gaussian Process is estimated by straightforward Laplace approximation. The RBF kernel we use has two hyper-parameters:

$$kernel_(x_1, x_2) = \alpha^2 \exp^{-\frac{1}{2}(x_1 - x_2)^T diag(\frac{1}{\beta^2},\ldots,\frac{1}{\beta^2})(x_1 - x_2)}, \tag{5.1}$$

where x_1 and x_2 are two examples, and α and β are the two hyper-parameters controlling the scale and shape of the exponential function. These hyper-parameters are optimized by the log marginal likelihood maximization. For completeness, we briefly present the classification based on the Gaussian process to model the distribution of predictive probabilities among all object categories. A more detailed description can be found in previous work Sun et al. [2016c]. The advantages of kernel-based approaches (as found in SVMs) with a prin-



(a) The graph model of basic GP.

(b) The graph model of GP regression and binary classification.

Figure 5.7: The difference between basic binary GP model and the multi-class classification model. In these Figures, x refers to examples, y refers to labels and f refers to latent variables. In (b), fij refers to f_i^j, which is the jth latent variable of the ith example.

cipled probabilistic framework are incorporated by using multi-class Gaussian process (GP) classifiers. Unlike SVM, the GP is probabilistic by default and can provide with the probability distributions across all objects categories. A multi-class GP classification is used, with the standard Laplace approximation to estimate the posterior and covariance hyper-parameters optimized by maximizing the log marginal likelihood. This approach closely follows that described in [Rasmussen] where their hyper-parameter optimization is extended from the binary case to the multi-class case by one-vs-all or one-vs-one voting using binary classifiers, the class-conditional distributions within multi-classification problem are unlikely to be well-modeled. Therefore, the multi-class GP classification [3] with hyper-parameter optimisation is integrated into the proposed visual system.

The rules for the symbol usage are illustrated here. Upper-case denotes matrices and lower-case denote vectors. The symbols with subscript and superscript '*' refer to testing examples. Other subscripts and superscripts refer to vector and matrix indices.

We start with a short introduction of GP. GP is a non-parametric model; a collection of latent variables (numeric values). For a basic GP model problem, as shown in Fig. 5.7(a), each example has a latent variable. Given the training examples X and testing example x_* and

[3] https://kevinlisun@bitbucket.org/kevinlisun/multi-class-gpc.git

their latent variables f and f_*. For the regression problem, the latent variable refers to the mean of the target, and for the classification (binary case) problem it can be squeezed into a sigmoid function to get probabilities of categories. In the regression problem, in order to predict a testing example, the latent value of the testing example can be estimated by the conditional probability of f_* given the joint distribution on training and testing examples $f_*|X, x_*, f$; these can be straightforwardly calculated since the joint distribution of f and f_* is $\begin{smallmatrix} f \\ f_* \end{smallmatrix} \sim \mathcal{N}(0, [\begin{smallmatrix} K_{XX} & K_{Xx_*} \\ K_{x_*X} & K_{x_*x_*} \end{smallmatrix}])$. The GP classification problem is introduced below.

In the classification problem, in order to predict a test example, the conditional probability $p(f_*|X, y, x_*)$ need to be solved, in which the key problem is to estimate the posterior of latent variables given on training examples and labels $p(f|X, y)$. For the multi-class classification problem, as shown in Fig. 5.7(b), each example will have C latent variables responding to C categories, and they will go to soft-max to generate the prediction probabilities. We detail the multi-class GP classification below.

For a classification problem, the GP classifier fits a real-valued latent variable to each observation. Jointly, the set of latent variables are given a GP prior (which typically enforces a degree of smoothness for the latent function over the input space). The classification probabilities are obtained by pushing the real values through a squashing function (e.g. the sigmoid function, soft-max function). The training phase consists of obtaining a posterior density over the latent function $p(f|X, y)$ (y is the training labels). Prediction consists of using this posterior to perform a regression to give the latent values at testing points, which are then squashed to provide predictive probabilities. To extend the GP to multi-class classification, one latent function is fitted for each of the C classes. The classification probabilities are obtained by squeezing the C function values for each observation through a soft-max function. To make predictions for a test point, C regressions are performed (one with each of the latent functions) and the resulting probabilities are pushed through the soft-max. The multi-class GP classification is detailed as follows.

In particular, given N training examples (with $\{n_1, n_2, ..., n_c\}$ examples in each class, $\sum_i n_i = N$), $X = \{x_1^1, ..., x_{n_1}^1, x_1^2, ..., x_{n_2}^2, ..., x_1^C, ..., x_{n_c}^C\}$, and corresponding labels are denoted $y = \{y_1^1, ..., y_N^1, y_1^2, ..., y_N^2, ..., y_1^C, ..., y_N^C\}$ where $y_i^c = 1$ if the ith example belongs to the cth class. This vector is therefore of length $Cn = C \times N$. In the description of this work, following [Rasmussen] the C sets of latent variables (each of length N) are concatenated into one Cn-length vector, f.

Ultimately, the prediction of the class of an unknown instance x_* needs to be solved as ([Rasmussen]):

$$P(y_*^c = 1|x_*, X, y) = \iint P(y_*^c = 1|f_*)p(f_*|f, x_*, X)p(f|X, y)\, df_*\, df. \qquad (5.2)$$

Now each of the terms on the right hand side is analysed in turn. The first term is the standard soft-max function:

$$P(y_*^c = 1|f_*) = \frac{\exp(f_*^c)}{\sum_j \exp(f_*^j)}, \tag{5.3}$$

where f_* is used to denote the C latent variables for the unknown instance. The second term in Eq. 5.2 is a standard noise-free GP regression. Defining the GP prior of this work with a zero mean function and kernel matrix K: $f|X \sim \mathcal{N}(0, K_{XX})$, and defining k_{x_*X} as the $1 \times N$ vector of the kernel function evaluated between the test point and all of the training points, and $k_{x_*x_*}$ as the kernel scalar evaluated at the test point, this is:

$$f_*|x_*, X, f \;\sim\; \mathcal{N}(k_{x_*X}K_{XX}^{-1}f,\; K_{x_*x_*} - k_{x_*X}K_{XX}^{-1}k_{Xx_*}). \tag{5.4}$$

In multi-class classification of GP, the covariance matrix K_{XX} is a $Cn \times Cn$ diagonal matrix consisting of C of $n \times n$ covariance matrices $\{k_{XX}^1, \cdots, k_{XX}^C\}$ on the diagonal corresponding to C classes. Similarly, K_{x_*X} and K_{Xx_*} are also diagonal matrices. The final term in Eq. 5.2 is the posterior density over the latent function for the training examples. In classification problems, this is not available in closed form, and the popular Laplace approximation [Williams and Barber, 1998] is used. This approximates the posterior with a multiple-variate Gaussian (in this case, a Cn dimensional Gaussian) centred at the maximum of the posterior and with covariance equal to the negative inverse of the Hessian matrix at the maximum.

$$p(f|X, y) \;\approx\; q(f|X, y) \;=\; \mathcal{N}(\hat{f}, -(\bigtriangledown\bigtriangledown \log p(f|X, y)|_{f=\hat{f}})^{-1}), \tag{5.5}$$

where $\hat{f}$ is the value of f that maximises the posterior and $\bigtriangledown\bigtriangledown \log p(f|X, y)|_{f=\hat{f}}$ is the Hessian of the log posterior distribution evaluated at the maximum.

Given the three terms in Eq. 5.2, it is possible to evaluate the integrals to obtain the required predictive probabilities. The conditional probability of f_*, given X, y, x_*, is:

$$p(f_*|X, y, x_*) = \int p(f_*|X, x_*, f)q(f|X, y)df, \tag{5.6}$$

where $q(f|X, y)$ is the Laplace approximation. As both $p(f_*|X, x_*, f)$ and $q(f|X, y)$ are Gaussian distributions (Eq. 5.4 and Eq. 5.5), it is possible to analytically evaluate this integral. The mean of the resulting Gaussian $\mu = \{\mu_1, \cdots, \mu_C\}$ in which each μ_c can be calculated by:

$$\mu_c = (k_{x_*X}^c)^T K_c^{-1} \hat{f}^c = (k_{x_*X}^c)^T (y^c - \hat{\pi}^c) \tag{5.7}$$

Then, the covariance matrix of the resulting Gaussian is:

$$\Sigma = diag(k_{x_* x_*}) - Q_*^T (K + W^{-1})^{-1} Q_*, \tag{5.8}$$

where W is the matrix containing second order partial derivatives of $\log p(y_i^c|f_i)$. Similar to K_{XX}, Q is the diagonal matrix $diag\{k^1_{x_* X}, ..., k^C_{x_* X}\}$, and $k^c_{x_* X}$ is the vector of covariance between the testing example and training examples with respect to the cth category.

Because of the form of the softmax function, evaluating the integral over f_* is not analytically tractable but is easily approximated via sampling from the predictive distribution over f_*. In particular, if S samples of the C latent variables are drawn, and $f_*^{c_s}$ donates sth sample, the predictive probability can be calculated as:

$$P(y_*^c = 1|X, x_*, f) \approx \frac{1}{S} \sum_{s=1...S} \frac{\exp(f_*^{c_s})}{\sum_j \exp(f_*^{j_s})}. \tag{5.9}$$

5.3.4 Hyper-Parameters optimisation

The square exponential kernel function (SEiso) is used:

$$k_{SEiso}(x_1, x_2) = \alpha^2 \exp^{-\frac{1}{2}(x_1 - x_2)^T diag(\frac{1}{\beta^2}, ..., \frac{1}{\beta^2})(x_1 - x_2)}, \tag{5.10}$$

where α and β are hyper-parameters of the kernel function. Sensible choice of hyper-parameters is crucial to getting good performance. Following [Rasmussen], the kernel parameters are optimised via maximising the Laplace approximation to the marginal likelihood (this could also be achieved by a cross-validation procedure). Broyden-Fletcher-Goldfarb-Shanno [Shanno, 1985] algorithm (BFGS) is employed for the optimisation. Since the BFGS is a derivative based optimisation method, the derivatives of the likelihood are necessary to be computed.

A pre-trained Gaussian process based classifier is used to work on the resultant encoded visual representation for each object from the stage above in order to predict the category of the object along with probability confidences. These confidences are then used to make a decision of the identity of the object. The visual system described in this chapter has been tested with two multi-class classifiers, Support Vector Machines and Multi-class Gaussian Process models.

5.3.5 Semantic visualization

This provides a semantic representation of the results from the classifier by color coding each object according to their respective categories. A color-coded visual representation is used for objects via the RVIZ tool available in ROS (http://wiki.ros.org/rviz), where each color corresponds to a category and probabilities produced by the classification step are used to place the object on the operating table in their respective bins in order of higher probabilistic confidences.

5.3.6 Human operator

After one shot recognition of all the objects present in the scene, the multi-class GP based classification, results in class label along with respective probabilistic confidences. The recognized objects are needed to be removed from the table top. Each object is removed by a human operator from the table and placed into its respective bin by observing the color-coded semantic representation that describes each object's label along with confidence.
A pseudo-code Algorithm of the proposed method is shown in Algorithm 1 [Khan et al., 2016b].

Algorithm 1 Interactive Objects Sorting

```
procedure INTERACTIVE OBJECTS SORTING
    segmentation into objects ← point cloud
    visual representation ← point cloud representing each object
    classification:confidences,labels ← visual representation
top:
    if no object on the table then return false
loop:
    if $confidence(i) > threshold$ then
        color code each object according to its class.
        pick and place object into bin with highest confidence.
        goto loop.
    goto top.
```

5.4 Experiments

These experiments are designed to demonstrate the validity of the proposed Gaussian Process-based interactive perception model, capable of improving the recognition rate with the aid of interaction with objects by demonstrating the task of semi-autonomous object sorting. These experiments have been carried out for five household object categories i.e Juice Bottles, Milk Packs, Bowls, Mugs and Juice Boxes. Scenes have been arranged based on combinations of known and unknown object instances, of 5 different object classes. Objects were placed in arbitrary poses and locations as shown in Fig. 5.8, which is a semantic visualization of the recognition and classification performed by the perception stage. In Fig. 5.8, the red color is for juice bottles, green for milk boxes and yellow for mugs. The training dataset consists of the above 5 object classes – this dataset is created by capturing point clouds of each object at angular intervals of approximately 20 degrees. In this section, the proposed GP-based perception model is evaluated for objects sorting into their respective bins. Note that the objects are placed directly on top of the table. An input point cloud is obtained from the ASUS Xtion pro camera from a single view. The point cloud is subsequently passed on to the pipeline.

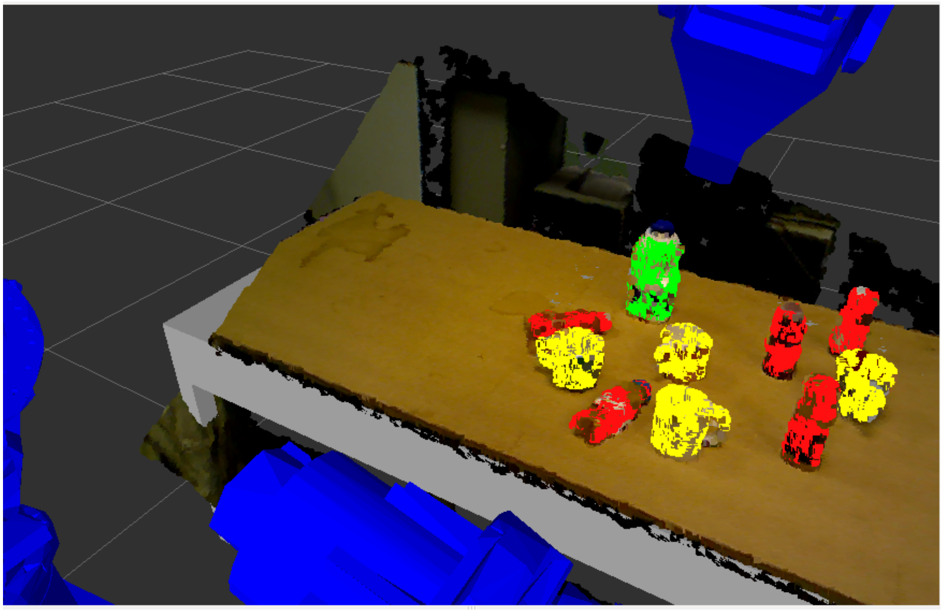
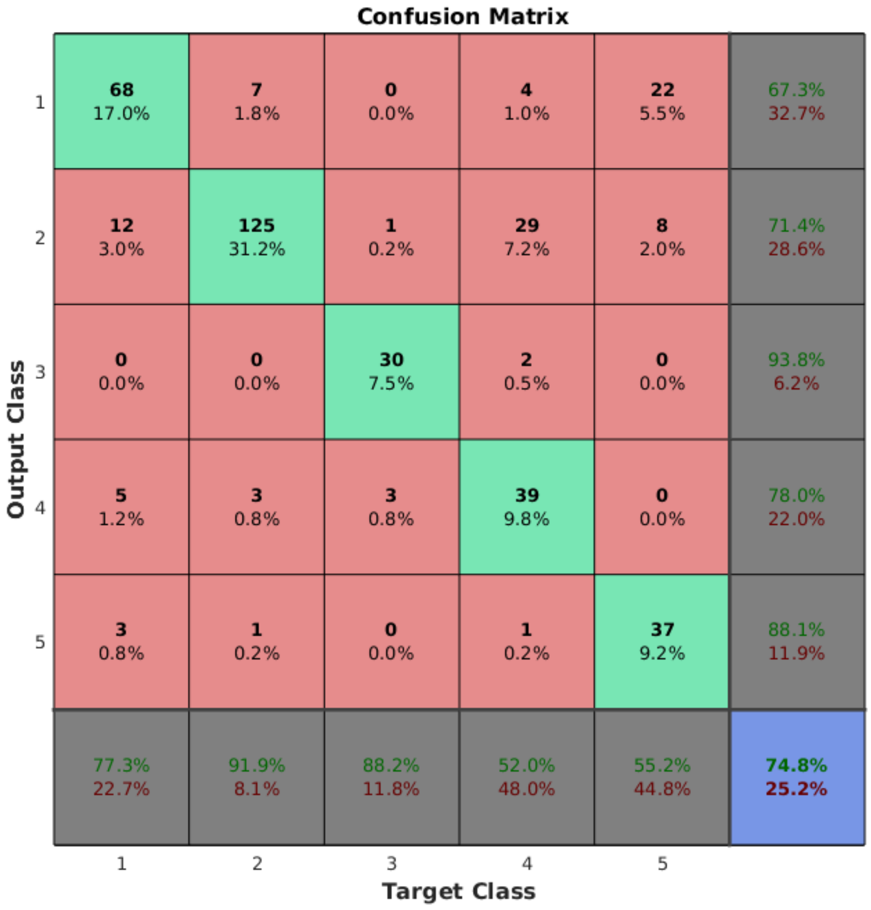


Figure 5.8: Robot recognising object categories according to colours [Khan et al., 2016b].

As stated before, these experiments consisted of evaluating simulation results on an SVM multi-class classifier and a GP-based classifier as shown in Fig. 5.10 and Fig. 5.9, respectively. These Figures present confusion matrices which are tables often used to describe the performance of a classification model (or ”classifier”) on a set of test data for which the true values are known. The diagonal of each confusion matrix, colored in green, shows the true positives and the bottom row gives the recognition rate for each category. Figure 5.11 shows the prediction confidences, where each vertical line corresponds to an object category i.e red, blue, black, green, yellow, respectively, different color marks outside its group, refer to incorrect predictions.

For the SVM multi-class classifier, an average recognition rate of 75% and for GP based multi-class classifier, 79% are obtained, by calculating (true positives + true negatives)/total number of samples. It can be noticed that the bowl’s category has the lowest recognition rate which in turns affects the overall recognition rate. This is due to the bowl’s intrinsic shape as it contains less visual information as well as consisting of a highly reflective surface, to which depth sensing is very sensitive (i.e. limitation of the ASUS Xtion pro camera).

To demonstrate the viability of the proposed GP based interactive perception model for improving the recognition rate towards the task of object sorting, two scenes have been con-



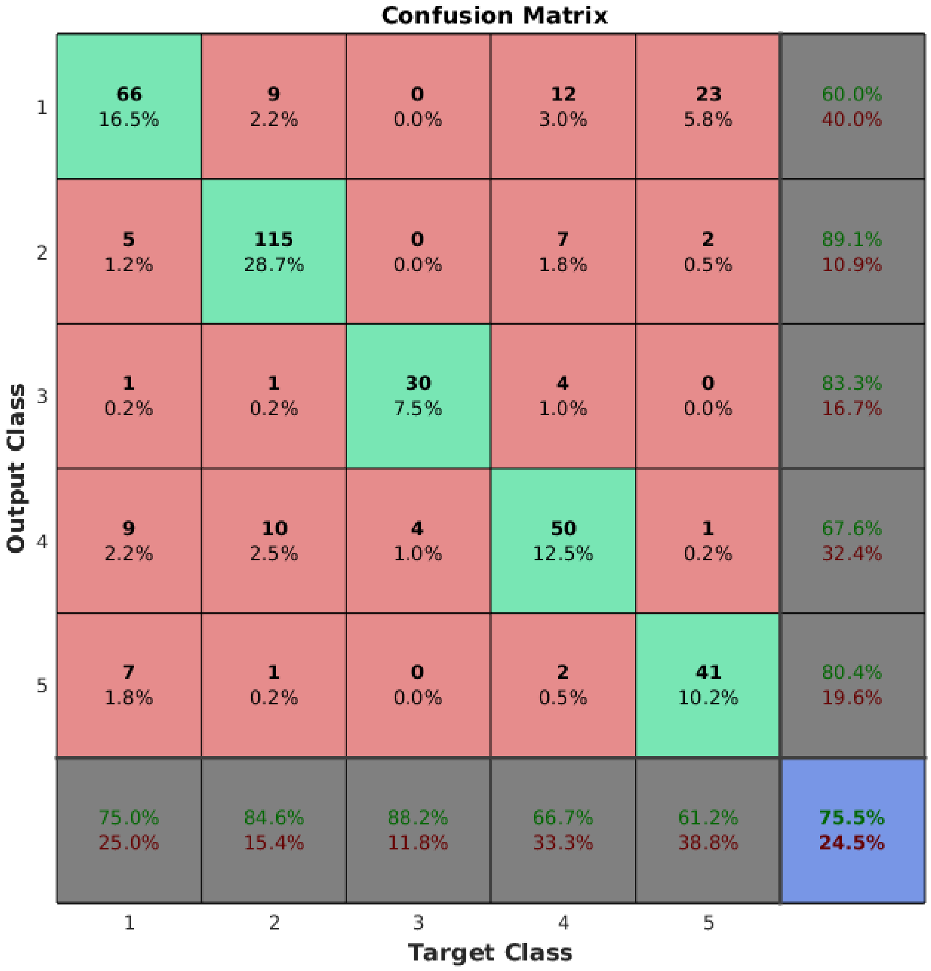
Confusion Matrix

Output Class \ Target Class	1	2	3	4	5	
1	**68** 17.0%	**7** 1.8%	**0** 0.0%	**4** 1.0%	**22** 5.5%	67.3% 32.7%
2	**12** 3.0%	**125** 31.2%	**1** 0.2%	**29** 7.2%	**8** 2.0%	71.4% 28.6%
3	**0** 0.0%	**0** 0.0%	**30** 7.5%	**2** 0.5%	**0** 0.0%	93.8% 6.2%
4	**5** 1.2%	**3** 0.8%	**3** 0.8%	**39** 9.8%	**0** 0.0%	78.0% 22.0%
5	**3** 0.8%	**1** 0.2%	**0** 0.0%	**1** 0.2%	**37** 9.2%	88.1% 11.9%
	77.3% 22.7%	91.9% 8.1%	88.2% 11.8%	52.0% 48.0%	55.2% 44.8%	**74.8%** **25.2%**

Figure 5.9: Confusion matrix shown obtained from multi-class gp classifier for 5 object categories.

structed; (1) without any occlusion and simple objects, and (2) with occlusion among objects and challenging objects such as bowls. Also, for each scene, two types of sorting operations are investigated. First random sorting is considered, where the output class of the classifier along with the highest object is first removed and placed into the respective bin. Second, by using GP based interactive perception model, the object is removed from the table based on confidence probability. Fig. 5.8 shows the output of the proposed visual system with the help of colors representing categories.

After all the objects have successfully been recognized and classified, objects were manually placed into their respective bin. Table 5.1 shows the success rate for the objects placed into the respective bin guided by the proposed GP-IPM. For complex scenes, containing objects having challenging shapes and suffering from occlusion, the proposed method outperforms random sorting by a factor of 30%.



	1	2	3	4	5	
1	**66** 16.5%	**9** 2.2%	**0** 0.0%	**12** 3.0%	**23** 5.8%	60.0% 40.0%
2	**5** 1.2%	**115** 28.7%	**0** 0.0%	**7** 1.8%	**2** 0.5%	89.1% 10.9%
3	**1** 0.2%	**1** 0.2%	**30** 7.5%	**4** 1.0%	**0** 0.0%	83.3% 16.7%
4	**9** 2.2%	**10** 2.5%	**4** 1.0%	**50** 12.5%	**1** 0.2%	67.6% 32.4%
5	**7** 1.8%	**1** 0.2%	**0** 0.0%	**2** 0.5%	**41** 10.2%	80.4% 19.6%
	75.0% 25.0%	84.6% 15.4%	88.2% 11.8%	66.7% 33.3%	61.2% 38.8%	**75.5%** **24.5%**

Figure 5.10: Confusion matrix shown obtained from multi-class svm classifier.

Table 5.1: Table showing true positives for Random objects sorting and GP-IPM based objects sorting. [Khan et al., 2016b]

Scene Complexity	**Method of Sorting**	**True Positives**					**Total**
		Juice Bottles	**Milk Cartons**	**Bowls**	**Mugs**	**Juice Boxes**	
Uncluttered	Random	5/5	1/1	-	4/4	-	9/9
	GP-IPM	5/5	1/1	-	4/4	-	9/9
Cluttered	Random	2/4	1/2	2/3	2/3	1/2	8/14
							57%
	GP-IPM	4/4	2/2	2/3	3/3	1/2	12/14
							86%

5.5 Conclusion

This chapter has described an interactive perception model for object sorting system, incorporating depth sensing, 3D feature extraction, object representation, and classification. The proposed visual interactive perception system achieves real-time object category identifica-

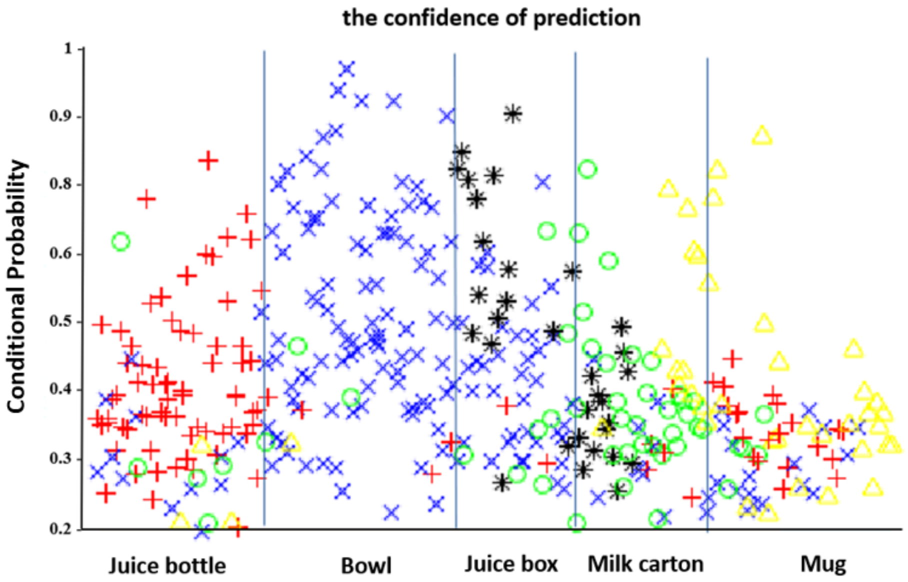


Figure 5.11: The classification performance under different confidences for 5 object categories. In this Fig., the confidence of the predictions are shown, in which each column corresponds to a object category. The correct prediction should be red, blue, black, green and yellow, respectively.

tion with associated perception confidence estimates, which serves to establish whether an observation is sufficient to make a classification or to determine if it is necessary to interact with the object to achieve a sufficiently reliable classification of the instance contained in the observation.

A dataset of household objects used in the evaluation during this investigation is established, and the experimental results show that the proposed perception pipeline achieves 80% classification accuracy for 5 object categories. As the autonomous robot manipulation phase is presented in the following chapter, this chapter focused on the visual perception components of the overall proposed system and presented sorting results obtained using a human operator to manipulate the objects. From these semi-autonomous sorting experiments, the proposed Gaussian Process-based interactive sorting system outperformed random sorting by up to 30% in terms of sorting accuracy. It has also been observed that the proposed interactive perception strategy not only mitigates segmentation failures prevalent in single-shot sorting, but it also increased perception performance in terms of recognition rate, thereby facilitating the sorting decision.

The proposed approach can be extended by integrating the perception model with an interaction model, to develop an autonomous visual architecture for scene understanding by robotic interactive perception. Where the task of objects sorting, demonstrated in this chapter by a semi-autonomous interactive perception model based on GP based classifier, can achieve complete autonomy. This is detailed in Chapter 6.

Chapter 6

Objects Manipulation - Interactive Perception

This chapter of the thesis proposes a fully autonomous integrated visual architecture for scene understanding by robotic interactive perception. Building on the premises of the last chapter, a modified perception stage is described. Following, the perception stage, an interaction stage is discussed in detail that performs a set of ad-hoc actions relying on the information received from the perception stage. More specifically, the interaction stage simply reasons over the information (class label and associated probabilistic confidence score) received from the perception stage to chose one of the following two actions. 1) An object class has been identified with high confidence, requiring removal of the object from the scene and place it in the designated basket/bin for that particular class. 2) An object class has been identified with less probabilistic confidence, since from observation and inspired from human behaviour of inspecting doubtful objects, an action is chosen to carry out more investigation for the object - confirm the object identity by taking more images from different views in isolation and perform operations of perception stage for those views, hence multiple perception-action/interaction cycles take place. From an application perspective, the task of autonomous category based objects sorting system is performed and the experimental design for the task is described in detail.

6.1 Introduction

In this Chapter, the benefits of perception by adding interaction into it are explored to simplify and improve the accuracy of understanding a real-world challenging scene made of household objects by an autonomous system. As stated before, essentially service robots need to have the ability to recognize objects in their immediate vicinity when working in dynamically evolving human environments. Ideally, these robots should be capable of detecting and classifying objects within their environment and then interacting with these objects without the need for human supervision to achieve the task at hand. Traditional vision techniques in uncontrolled lighting and complex environments tend to yield low accuracy in understanding a scene without interaction or feedback. When understanding challenging scenes, single shot recognition fails or have low success rates without manipulating objects. The authors in [Gibson, 1966], argue that physical interaction further augments perceptual processing beyond what it can be achieved by invoking deliberate pose changes. This is because interactive perception allows the perception module to acquire more information about the object(s) in the environment.

Based on the proposed approach, it is also argued that interaction with the scene and objects can significantly improve the success rate while detecting object classes in front of a service robot. It also has the potential to reduce the complexity of the observed scene, in this case, encoded within a heuristic that directs the robot to grasp an object under investigation, separate it from other objects prior to investigating it further. Therefore, a better understanding of a scene i.e improving classification accuracy is achieved by robotic interactive perception. For the validation of the proposed autonomous visual architecture by robotic interactive perception, the proposed interactive perception pipeline demonstrated the task of sorting everyday household objects into bins, assigned to these objects according to their category. The proposed interactive perception pipeline does not require prior knowledge about the environment or scene, and is capable of recognizing objects in the scene by means of active object manipulation. Fig. 6.1 illustrated the proposed visual architecture. To recap from the previous chapter, the visually assisted object sorting system is capable of segmenting a set of household objects lying directly on the robot's workspace table and performs categorizing those objects into their respective object classes (fie.g. juice bottles, mugs, etc.). The system has been pre-trained on a subset of these object instances, while the other subset of the objects investigated have not been used to pre-train the system. The perception model of the proposed autonomous interactive perception system consists of the same capabilities as to the one presented in Chapter 5, i.e portable, invariant to 6 DOF object pose changes and operates in real-time. The perception stage of the interactive perception pipeline comprises of a modified object segmentation module, visual representation, classification, semantic visualization and autonomous robotic manipulation. For completeness and comparison, the perception stage of the interactive perception pipeline, together with a multi-class Gaussian process classifier, has been cross-validated by comparing the classification results obtained using an SVM multi-class classifier to ground truth. The reason for the choice of choosing GP based multi-class classifier over SVM like classifier is not only that GP based classifier gives equal or better classification accuracy but also, provides with classification confidence which plays a key role in the proposed architecture. It is also claimed that for the first time

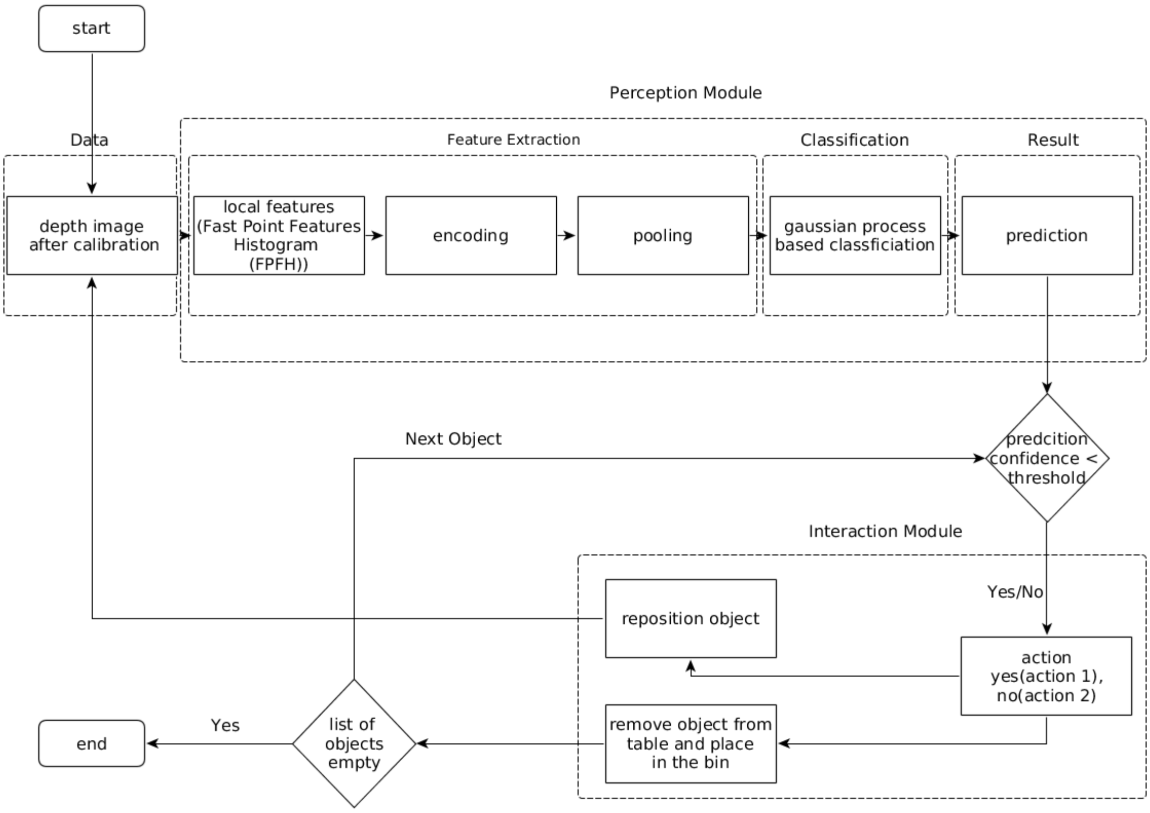

start
Perception Module
Data
Feature Extraction
Classification
Result
depth image after calibration
local features (Fast Point Features Histogram (FPFH))
encoding
pooling
gaussian process based classficiation
prediction
Next Object
predcition confidence < threshold
Interaction Module
Yes/No
reposition object
action yes(action 1), no(action 2)
end
Yes
list of objects empty
remove object from table and place in the bin

Figure 6.1: The proposed visual architecture consisting of perception stage and interaction stage, working on the principle of multiple perception-action cycles for scene understanding tasks.

that interactive perception is based on GP classification for rigid objects.

An interaction stage follows the perception stage in the proposed visual perception architecture that orchestrates the set of object manipulation skills available. Two simple manipulation skills/actions are available for the interaction stage to chose from after receiving the required information from the perception stage: 1) An object class has been identified with high confidence, so the action of pick an object from the tabtop and place it in the designated basket/bin for that particular class. 2) An object class has been identified with less probabilistic confidence, since from observation and inspired from the human behavior of inspecting doubtful objects, an action is chosen to investigate that object further. This means, to confirm the object identity by taking more images from different views in isolation and perception stage performs the visual operation for those views, hence multiple perception-action/interaction cycles take place.

The operating scenario adopted in this thesis consists of completing the task of sorting and removing objects from a table-top into the object's designated bins according to their respective classes. After receiving point cloud from the ASUS xtion pro camera, of the scene, one shot recognition takes place. A list of objects identified along with their respective classification confidences is maintained, such as a bottle or juice box, which can potentially be partially occluded, have been influenced by varying lighting, background. Based on the classification label along with the object's corresponding classification values (i.e. confidence scores), objects are picked-and-placed into their respective bins. For objects with a classification confidence score lower than a pre-defined threshold, this object is actively explored by manipulating and interacting with the object. Based on the above scenario, this chapter presents results obtained from experiments where the object are sorted by the probabilities of the class predictions obtained from the GP based multi-class classifier.

Organization of the rest of the chapter is followed as Section 6.2 presents motivation and objectives for the investigation of the proposed visual perception architecture for scene understanding through robotic interaction. Section 6.3, briefly describes the hardware and software facilities used during the development of the proposed approach. Section 6.4 details the proposed methodology opted for developing the proposed visual perception architecture featuring an interaction stage as an addition and extension to the proposed visual perception stage presented in Chapter 5. Section 6.5 discusses the experimental approach taken to perform the task of objects sorting, using the proposed visual perception architecture involving interaction with objects in the scene. Finally, Section 6.6 concludes the chapter with remarks on the effectiveness and advancement to the state-of-the-art by the proposed architecture.

6.2 The Motivation and objectives

The investigation reported in Chapter 5 provides a perception stage that is viable for developing an autonomous visual architecture for scene understanding involving robotic interaction and providing the feedback to the perception stage. The main objective of this chapter is to study the effect of an autonomous system interacting with the scene while carrying out the task of scene understanding. Specifically, to answer the question of, what objects are in the scene, and if it struggles to identify an object, what it can do with it that will improve the object recognition following the principle of multiple perception-action cycles without the involvement of human.

6.3 Hardware and software details

This section provides details of different hardware and software used during the development of the proposed interactive perception system.

6.3.1 Hardware

This sub-section describes the hardware details of the different vision systems, robots, and workstation used and tested for the proposed visual perception architecture for scene understanding.

Vision systems

Since the proposed perception stage pipeline utilizes depth image as input, it requires a vision system capable of producing quality point clouds. Active vision system reported in Chapter 3 provides low-cost good quality RGB images and depth images for non-rigid objects such as clothes but, quality of the depth images produced for rigid objects is poor and is not suitable for the purpose of the proposed approach. RGBD cameras were the next in line to be tested. Cameras such as ASUS Xtion Pro, Mircosoft Kinect Xbox One, Intel ZR300, ZED stereo camera and ASUS Xtion 2 have been tested for generating point clouds of the scenes. ASUS Xtion 2 produced the best quality point cloud, however, it suffers from firmware issues and is not reliable. ASUS Xtion Pro and Kinect Xbox One, generate decent quality point clouds and both have been used.

Dual arm robots

The proposed approach has been employed both on Baxter robot (shown in Fig. 6.3 with vision system of Mircosoft Kinect Xbox One and on Yaskawa Motoman robot with ASUS Xtion pro. To carry out manipulation tasks precisely i.e estimating object's location with precision, and equipped with the smart grasping system as shown in Fig. 6.10 as one of the end effectors, Yaskawa robot has been the focus in designing the experiments to validate the proposed visual perception architecture.

Workstation

Workstation connecting with Yaskawa robot has the following specifications: **Processor**: Intel Core i7-7700K 4.2 GHz QuadCore 8MB Cache Processor.
Memory: Corsair CMK32GX4M2A2133C13 Vengeance LPX 32 GB (2 x 16 GB) DDR4 2133 MHz C13 XMP 2.0.
Graphics Card: Gigabyte GeForce GTX 1080 Ti NVIDIA Founders Edition 11 GB GDDR5 PCI-Express 3.0 352 Bit Graphics Card.

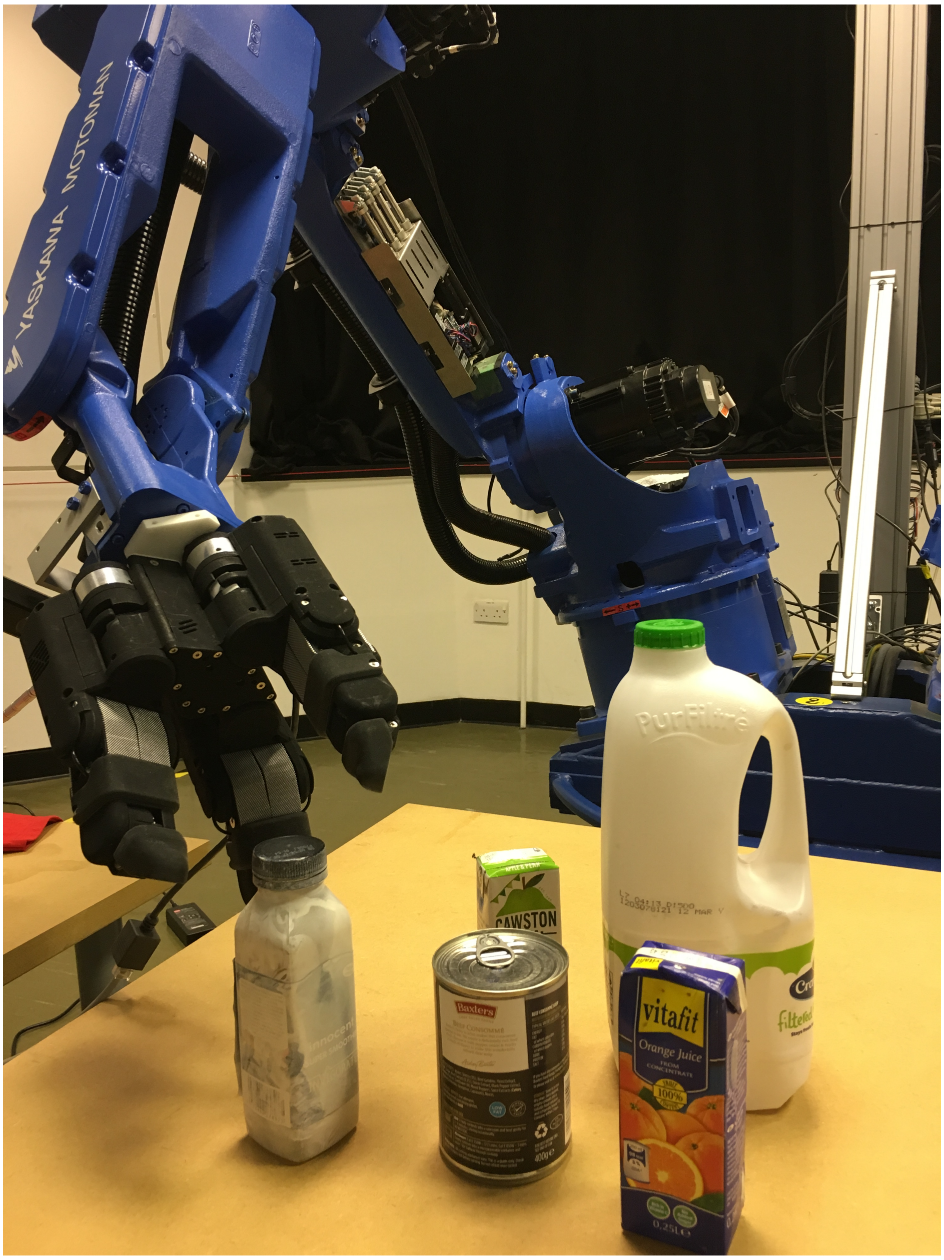


Figure 6.2: Smart Grasping System, installed on Yakawa Motoman Robot, making an attempt to pick an object in the scene.

6.3.2 Software

Visual perception architecture proposed in this chapter is developed using Robot Operating system (ROS), Point Cloud Library (PCL) and MATLAB with ROS toolbox and can be accessed at the following URL: `https://github.com/gerac83/isee_aamir`.

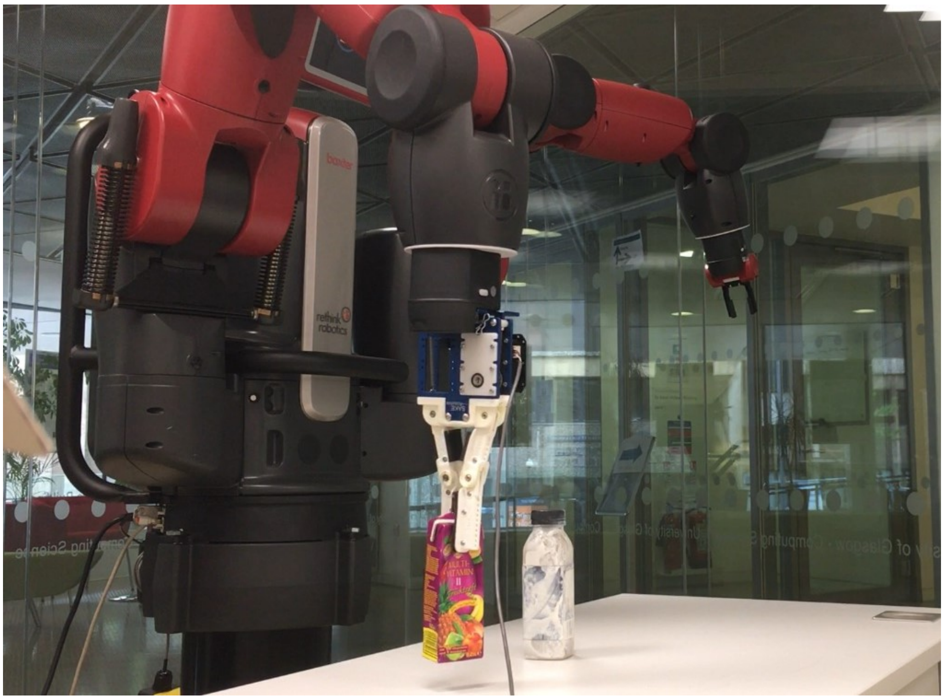


Figure 6.3: Visual perception architecture, investigating an object in the scene, employed on Baxter robot.

6.4 Methodology

The approach for implementing autonomous object sorting system based on interactive perception consists of two main stages: a perception stage and an interaction stage working in a loop until the given task in hand is accomplished i.e objects sorting according to their categories in this thesis. Where the output of the perception stage is fed into the interaction stage, and vice versa to accomplish the task of sorting, household objects placed on a table-top following perception-action cycles. The perception stage is responsible for object detection/segmentation and object recognition, and the interaction stage verifies object class labels by interacting with the object and picking the object from the table-top and dropping it in the bin designated for each class. Before the details of each of the two stages are given, it is worth revisiting camera and Hand-Eye calibration process which is needed for an autonomous system to fulfill the manipulation tasks using vision systems.

Camera and Hand-Eye calibration

Employing camera calibration, it is possible to measure the size of an object in world units in the scene. Therefore, the importance of camera calibration is realized in robotics, for navigation systems, and 3-D scene reconstruction. In addition, hand-eye calibration is needed to integrate cameras within a robot's kinematic frame chain. For RGB-D cameras such as ASUS Xtion Pro, default camera parameters are given. The hand-eye calibration process, specifically, to integrating such cameras into Yaskawa Motoman robot's kinematic frame is described by the Fig. 6.4 and discussed in Chapter 4. From Fig. 6.4, it can be deduced that the transform from robot's base to the camera can be obtained and integrated it into the robot's kinematic frame chain.

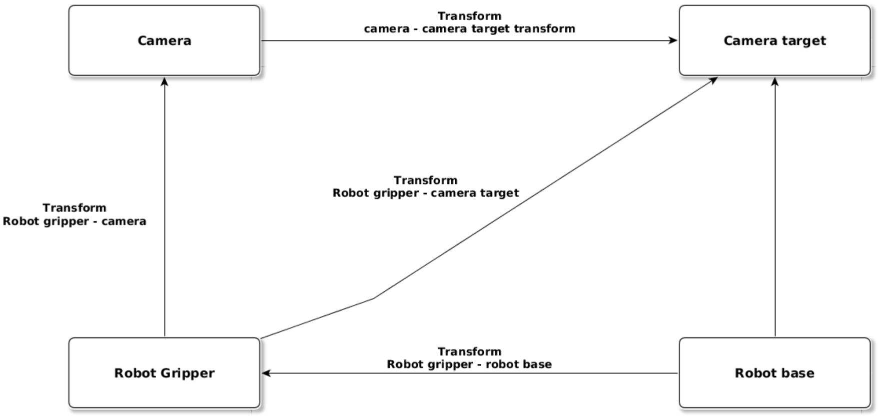


Figure 6.4: Handy Eye Caliration for intergrating camera into Yaskawa Motoman Robot.

6.4.1 Perception stage

The perception stage is made of three sub-stages: Modified Object segmentation from object segmentation discussed in the previous chapter, visual representation, and category classification. The visually assisted vision system for autonomous object sorting is implemented, and for comparison with SVM classifier in terms of classification accuracy, an evaluation is presented from cross-validating in the simulation. Also, a demonstration of the proposed approach is applied to Yaskawa Motoman robot. The proposed system utilizes the depth images from an ASUS Xtion pro camera mounted on one of the arms of the robot which serves as an input and triggers the rest of the pipeline of the system. In 6.5, the perception model implemented is shown. In the following sections, the segmentation component is described in detail which is modified from the corresponding component in Chapter 5 and for completeness other components of the perception stage of the proposed pipeline are briefly described.

Segmentation

To effectively utilize the feedback from the interaction stage, the object segmentation module works in two modes: Foveated and Non-Foveated, which are discussed briefly as:

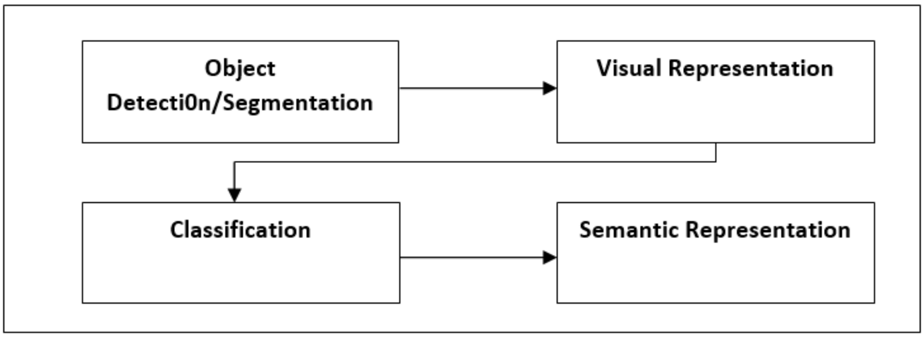


Figure 6.5: Perception stage of the proposed autonomous visual architecture for scene understanding applied to the task of object sorting.

Non-Foveated mode

In the non-Foveated mode, the object segmentation module the perception stage follows the same pipeline as discussed in the previous chapters, such as for an input point cloud of the complete scene, a pass-through filter is applied to keep points within a defined range that covers the main work-cell area i.e table and its boundaries. Next, table plane is detected from the remaining point cloud using Random Consensus Sampling (RANSAC) [Fischler and Bolles, 1981] which estimates all the points in a point cloud for a model plane. As a final step, objects lying on the table are segmented by computing a convex hull from the plane coefficients and create a prism of a maximum of 30 cm to only consider the objects on a table top and the maximum height of the objects for sorting task is less than 30 cm. This process of segmentation remains the same when one the perception stage is performing one-shot recognition.

Foveated mode

When the feedback received from the interaction stage indicates the visual perception architecture needs to investigate an object more i.w low predictive probability for a given object class, then, this foveated mode is enabled. This is achieved by applying a pass-through filter to keep only points that belong to the object. For this purpose two strategies have been employed dependent on interaction stage: 1) segmentation from a designated location on the table, 2) in-hand segmentation.

Segmentation from the designated location on the table is performed when the interaction stage has placed an object on a place on the table designated for objects that need more investigation and contains one object at a time. For this type of the segmentation, the pipeline in 6.4.1 is performed.

For **in-hand segmentation**, the interaction stage picks the object and holds it in front of the camera, the size of the object is used to segment the object from the gripper.

Visual representation

Low-level features are computed by Fast Point Feature Histogram (FPFH) [Rusu et al., 2009a] to represent local 3D curvature features. For each query point, p_q and its neighbors, a set of tuples α, ϕ, θ are computed that results in Simplified Point Feature Histograms (SPFH). Subsequently, for each point, its k neighbors are re-evaluated, and the neighboring SPFH features are used to weight the final histogram p_q (FPFH) as follows:

$$FPFH(P_q) = SPFH(P_q)\frac{1}{k}\sum_{i=1}^{k}\frac{1}{w_k}.SPFH(P_k)$$

where the weight ω_k represents a distance between the query point p_q and a neighbor point p_k in some given metric space, thus scoring the (p_q, p_k) pair, but could just as well be selected as a different measure if necessary. To understand the importance of this weighting scheme, the figure below presents the influence region diagram for a k-neighborhood set centered at p_q.

These features are then encoded to create a dictionary containing codes for all the trained objects. For each segmented object, a visual representation at object level is acquired by means of the Bag-Of-Words model. We briefly discuss the process.

For training, features extracted for all the captured views of all the objects in the database, K-means clustering is performed over all the vectors to obtain feature codes from the centers of the computed clusters. We set 256 K clusters as the size of our codebook as we found this size to give the effect and better results in comparison to sizes of 128 and 512. A putative visual representation for each object is then obtained by comparing all the features in the codebook generated by finding the minimum Euclidean distance, max pooling, and l2 normalization, resulting in 256 dimensional long vectors for each pose of an object. At the test phase, features extracted for all the objects are encoded using the codebook generated and fed to the classifier.

Classification

In the proposed interactive perception stage, the distribution of predictive probabilities among all object categories is modeled by employing multi-class Gaussian Process Classification [Rasmussen, Sun et al., 2016c].

A brief description of the GP based classification is provided in Chapter 5 and a more detailed description can be found in our previous work [Sun et al., 2016c].

A pre-trained Gaussian process based classifier is used to work on the resultant encoded visual representation for each object from the stage above in order to predict the category of the object along with probability confidences. These confidences are then used to make a decision of the identity of the object. The visual system described in this paper has been tested with two multi-class classifiers, Support Vector Machines and Multi-class Gaussian Process models [Rasmussen].

Semantic visualization

Semantic visualization is an added tool provided with our proposed system to visualize the output of GP multi-class classification. A color-coded visual representation is used for objects via the RVIZ tool available in ROS (http://wiki.ros.org/rviz), where each color corresponds to a category.

6.4.2 Interaction stage

This stage of the autonomous visual architecture applied to the task of objects sorting is responsible for carrying out the manipulation tasks, , such as, pick an object as shown in Figures 6.10 and 6.11, place an object on the table for further investigation at a designated area or hold it in-hand as shown in Figures 6.14 and 6.15, in front of the camera, rotating an object in order to inspect different views, and remove objects from tabletop into designated bin as shown in Figures 6.12 and 6.13. Fig. 6.6 shows the interaction model opted for the task of object sorting.

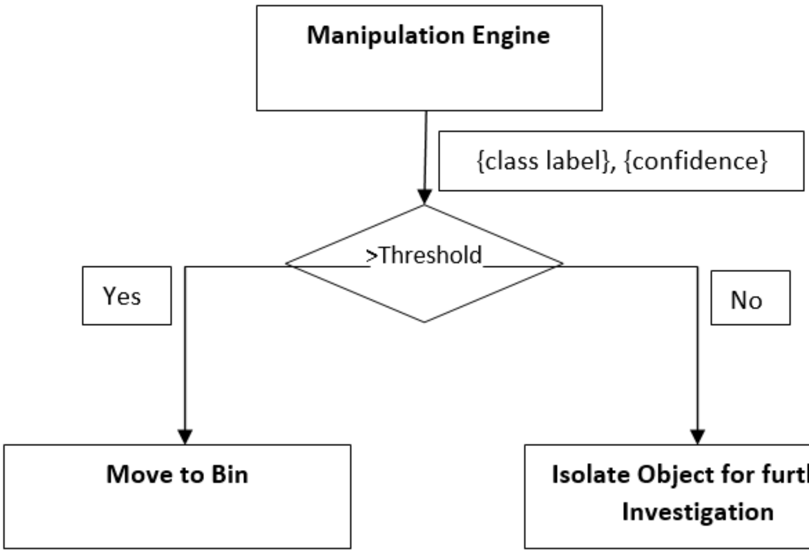

Manipulation Engine
{class label}, {confidence}
>Threshold
Yes
No
Move to Bin
Isolate Object for further Investigation

Figure 6.6: Interaction stage of the proposed autonomous visual architecture for scene understanding, applied to the task of object sorting.

An interaction stage follows the perception stage in the proposed visual perception architecture that orchestrates the set of object manipulation skills available. Two simple manipulation skills/actions are available for the interaction stage to chose from after receiving the required information from the perception stage: 1) An object class has been identified with high confidence, so this object can be picked from the scene (tabtop in this case) and place it in the designated basket/bin for that particular object class. 2) An object class has been identified with less probabilistic confidence, since from observation and inspired from the human behavior of inspecting doubtful objects, an action is chosen to investigate that object more. This means, to confirm the object identity by taking more images from different views in isolation and perception stage perform the visual operation for those views, hence multiple perception-action/interaction cycles take place.

Upon receiving the class label, confidence and location information from the perception model, the interaction stage in the proposed visual architecture, orchestrates the set of object manipulation skills available. Two simple manipulation skills/actions are available for the interaction stage to chose from after receiving the required information from the perception stage. One of these two actions is chosen for each object, from a maintained list of all the objects in the scene along with their classification confidence score. Before an action/skill is selected, each object in the list is checked for whether or not, its confidence is above a threshold. If the confidence score of an observed object is above the threshold, the object is thus removed from the table top and placed into their respective classification bin by invoking action 1. Alternatively, if the confidence score is lower than the threshold, the object is further investigated by picking the object and actively exploring the view-sphere of the object in order to provide feedback to the perception model, thus, invoking action 2. The underlying hypothesis is that by exploring a different view of the object, this new view can increase the classification score and improve the recognition accuracy. Finally, the object is moved into the respective bin.

It must be noticed that the manipulation actions/skill developed and integrated into the proposed visual perception architecture are simple and doesn't provide complex grasping solutions. Because, finding a grasping solution for objects with different shapes is still an open research area and current solutions depend on the availability of the type of end effector i.e suction cups, two parallel fingers etc. Due to these limitations, grasping has not been the focus of investigation while developing the proposed visual architecture.

6.5 Experiments

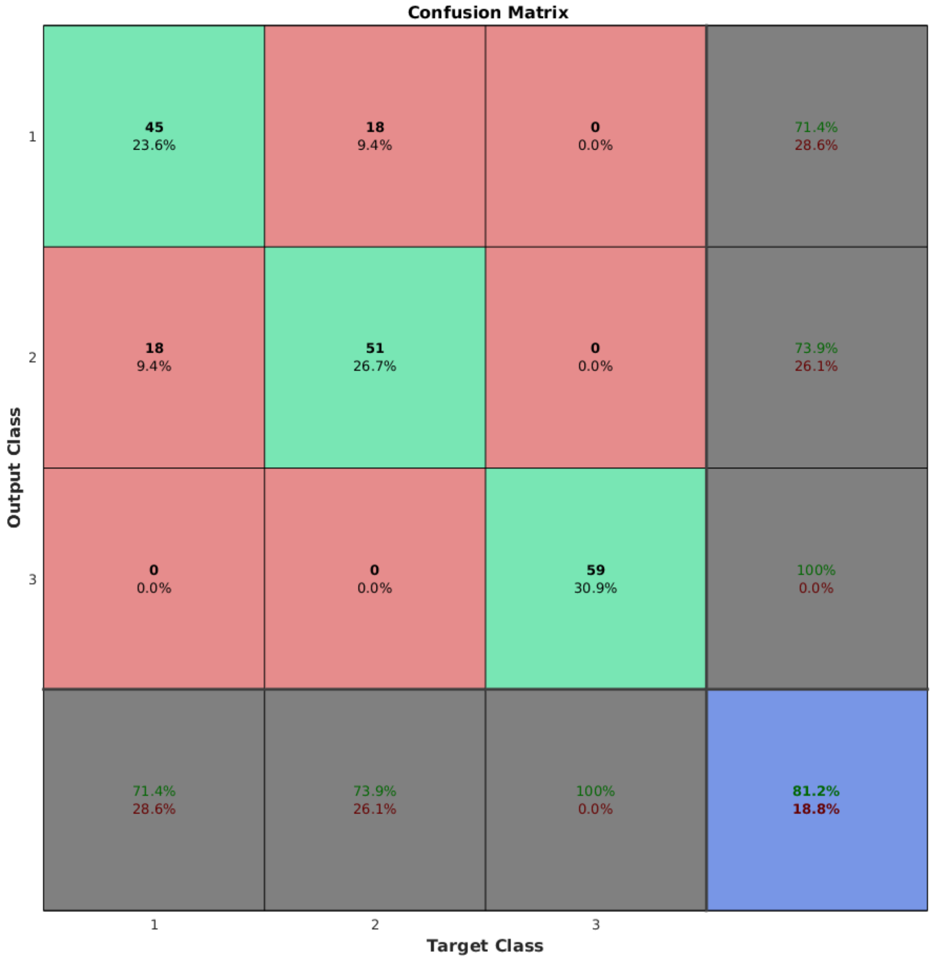
These experiments are designed to demonstrate the validity of the proposed visual architecture incorporating interactive perception. Furthermore, it is integrated into an autonomous robotic system, for scene understanding. The proposed approach is simple and capable of improving the classification accuracy by interacting with objects by applying the approach to the real-world application of carrying out the task of autonomous object sorting. It is pointed out, that the dataset constructed for these experiments contain objects which can be accommodated by both the simple manipulation tasks developed to demonstrate the effectiveness of the proposed approach and the availability of the smart grasping system as an end effector. The developer version of the grasping system has a set of defined grasping actions. For these reasons, the objects chosen to construct the database of objects and construct the test scene contain three household object categories i.e Juice Bottles, Milk Packs, and mugs. These scenes are arranged by combinations of known and unknown object instances of the 3 different object classes used. Objects were placed in arbitrary poses and locations. The dataset is created by capturing point clouds of each object at angular intervals of approximately 20 degrees and it will be made public to test and use.

To show the performance of GP based multi-class classifier in comparison with an SVM multi-class classifier, results are shown from simulated evaluation in Fig. 6.8 and Fig. 6.7, respectively. The diagonal of each confusion matrix, colored in green, shows the true positives and the bottom row gives the recognition rate for each category. Also in Fig. 6.9 illustrates the conditional probability against testing examples for three object classes. Same colored testing examples shown in the Fig. belong to the same class.

For the SVM multi-class classifier, an average recognition rate of 83% approx. and for the GP based multi-class classifier, 92% approx. is obtained. It must be pointed out that the recognition rate is expected to be lower as reported in Chapter 5 for challenging scenes, since the use of a low-cost sensor, in these experiments. That also augments the argument towards interactive perception model where one shot recognition fails.

By taking advantage of the dual arm Yaskawa Motoman robot, one arm equipped with ASUS Xtion pro is designated to capture the data while the other arm is used to fulfill the manipulation tasks needed. This configuration of using the two arms, in contrast to a single arm, eliminates the problem of not having a clear view of the scene and/or objects contained in them, by occlusion caused by the end effector of the same arm.

For arranging the scene, objects are placed directly on the table. An input point cloud is obtained from the RGBD camera from one view. The point cloud is subsequently passed on to our pipeline to trigger the perception model of our proposed system. After obtaining a point cloud, all the objects are segmented, features of each object are extracted and fed to the pre-trained Gaussian process based classifier. The GP based classifier provides both class label for each object and associated prediction confidence scores. After the class labels and related confidences are received, the interaction stage of the proposed system takes over. The interaction stage performs the manipulation tasks. The interaction stage keeps and maintains a list of objects according to their respective confidences. Additionally, it also keeps information about the object location and size for all the detected objects on the table-



Confusion Matrix

Output Class \ Target Class	1	2	3	
1	45 23.6%	18 9.4%	0 0.0%	71.4% 28.6%
2	18 9.4%	51 26.7%	0 0.0%	73.9% 26.1%
3	0 0.0%	0 0.0%	59 30.9%	100% 0.0%
	71.4% 28.6%	73.9% 26.1%	100% 0.0%	81.2% 18.8%

Figure 6.7: Confusion matrix for the multi-class SVM classifier.

top. Experiments carried out after this step to investigate the impact of interactive perception through an application of object sorting task.

For consistency and validity of the proposed visual perception architecture based on GP classification for improving the recognition rate towards the task of object sorting, three working scenarios have been designed;

1. For confidences of objects classified above a pre-defined threshold, the object is removed from the table top directly into bins w.r.t their predicted labels.

2. for confidences of objects classified lower than a pre-defined threshold, before moving them to their bins, are first moved to an area where the object will have none or little occlusion, and capture a few more point clouds for at least 3 views to observe the increase in confidence along with the class label. This area, where further investigation of the object takes place, can't contain any object from the original scene.

3. Repeat 2, but in this case, instead of placing each object back on the table, the observation of different poses and capture of point cloud from each pose will be made in hand of the robot after picking each object. The rationale for this scenario is for better and generalized

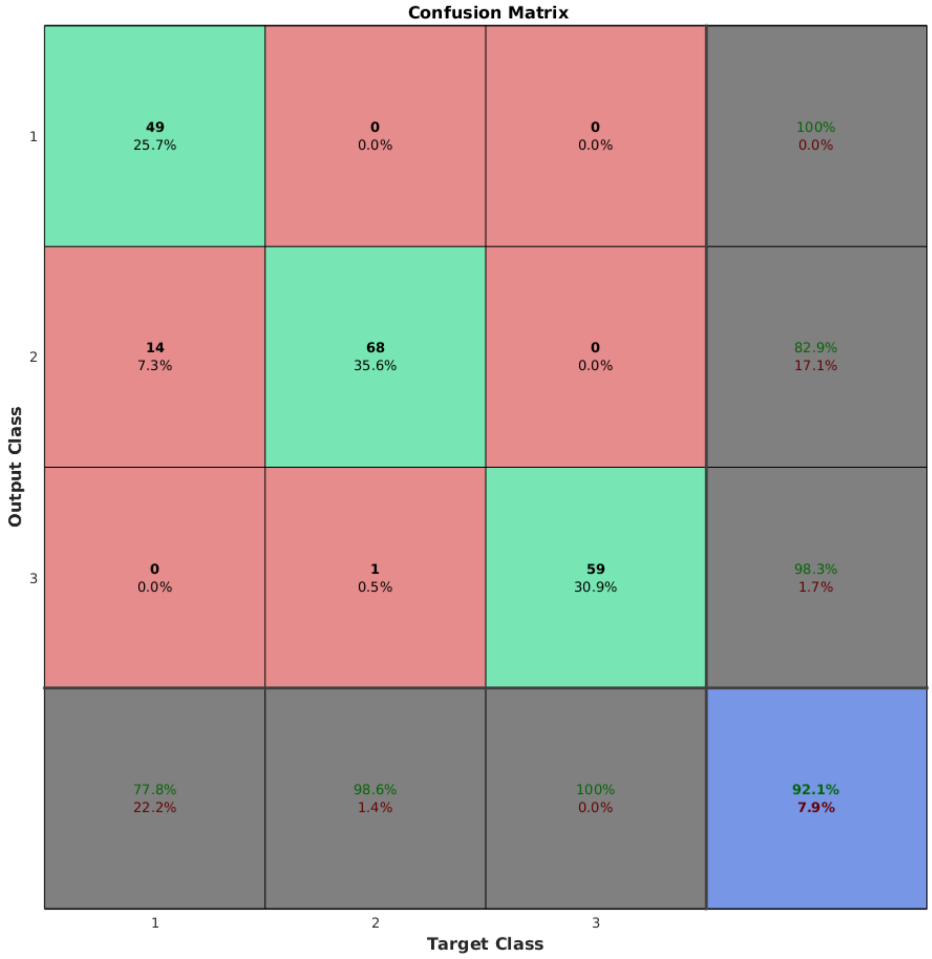


Figure 6.8: Confusion matrix for multi-class gp classifier for 5 object categories.

segmentation results as compared to putting it back on the table. In hand, observation is imitation from the observation of what human at early ages do, both for verification and learning of objects.

After all the objects have successfully been recognized and classified accordingly, objects are placed into their respective bin by Yaskawa Motoman robot arm.

Note: The experimental design to validate the proposed approach for the task of autonomous objects sorting is implemented and integrated into Yaskawa Motoman Robot. Unfortunately, the end effector i.e the smart grasping system ran into hardware issues and and wasn't available until the presentation of this thesis.

6.6 Conclusion

This chapter described a simple and generic visual architecture for scene understanding by enabling an autonomous system to interact with the scene to achieve a substantial increase in

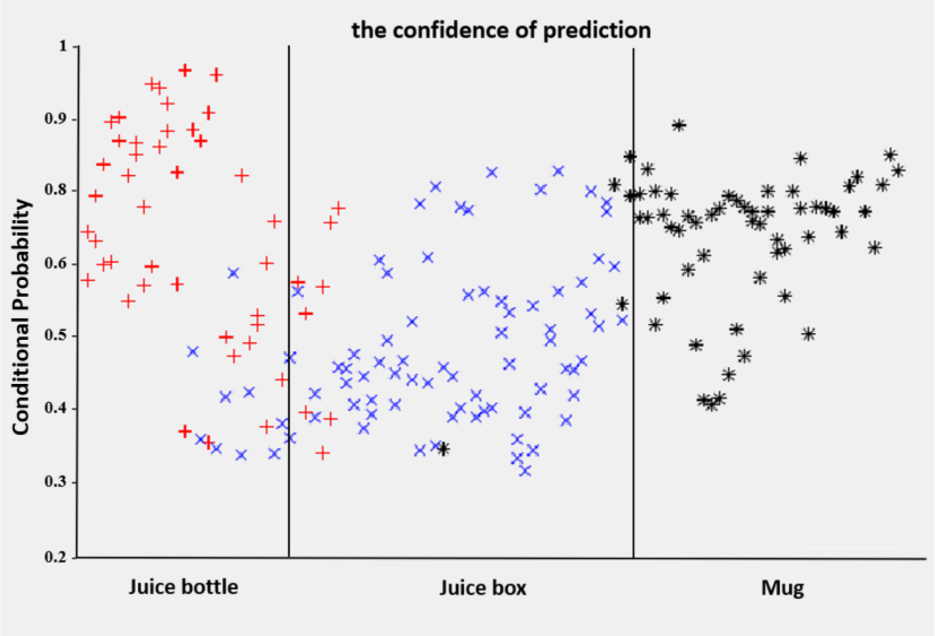


Figure 6.9: The classification performance under different confidences for 3 object categories.

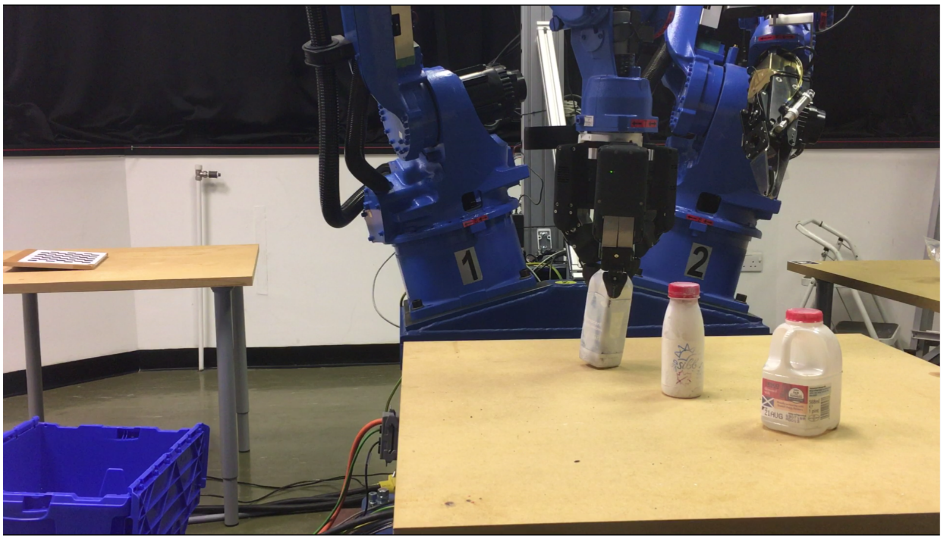


Figure 6.10: Interaction stage of the proposed autonomous visual architecture for scene understanding, while picking a juice bottle.

scene understanding over non-interactive approaches through an application of autonomous objects sorting. Further, the proposed approach is applied to the task of autonomous objects sorting, incorporating depth sensing, 3D feature extraction, object representation, and classification. The proposed visual architecture achieves near real-time object category identification with associated perception confidence estimates, which serves to establish whether an observation is sufficient to make a classification or to determine if it is necessary to interact with the object to achieve a sufficiently reliable classification of the instance contained in

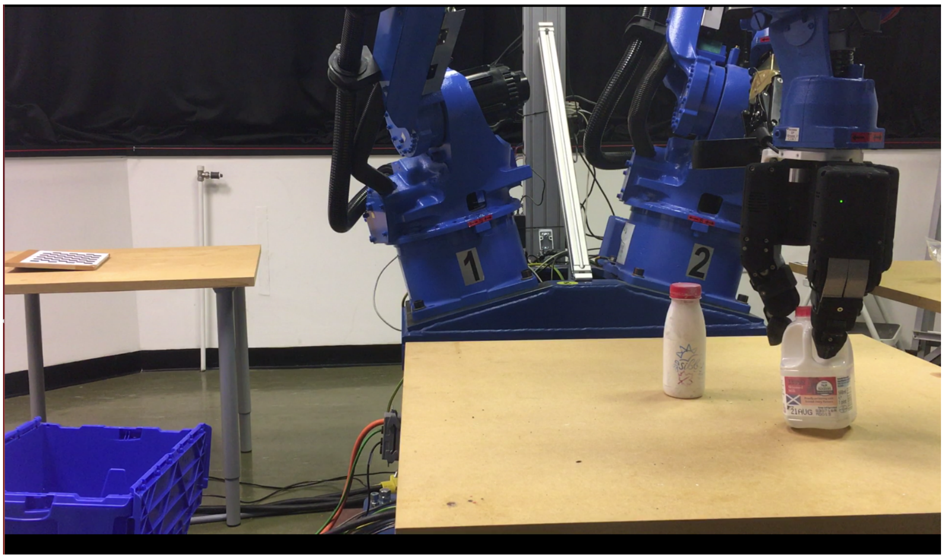


Figure 6.11: Interaction stage of the proposed autonomous visual architecture for scene understanding, while picking a milk-box.

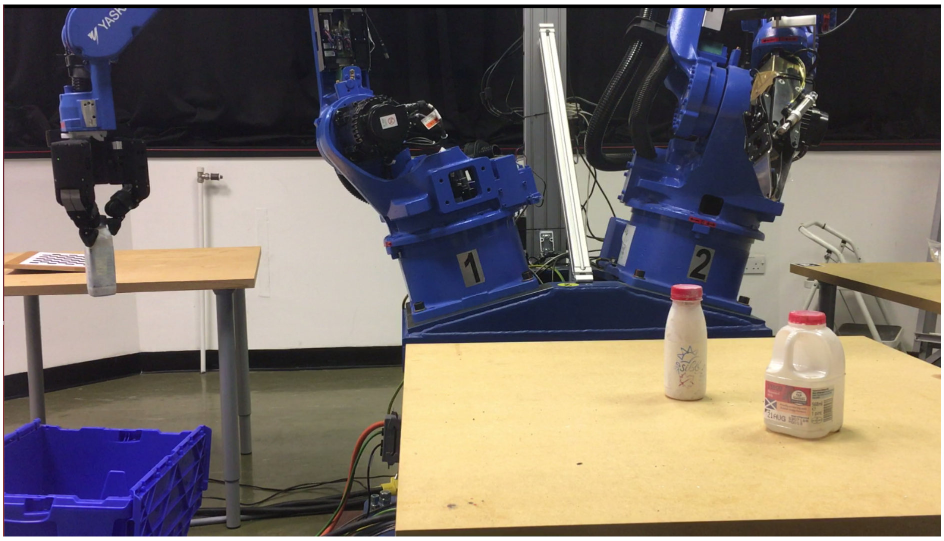


Figure 6.12: Interaction stage of the proposed autonomous visual architecture for scene understanding, while placing a juice bottle.

the observation.

During the development of the proposed visual architecture, interesting lessons have been learned from the contributions, limitations have been realized and provide insight into future work. These are described in the penultimate Chapter, Chapter 7.

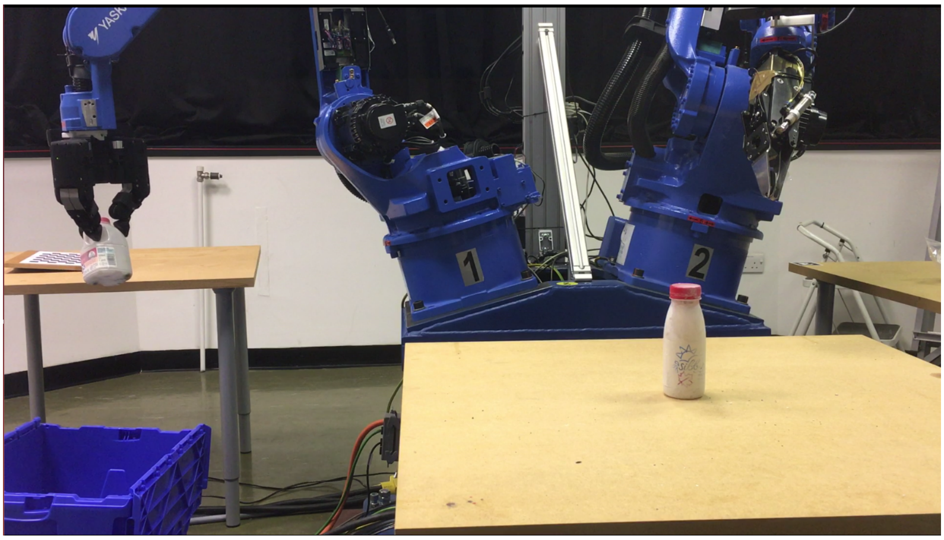


Figure 6.13: Interaction stage of the proposed autonomous visual architecture for scene understanding, while placing a milk-box.

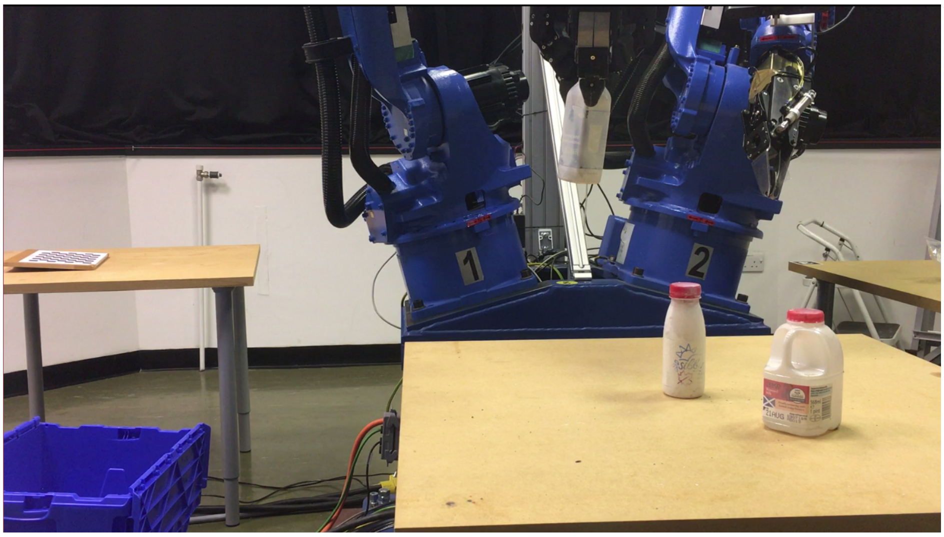


Figure 6.14: Interaction stage of the proposed autonomous visual architecture for scene understanding, while investigating the juice bottle more by rotating the juice bottle and taking images.

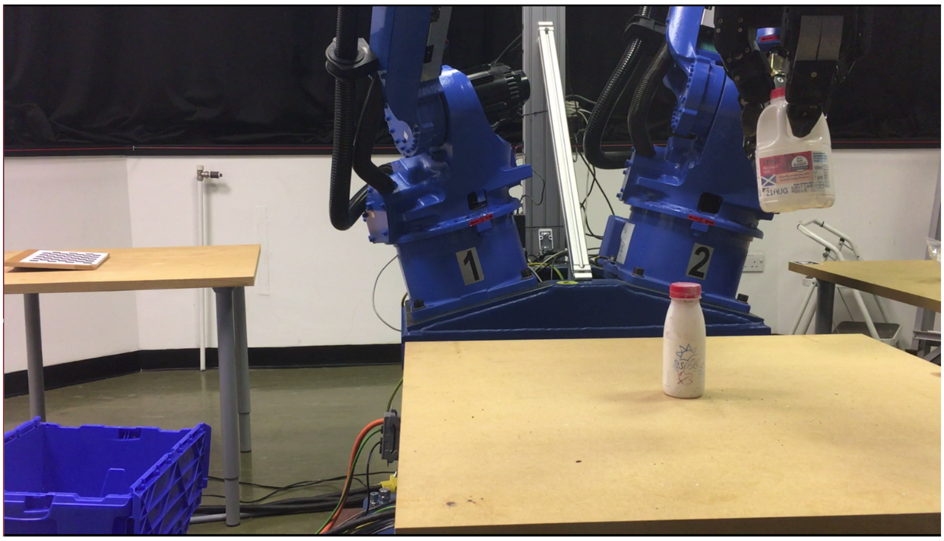


Figure 6.15: Interaction stage of the proposed autonomous visual architecture for scene understanding, while investigating milk-box more by rotating the milk-box and taking images.

Chapter 7

Conclusion

This chapter finalizes the thesis by presenting the contributions made during the course of the thesis. It then states the hypothesis validation. Limitations in the current visual perception architecture of scene understanding by robotic manipulation are discussed. Future work suggestions are then detailed while to extend the proposed visual architecture for scene understanding by robotic interaction.

7.1 Summary of Contributions

This section contains the summary of contributions during the course of producing this thesis in four parts in chronological order: Contribution to active binocular robot vision architecture [Aragon-Camarasa et al., 2010], Camera-Hand Eye Calibration, Visual perception model for scene understanding by interactive perception, an autonomous visual perception architecture capable of scene understanding by interactive perception. These are discussed in the following section.

7.1.1 A Portable Active Binocular Robot Vision Architecture

This thesis presented a successful demonstration of portable active binocular robot head that integrates visual behaviors in a unified and parsimonious architecture that is capable of autonomous scene exploration. This robot architecture can identify and localise multiple same-class and different-class object instances while maintaining vergence and directing the system's gaze towards scene regions and objects.

In terms of functional accuracy and performance of the portable active vision architecture, it has been validated over challenging scenes and realistic scenarios in order to investigate and study the performance of the visual behaviours as an integrated architecture. By carrying out a qualitative comparison with current robot vision systems whose performance has been reported in the literature, it is argued that our architecture clearly advances the reported state-of-the-art [Aragon-Camarasa et al., 2010, Arbib et al., 2008, Ma et al., 2011, Meger et al., 2010, Rasolzadeh et al., 2010] in terms of our system's innate visual capabilities and portability to different environment settings, e.g. multiple same-class object identification and tolerated degree of visual scene complexity. In addition, this architecture is therefore portable in order to be adapted to different hardware configuration, feature description, and view-points.

7.1.2 Calibration of Active Binocular and RGBD Vision Systems

This thesis also made a contribution by proposing a calibration methodology to calibrate the portable active binocular robot head system integrated within an industrial dual-arm Yakasawa robot and described a comparative experiment on the integration of RGB cameras into the Rethink Baxter robot. Moreover, a simple, yet effective solution to dynamically update cameras' extrinsic parameters in order to achieve geometric compatibility with respect to the robot's kinematic chain has been reported. By comparing our binocular robot head with consumer RGBD cameras, it can be concluded that our robot head provides an off-the-self depth sensing solution capable of reconstructing the observed 3D scene within sub millimetre accuracy. Likewise, it is shown that the implemented calibration routines in this chapter can provide reliable results that allow reconstruction with relative 3-D errors of less than 0.3 millimetres. An open-source ROS package that implements our calibration methods can be found at `https://github.com/gerac83/glasgow_calibration`. A video demonstration of the calibration of our active binocular robot head can be accessed at: `https://youtu.be/9OYy9Q_bN2w`.

7.1.3 Interactive Perception Model Based on Gaussian Process Classification

This thesis makes contribution by proposing a visual interactive perception system achieving real-time object category identification/recognition with associated classification confidence (probabilistic) estimates, which serves to establish whether an observation is sufficient to make a classification or to determine if it is necessary to interact with the object to achieve a sufficiently reliable classification of the instance contained in the observation.

7.1.4 Objects Manipulation in Scene Understanding By Interactive Perception

Extending the visual perception model by interactive perception from Chapter 5, the proposed architecture is made fully autonomous and integrated into dual arm robot test-bed. An application of autonomous objects sorting is demonstrated by applying the proposed visual perception architecture. The whole architecture is two-fold. On one hand, the visual perception stage, handles vision related behaviours such as, object segmentation, identification, classification with classification confidence (probabilistic). On the other hand, the interaction stage takes one of the two actions. One, if the object's predictive confidence besides class label is higher than a predefined threshold, the interaction stage carries the action of picking up the object and placing it in the class designated bin. Second, if the object's predictive confidence besides class label is lower than a predefined threshold, the interaction stage chooses the action, that picks up the object and investigate it by rotating with an angle 30 degrees and capture three more distinct views. For each new capture image, the perception stage is invoked in foveated mode, where the image of only the object in hand is captured. This multiple perception-action cycles continue and at the end of three captures, the maximum number of class prediction along with higher probabilities is taken as the true class and at this stage, action 1 is executed.

7.2 Revisiting Thesis Statement

This thesis advances the state-of-the-art in scene understanding by robotic interactive perception compared to non-interactive approaches. On one hand, an autonomous system equipped with an active vision system, more specifically, the portable active vision system (Chapter 3), can explore a scene in detail. By following the underlying principle of structuring visual perception as what and where systems, whose joint interaction affords the necessary visual understanding required to conduct robotic hand-eye grasping and manipulation tasks. For these reasons, experimental validation of the portability, functional accuracy of the portable active binocular robot vision system is presented in Chapter 3. From results obtained and discussed in Chapter 3, it can be deduced that by actively exploring a scene for scene understanding, provides an improved solution over non-interactive systems. Additionally, a camera-hand eye calibration approach is also devised and detailed in Chapter 4, to integrate the vision system into robotic's kinematic frame's chain. Which enables the robot to perform manipulation tasks supported by visual stimuli.

On the other hand, an autonomous visual architecture is proposed for scene understanding, from the perspective of interacting with the object in the scene by repositioning it, in order to

obtain new, potentially more diagnostic, views of it while performing the scene understanding task.
This proposed architecture has been implemented and integrated into a dual-arm robotic testbed. Furthermore, the proposed approach has been demonstrated by performing the task of house-hold objects sorting. This proposed approach consists of two stages namely: Perception stage and Interaction stage. The perception stage performs visual tasks while the interaction stage follows the perception stage in the proposed visual perception architecture that is responsible for object manipulation skills. Chapter 5 and Chapter 6 present a visual architecture integrated with robotic manipulation skills performing object repositioning in the scene, that improves the classification accuracy compared to non-interactive perception models. Additionally Chapter 5 and Chapter 6 verifies the base capability of the proposed visual perception architecture integrated into fully autonomous robotic system, that demonstrates, that by performing objects sorting of house-hold objects that requires understanding of what (house-hold object) is in the scene and what it can do with it (grasp & reposition the object).

7.3 Limitations and Future Work

This section discusses the limitations encountered in the course of producing this thesis and ideas for future improvements are suggested.

7.3.1 Active Vision System

In biological systems, it is found that a region in the scene that is sufficiently salient can capture the attention of an observer more than once during a visual task [Wolfe and Whitney, 2015, Wurtz et al., 2011]. Our current inhibition of return behaviour, however, has been formulated explicitly to prevent the robot from visiting a previously attended location. We propose to revise this behaviour by incorporating an exponential decay criterion that dictates the mean-lifetime of inhibition of an attended location. The robot would then be able to re-visit a previously attended location, perhaps in the context of a spatial awareness model with a cognitive module.

Another interesting research question that needs to be investigated is: while performing the scene exploration task, what areas of interest in the scene need to explored and in what order? Here, underline principle is to reduce the number of saccades for the active vision system and perform visual search task efficiently.

Another possible extenstion to exisiting active vision system can be substituting the macro script with a cognitive/intelligent layer, the sequence of behaviours required to convey a visual task can be generated deliberatively, thereby, removing the fixed-task limitation of the current control scheme. Accordingly, the architecture discussed in this thesis has been designed such that a deliberative/cognitive module might replace the fixed script in future modifications of the robot system without altering the underlying visual behaviours.

7.3.2 Improving Quality of Depth Images Generated by the Active Vision System

As mentioned in section 2.4.3, the resident stereo matcher for active vision system is inclined to work on deformable objects better where the boundaries are continuous and produce good quality depth images for such objects. It is therefore interesting to improve the stereo matcher for the active vision system to handle the discontinuities at the boundaries of rigid objects and produce good quality depth images which has been a desirable commodity in need in the robotic vision community.

7.3.3 Camera-Hand Eye Calibration

Currently, the investigation aims at as how to maintain dynamic intrinsic camera calibration (i.e. change of focus) to enable the robot head to focus different depth of fields. There is a need to validate the extrinsic dynamic calibration in terms of drift errors and accuracy of the computed point clouds.

7.3.4 Visual Perception Model for Interactive Perception

This thesis presented a visual perception architecture for scene understanding by robotic interactive perception, by successfully demonstrating the application of sorting house-hold objects into categories. With our own limited created database and for the scope of this thesis, the visual architecture proposed for proof of concept in this thesis suffice. However, recently, as the dataset of variety of objects has been created at the resident lab, for the proposed architecture to work with large number of objects, incorporating deep learning techniques for better and accurate object recognition - both for rigid and non-rigid objects, better localization and pose estimation, would add scalability and reliability into the architecture.

Also, the proposed architecture struggles with scenes made up in a pile configuration. The current technique of segmentation of the scene into possible objects fails. Replacing current segmentation technique with more robust segmentation method such as [Badrinarayanan et al., 2017], and integrating into the architecture would extend the use of the work presented in this thesis to wider applications.

Finally, the proposed visual architecture only takes into account the depth image and extract diagnostic visual cues from it. Combining the visual cues both from RGB and Depth images would add both reliability and robustness to the scene understanding which require reasoning (what to see and what the robot can do with it) as compared to just producing classification label according to category. Fig. 7.1 illustrates recent work by [Sun et al., 2017b], that provides an example of incorporating different visual cues.

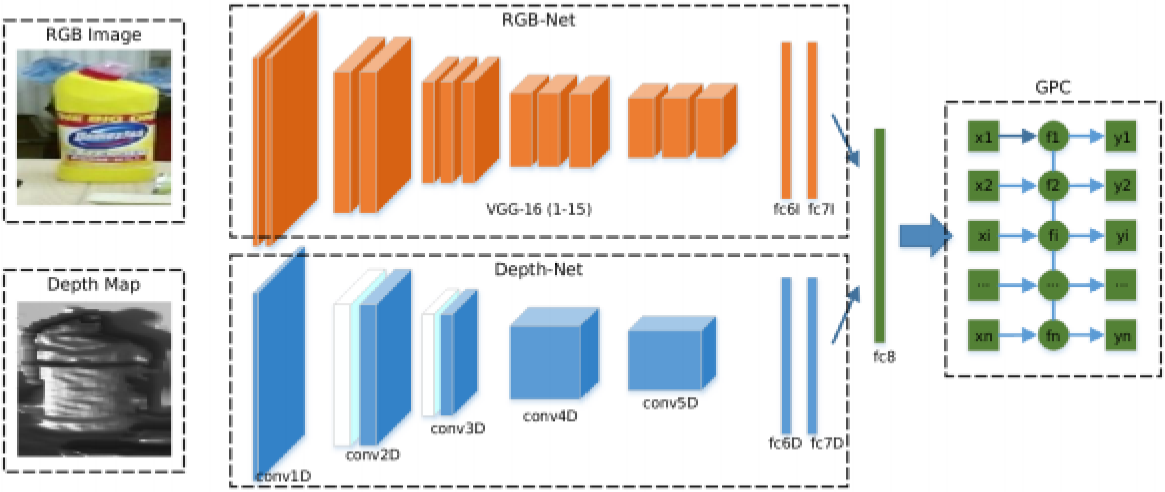


Figure 7.1: The architecture of proposed multi-modal DCNN-GPC. The inputs of the DCNNs are the raw RGB image and depth map of the object proposal. The architecture consists of three components: RGB-Net (shown in yellow) Depth-Net (shown in Blue) and non-parametric GPC (shown in Green).

7.3.5 Objects Manipulation in Scene Understanding By Interactive Perception

Despite the fact that grasping strategy for objects manipulation is not the focus of the work presented in this thesis, it is still worth discussing the effect of grasping on scene understanding involving robotic interactive perception. The interaction stage, in the proposed visual perception architecture is made of simple actions (details in Chapter 6). More specifically, after the information received from the perception stage, the object chosen for manipulation is always picked in the same way. The object is picked at a right angle, at most 3 cms deep, so that if the object needs to be investigated further, still should have a visible part of it to have a diagnostic view. This poses a restriction on the number of objects with different shapes to be handled by the proposed approach as failed grasps are inevitable for objects at interaction stage, which are short in height. With the availability of the large dataset, techniques reported such as [Levine et al., 0], developed a deep learning architecture that learns from a large convolutional neural network to predict the probability of quality grasp solutions as shown in Fig. 7.2.

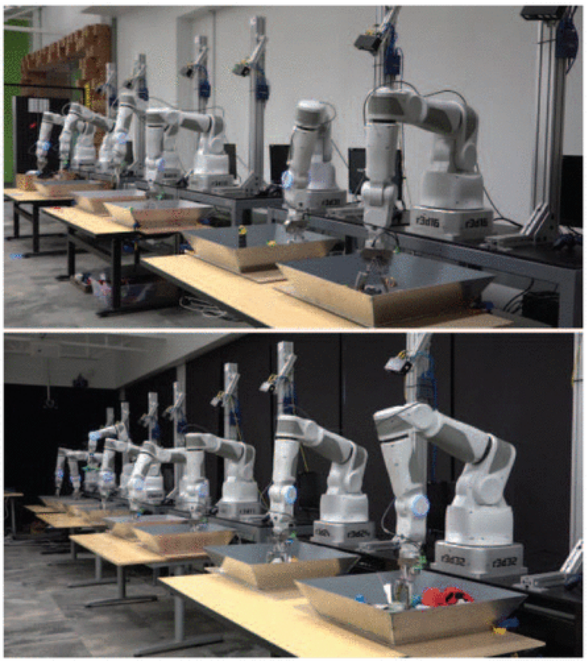


Figure 7.2: The large-scale data collection setup for the first set of experiments, consisting of 14 robotic manipulators. They collected over 800,000 grasp attempts to train the CNN grasp prediction model..

Another notable example is from [Mahler et al., 2017a,b], who have developed benchmark datasets (i.e. Dexnet 1.0, 2.0 and 3.0) where grasping candidates are learned using annotated datasets consisting of 150 3D models of which 1M rendered poses are extracted. Techniques like these are good motivation for incorporating better grasping strategies into the proposed visual architecture and in turn, can extend the affect of this architecture on a wider range of applications.

7.3.6 Objects Appearance Learning By Interactive Perception

Building on the premises of previous sections 7.3.4 & 7.3.5, the visual perception architecture could be extended with the capabilities of learning the appearance of unknown objects by modifying the in hand object manipulation skill described in Chapter 7. Both recognition task and learning the appearance of an unknown object, an important research question is what pose of the object is most useful to be identified with. [Camarasa, 2012] validated that in terms of underlying properties from which learned canonical representations must be composed of, namely, the goodness of recognition, familiarity, and functionality (according to [Blanz et al., 1999] and [Ullman et al., 2002]).

7.3.7 Investigation into Interaction with Object in Scene Strategy

To improve recognition accuracy in scene understanding by robotic interactive perception, it is vital to obtain information that contains the knowledge of which pose of an object is diagnostic to its identity. A possible solution is to devise a technique which at the interaction stage, finds poses of objects in the scene that have been identified diagnostic to their respective identities at the training/learning stage.

Further extension may include, more advanced deep learning based object detection methods such as [Sun et al., 2017c] and further build a 3D semantic map [Sun et al., 2018, Zhao et al., 2017a,b]. Moreover, an investigatation can be carried out into the possibility to adapt the manipulation system to a mobile platform [Sun et al., 2017a, Yan et al., 2018, Zaganidis et al., 2018, Zhao et al., 2018].

Bibliography

Alexandre Alahi, Raphael Ortiz, and Pierre Vandergheynst. Freak: Fast retina keypoint. *Computer Vision and Pattern Recognition (CVPR)*, 2012, 2012.

P. F. Alcantarilla, A. Bartoli, and A. J. Davison. KAZE features. In *Eur. Conf. on Computer Vision (ECCV)*, 2012.

John Aloimonos, Isaac Weiss, and Amit Bandopadhay. Active vision. *International Journal of Computer Vision*, 1(4):333–356, January 1988. ISSN 0920-5691. doi: 10.1007/BF00133571.

G. Aragon-Camarasa and J. P. Siebert. A hierarchy of visual behaviours in an active binocular robot. In Theocharis Kyriacou, Ulrich Nehmzow, Chris Melhuish, and Mark Witkowski, editors, *Towards Autonomous Robotic Systems, TAROS 2009*, pages 88–95, 2009.

Gerardo Aragon-Camarasa and J Paul Siebert. Unsupervised clustering in Hough space for recognition of multiple instances of the same object in a cluttered scene. *Pattern Recognition Letters*, 31(11):1274–1284, August 2010. ISSN 01678655. doi: 10.1016/j.patrec.2010.03.003.

Gerardo Aragon-Camarasa, Haitham Fattah, and J Paul Siebert. Towards a unified visual framework in a binocular active robot vision system. *Robotics and Autonomous Systems*, 58(3):276–286, March 2010. ISSN 09218890. doi: 10.1016/j.robot.2009.08.005.

Gerardo Aragon-Camarasa, Haitham Fattah, and J. Paul Siebert. Towards a unified visual framework in a binocular active robot vision system. *Robotics and Autonomous Systems*, 58:276–286, 2010.

M Arbib, G Metta, and P der Smagt. Neurorobotics: From vision to action. In Bruno Siciliano and Oussama Khatib, editors, *Springer Handbook of Robotics*, pages 1453–1480. Springer Berlin Heidelberg, 2008. ISBN 978-3-540-30301-5. doi: 10.1007/978-3-540-30301-5_63.

A. Aydemir and P. Jensfelt. Exploiting and modeling local 3d structure for predicting object locations. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 3885–3892, Oct 2012. doi: 10.1109/IROS.2012.6386111.

Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

D H Ballard. Animate vision. *Artificial Intelligence*, 48(1):57–86, 1991. ISSN 00043702. doi: 10.1016/0004-3702(91)90080-4.

P. Bariya and K. Nishino. Scale-hierarchical 3d object recognition in cluttered scenes. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1657–1664, June 2010. doi: 10.1109/CVPR.2010.5539774.

Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded up robust features(surf). *Computer Vision and Image Understanding*, 110(3):346–359, June 2008a.

Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Comput. Vis. Image Underst.*, 110(3):346–359, June 2008b. ISSN 1077-3142. doi: 10.1016/j.cviu.2007.09.014. URL `http://dx.doi.org/10.1016/j.cviu.2007.09.014`.

Paul J Besl and Neil D McKay. Method for registration of 3-d shapes. In *Robotics-DL tentative*, pages 586–606. International Society for Optics and Photonics, 1992.

Volker Blanz, Michael J Tarr, and Heinrich H Blthoff. What object attributes determine canonical views? *Perception*, 28(5):575–599, 1999. doi: 10.1068/p2897. URL `https://doi.org/10.1068/p2897`. PMID: 10664755.

Marcel Brückner, Ferid Bajramovic, and Joachim Denzler. Intrinsic and extrinsic active self-calibration of multi-camera systems. *Machine Vision and Applications*, 25(2):389–403, 2014. ISSN 1432-1769. doi: 10.1007/s00138-013-0541-x. URL `http://dx.doi.org/10.1007/s00138-013-0541-x`.

Michael Calonder et al. *Brief: Binary robust independent elementary features*. Springer, Berlin, 2010.

G. Aragon Camarasa. A hierarchical active binocular robot vision architecture for scene exploration and object appearance learning. *phd thesis*, 2012. URL `http://theses.gla.ac.uk/id/eprint/3640`.

John Canny. A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 8:6, November 1986.

S Chen, Y Li, and Ngai M Kwok. Active vision in robotic systems: A survey of recent developments. *International Journal of Robotics Research*, 30(11), 2011. doi: 10.1177/0278364911410755.

M. M. Chun and J. M. Wolfe. Visual attention. In E. B Goldstein, editor, *The Blackwell Handbook of Perception*, pages 272–310. Blackwell Publishers Ltd, Ch. 9, 2004.

Dan Ciresan, Ueli Meier, and Jürgen Schmidhuber. Multi-column deep neural networks for image classification. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3642–3649. IEEE, 2012.

Paul Cockshott, Susanne Oehler, Tian Xu, Paul Siebert, and Gerardo Aragon-Camarasa. A parallel stereo vision algorithm. In *Many-Core Applications Research Community Symposium 2012*, 2012.

Alvaro Collet, Manuel Martinez, and Siddhartha S. Srinivasa. The moped framework: Object recognition and pose estimation for manipulation. *The International Journal of Robotics Research*, 2011.

Thomas M Cover and Peter E Hart. Nearest neighbor pattern classification. *Information Theory, IEEE Transactions on*, 13(1):21–27, 1967.

N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In Computer Vision and Pattern Recognition, editors, *CVPR 2005*, pages 25–25, vol.1, no., pp.886,893 vol. 1, June 2005a. IEEE Computer Society Conference on.

Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*, CVPR '05, pages 886–893, Washington, DC, USA, 2005b. IEEE Computer Society. ISBN 0-7695-2372-2. doi: 10.1109/CVPR.2005.177. URL `http://dx.doi.org/10.1109/CVPR.2005.177`.

T. Dang, C. Hoffmann, and C. Stiller. Continuous stereo self-calibration by camera parameter tracking. *IEEE Transactions on Image Processing*, 18(7):1536–1550, July 2009. ISSN 1057-7149. doi: 10.1109/TIP.2009.2017824.

A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse. Monoslam: Real-time single camera slam. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):1052–1067, June 2007. ISSN 0162-8828. doi: 10.1109/TPAMI.2007.1049.

Andrew J Davison. Real-time simultaneous localisation and mapping with a single camera. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 1403–1410. IEEE, 2003.

Yining Deng and BS Manjunath. Unsupervised segmentation of color-texture regions in images and video. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23 (8):800–810, 2001.

MehmetR. Dogar, MichaelC. Koval, Abhijeet Tallavajhula, and SiddharthaS. Srinivasa. Object search by manipulation. *Autonomous Robots*, 36(1-2):153–167, 2014. ISSN 0929-5593. doi: 10.1007/s10514-013-9372-x.

Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. *arXiv preprint arXiv:1310.1531*, 2013.

A. Doumanoglou, A. Kargakos, Tae-Kyun Kim, and S. Malassiotis. Autonomous active recognition and unfolding of clothes using random decision forests and probabilistic planning. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 987–993, May 2014a. doi: 10.1109/ICRA.2014.6906974.

Andreas Doumanoglou, Tae-Kyun Kim, Xiaowei Zhao, and Sotiris Malassiotis. Active random forests: An application to autonomous unfolding of clothes. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision ECCV 2014*, volume 8693 of *Lecture Notes in Computer Science*, pages 644–658. Springer International Publishing, 2014b. ISBN 978-3-319-10601-4. doi: 10.1007/978-3-319-10602-1_42. URL `http://dx.doi.org/10.1007/978-3-319-10602-1_42`.

Jakob Engel, Thomas Schöps, and Daniel Cremers. Lsd-slam: Large-scale direct monocular slam. In *Computer Vision–ECCV 2014*, pages 834–849. Springer, 2014.

L. Ferraz, X. Binefa, and F. Moreno-Noguer. Very fast solution to the pnp problem with algebraic outlier rejection. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 501–508, 2014.

J. M. Findlay and I. D. Gilchrist. *Visual attention: the active vision perspective*. Springer Verlag, Ch. Vision and Attention, pp. 85–106. 2.1.1, 2.4.1, 2001.

Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *ACM Com*, 1981.

P. Fitzpatrick. First contact: an active vision approach to segmentation. In Ieee/rsj International, editor, *Conference on Intelligent Robots and Systems, 2003*. IROS 2003). Proceedings, 2003.

Jerome H Friedman, Jon Louis Bentley, and Raphael Ari Finkel. An algorithm for finding best matches in logarithmic expected time. *ACM Transactions on Mathematical Software (TOMS)*, 3(3):209–226, 1977.

Yasutaka Furukawa and Jean Ponce. Accurate camera calibration from multi-view stereo and bundle adjustment. *International Journal of Computer Vision*, 84(3):257–268, 2009. ISSN 1573-1405. doi: 10.1007/s11263-009-0232-2. URL `http://dx.doi.org/10.1007/s11263-009-0232-2`.

Guglielmo Gemignani, Roberto Capobianco, Emanuele Bastianelli, Domenico Daniele Bloisi, Luca Iocchi, and Daniele Nardi. Living with robots: Interactive environmental knowledge acquisition. *Robotics and Autonomous Systems*, 78:1 – 16, 2016. ISSN 0921-8890. doi: https://doi.org/10.1016/j.robot.2015.11.001. URL `http://www.sciencedirect.com/science/article/pii/S0921889015002468`.

J. J. Gibson. *The Ecological Approach to Visual Perception*. Houghton Mifflin, Boston, MA, USA, 1979.

J. J. Gibson. *The senses considered as perceptual systems*. Houghton Mifflin, Boston, 1966.

Benjamin F Gregorski, Bernd Hamann, and Kenneth I Joy. Triangulation. In *Proc. ARPA Image Understanding Workshop*, pages 957–966, 1994.

M. Gupta, J. Muller, and Ieee G. S. Sukhatme. Transactions on automation science and engineering. *Using Manipulation Primitives for Object Sorting in Cluttered Environments*, pages 608–614, 2015.

P. Hansen, H. Alismail, P. Rander, and B. Browning. Online continuous stereo extrinsic parameter estimation. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1059–1066, June 2012. doi: 10.1109/CVPR.2012.6247784.

C. Harris and M. Stephens. A combined corner and edge detector. In *Proceedings of the 4th Alvey Vision Conference*, pages 147–151, 1988.

R. Held and A. Hein. Movement-produced stimulation in the development of visually guided behavior. *Journal of Comparative and Physiological Psychology*, pages 872–876, 1963.

Tin Kam Ho. The random subspace method for constructing decision forests. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(8):832–844, 1998.

B. K. P. Horn. *Shape from Shading: A Methodfor Obtaining the Shape of a Smooth Opaque Objectfrom One View*. PhD thesis, Massachusetts Institute Of Technology, Cambridge, MA, 1970.

B. K. P. Horn and M. Brooks. *Shape from Shading*. MIT Press, Cambridge, MA, 1989.

Shahram Izadi, David Kim, Otmar Hilliges, David Molyneaux, Richard Newcombe, Pushmeet Kohli, Jamie Shotton, Steve Hodges, Dustin Freeman, Andrew Davison, et al.

Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 559–568. ACM, 2011.

Eric Jang, Sudheendra Vijaynarasimhan, Peter Pastor, Julian Ibarz, and Sergey Levine. End-to-end learning of semantic grasping. preprint, july, 2017.

Kevin Jarrett, Koray Kavukcuoglu, Marc'Aurelio Ranzato, and Yann LeCun. What is the best multi-stage architecture for object recognition? In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 2146–2153. IEEE, 2009.

Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the ACM International Conference on Multimedia*, pages 675–678. ACM, 2014.

A.E. Johnson and M. Hebert. Using spin images for efficient object recognition in cluttered 3d scenes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(5):433–449, May 1999. ISSN 0162-8828. doi: 10.1109/34.765655.

Krishnanand N. Kaipa, Akshaya S. Kankanhalli-Nagendra, Nithyananda B. Kumbla, Shaurya Shriyam, Srudeep Somnaath Thevendria-Karthic, Jeremy A. Marvel, and Satyandr K. Gupta. *Addressing perception uncertainty induced failure modes in robotic bin-picking*. Robotics and Computer-Integrated Manufacturing, 2016.

Branko Karan. Calibration of kinect-type rgb-d sensors for robotic applications. *FME Transactions*, 43(1):47–54, 2015.

D. Katz and O. Brock. Manipulating articulated objects with interactive perception. In Ieee International, editor, *Conference on Robotics and Automation, 2008*. ICRA, 2008.

B. Kehoe, A. Matsukawa, S. Candido, J. Kuffner, and K. Goldberg. Cloud-based robot grasping with the google object recognition engine. In *2013 IEEE International Conference on Robotics and Automation*, pages 4263–4270, May 2013. doi: 10.1109/ICRA.2013.6631180.

A. Khan, G. Aragon-Camarasa, L. Sun, and J. P. Siebert. On the calibration of active binocular and rgbd vision systems for dual-arm robots. In *2016 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 1960–1965, Dec 2016a. doi: 10.1109/ROBIO.2016.7866616.

A. Khan, L. Sun, G. Aragon-Camarasa, and J. P. Siebert. Interactive perception based on gaussian process classification for house-hold objects recognition sorting. In *2016 IEEE*

International Conference on Robotics and Biomimetics (ROBIO), pages 1087–1092, Dec 2016b. doi: 10.1109/ROBIO.2016.7866470.

Aamir Khan and Farooq Hasan. Principal component analysis-linear discriminant analysis feature extractor for pattern recognition. *arXiv preprint arXiv:1202.1177*, 2011.

Aamir Khan, Gerardo Aragon-Camarasa, and Jan Paul Siebert. *A Portable Active Binocular Robot Vision Architecture for Scene Exploration*, pages 214–225. Springer International Publishing, Cham, 2016c. ISBN 978-3-319-40379-3. doi: 10.1007/978-3-319-40379-3_22. URL `http://dx.doi.org/10.1007/978-3-319-40379-3_22`.

Kourosh Khoshelham and Sander Oude Elberink. Accuracy and resolution of kinect depth data for indoor mapping applications. *Sensors*, 12(2):1437, 2012. ISSN 1424-8220. doi: 10.3390/s120201437. URL `http://www.mdpi.com/1424-8220/12/2/1437`.

Georg Klein and David Murray. Parallel tracking and mapping for small ar workspaces. In *Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, ISMAR '07, pages 1–10, Washington, DC, USA, 2007. IEEE Computer Society. ISBN 978-1-4244-1749-0. doi: 10.1109/ISMAR.2007.4538852. URL `http://dx.doi.org/10.1109/ISMAR.2007.4538852`.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

H. Kwon, J. Park, and A. C. Kak. A new approach for active stereo camera calibration. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 3180–3185, April 2007. doi: 10.1109/ROBOT.2007.363963.

S. A. Shafer L. B. Wolff and G. E. Healey. Shape Recovery.Physics-Based Vision: Principles and Practice, Boston.

Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2169–2178. IEEE, 2006.

Stefan Leutenegger, Margarita Chli, and Roland Y. Siegwart. Brisk: Binary robust invariant scalable keypoints. *Computer Vision (ICCV)*, 2011, 2011.

Sergey Levine, Peter Pastor, Alex Krizhevsky, Julian Ibarz, and Deirdre Quillen. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *The International Journal of Robotics Research*, 0(0):0278364917710318,

0. doi: 10.1177/0278364917710318. URL https://doi.org/10.1177/0278364917710318.

Wai Li. Ho and kleeman, lindsay, segmentation and modeling of visually symmetric objects by robot actions, the international journal of robotics research. pages 1124–1142, 2011.

Tsz-Wai Rachel Lo and J Paul Siebert. Local feature extraction and matching on range images: 2.5 d sift. *Computer Vision and Image Understanding*, 113(12):1235–1250, 2009.

D. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, pages 1150–1157, vol.2. 2.2.2, 1999. Vol. 2. IEEE.

D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004a.

D G Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004b. ISSN 0920-5691. doi: 10.1023/B:VISI.0000029664.99615.94.

DavidG. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004c. ISSN 0920-5691.

Natalia Lyubova, Serena Ivaldi, and David Filliat. From passive to interactive object learning and recognition through self-identification on a humanoid robot, autonomous robots. 2016.

Jeremy Ma, Timothy H Chung, and Joel Burdick. A probabilistic framework for object search with 6-dof pose estimation. *The International Journal of Robotics Research*, 30 (10):1209–1228, 2011. doi: 10.1177/0278364911410090.

Jeffrey Mahler, Jacky Liang, Sherdil Niyaz, Michael Laskey, Richard Doan, Xinyu Liu, Juan Aparicio Ojea, and Ken Goldberg. Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. *CoRR*, abs/1703.09312, 2017a. URL http://arxiv.org/abs/1703.09312.

Jeffrey Mahler, Matthew Matl, Xinyu Liu, Albert Li, and David Gealyand Ken Goldberg. Dex-net 3.0: Computing robust robot suction grasp targets in point clouds using a new analytic model and deep learning. *CoRR*, abs/1703.09312, 2017b. URL http://arxiv.org/abs/1703.09312.

S. Marden and J. Guivant. Improving the performance of icp for realtime applications using an approximate nearest neighbour search. In *Proceedings of Australasian Conference on*

Robotics and Automation, page 35. Australasian Conference on Robotics and Automation, 2012.

D. Marr and E. Hildreth. *Theory of Edge Detection*, volume 207. Proceedings of the Royal Society of London. Series B, Biological Sciences, No. 1167.(Feb. 29 pp. 187-217, 1980.

D. Marr, S. Ullman, and T. Poggio. Vision: A computational investigation into the human representation and processing of visual information. *W. H. Freeman & Company, San Francisco*, 2:1, 1983.

J Matas, O Chum, M Urban, and T Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *In British Machine Vision Conference*, volume 1, pages 384–393, 2002.

Stefano Mattoccia, Simone Giardino, and Andrea Gambini. *Accurate and Efficient Cost Aggregation Strategy for Stereo Correspondence Based on Approximated Joint Bilateral Filtering*, pages 371–380. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010. ISBN 978-3-642-12304-7. doi: 10.1007/978-3-642-12304-7_35. URL `https://doi.org/10.1007/978-3-642-12304-7_35`.

D Meger, A Gupta, and J J Little. Viewpoint detection models for sequential embodied object category recognition. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 5055–5061, 2010. doi: 10.1109/ROBOT.2010.5509703.

F. Mokhtarian, N. Khalili, and P. Yuen. Multi-scale free-form 3d object recognition using 3d models. *Image and Vision Computing*, 19(5):271 – 281, 2001. ISSN 0262-8856. doi: https://doi.org/10.1016/S0262-8856(00)00076-7. URL `http://www.sciencedirect.com/science/article/pii/S0262885600000767`.

G. R. Mueller and H. J. Wuensche. Continuous extrinsic online calibration for stereo cameras. In *2016 IEEE Intelligent Vehicles Symposium (IV)*, pages 966–971, June 2016. doi: 10.1109/IVS.2016.7535505.

Robin Murphy and Amol Mali. Lessons learned in integrating sensing into autonomous mobile robot architectures. *Journal of Experimental & Theoretical Artificial Intelligence*, 9(2):191–209, April 1997. ISSN 0952-813X. doi: 10.1080/095281397147077.

J. Neubert and N. J. Ferrier. Robust active stereo calibration. In *Robotics and Automation, 2002. Proceedings. ICRA '02. IEEE International Conference on*, volume 3, pages 2525–2531, 2002. doi: 10.1109/ROBOT.2002.1013611.

R. A. Newcombe, S. J. Lovegrove, and A. J. Davison. Dtam: Dense tracking and mapping in real-time. In *2011 International Conference on Computer Vision*, pages 2320–2327, Nov 2011a. doi: 10.1109/ICCV.2011.6126513.

Richard A Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on*, pages 127–136. IEEE, 2011b.

A. Noe. *Action in Perception*. MIT Press, Action in Perception, USA, 1979.

J. Novatnack and K. Nishino. Scale-dependent 3d geometric features. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8, Oct 2007. doi: 10.1109/ICCV.2007.4409084.

Timo Ojala, Matti Pietikäinen, and Topi Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(7):971–987, 2002.

M. Okutomi and T. Kanade. A multiple-baseline stereo. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 15(4):353–363, 1993.

J. Kevin O'Regan and Alva No. A sensorimotor account of vision and visual consciousness. *Behavioral and Brain Sciences*, 24(5):939973, 2001. doi: 10.1017/S0140525X01000115.

Diana Pagliari and Livio Pinto. Calibration of kinect for xbox one and comparison between the two generations of microsoft sensors. *Sensors*, 15(11):27569, 2015. ISSN 1424-8220. doi: 10.3390/s151127569. URL `http://www.mdpi.com/1424-8220/15/11/27569`.

John C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *ADVANCES IN LARGE MARGIN CLASSIFIERS*, pages 61–74. MIT Press, 1999.

Stylianos Ploumpis, Angelos Amanatiadis, and Antonios Gasteratos. A stereo matching approach based on particle filters and scattered control landmarks. *Image and Vision Computing*, 38(Supplement C):13 – 23, 2015. ISSN 0262-8856. doi: https://doi.org/10.1016/j.imavis.2015.04.001. URL `http://www.sciencedirect.com/science/article/pii/S0262885615000347`.

M. I. Posner and Y. Cohen. Components of visual orienting. *Vol.*, 2(4):531–556, 1984.

Eric T. Psota, Jdrzej Kowalczuk, Jay Carlson, and Lance C. Perez. *A local iterative refinement method for adaptive support-weight stereo matching*, volume 1, pages 271–277. 2011. ISBN 9781601321916.

Jess M. Prez and Pablo Snchez. Real-time stereo matching using memory-efficient belief propagation for high-definition 3d telepresence systems. *Pattern Recognition Letters*, 32 (16):2250 – 2253, 2011. ISSN 0167-8655. doi: https://doi.org/10.1016/j.patrec.2011.06.016. URL `http://www.sciencedirect.com/science/article/pii/S0167865511001929`. Advances in Theory and Applications of Pattern Recognition, Image Processing and Computer Vision.

Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, and Andrew Y Ng. Ros: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5, 2009.

J. Ross Quinlan. Induction of decision trees. *Machine learning*, 1(1):81–106, 1986.

S. Rana, S. Gaj, A. Sur, and P. K. Bora. Detection of fake 3d video using cnn. In *2016 IEEE 18th International Workshop on Multimedia Signal Processing (MMSP)*, pages 1–5, Sept 2016. doi: 10.1109/MMSP.2016.7813368.

Carl Edward Rasmussen. Gaussian processes for machine learning. 2006.

B. Rasolzadeh, K. Huebner M. Bj"orkman, and D. Kragic. An active vision system for detecting, fixating and manipulating objects in the real world. *The International Journal of Robotics Research Online First, published on August*, 28, 2009.

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, 2010. ISSN 0278-3649. doi: 10.1177/0278364909346069.

Gunnar Rätsch, Takashi Onoda, and K-R Müller. Soft margins for adaboost. *Machine learning*, 42(3):287–320, 2001.

Ethan Rublee et al. Orb: an efficient alternative to sift or surf. *Computer Vision (ICCV), 2011*, 2011.

R. B. Rusu, N. Blodow, and M. Beetz. Robotics and automation. 2009:3212–3217, 2009a.

Radu Bogdan Rusu. Semantic 3d object maps for everyday manipulation in human living environments. *KI-Künstliche Intelligenz*, 24(4):345–348, 2010.

Radu Bogdan Rusu and Steve Cousins. 3D is here: Point Cloud Library (PCL). In *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, May 9-13 2011.

R.B. Rusu, N. Blodow, and M. Beetz. Fast point feature histograms (fpfh) for 3d registration. In *IEEE International Conference on Robotics and Automation*, pages 3212–3217, May 2009b. doi: 10.1109/ROBOT.2009.5152473.

S Rasoul Safavian and David Landgrebe. A survey of decision tree classifier methodology. *IEEE transactions on systems, man, and cybernetics*, 21(3):660–674, 1991.

Joaquim Salvi, Xavier Armangu, and Joan Batlle. A comparative review of camera calibrating methods with accuracy evaluation. *Pattern Recognition*, 35(7):1617 – 1635, 2002. ISSN 0031-3203. doi: http://dx.doi.org/10.1016/S0031-3203(01)00126-1. URL `http://www.sciencedirect.com/science/article/pii/S0031320301001261`.

Iliyana Samardzhieva and Aamir Khan. Necessity of bio-imaging hybrid approaches accelerating drug discovery process (mini-review). *International Journal of Computer Applications*, 182(6):1–10, Jul 2018. ISSN 0975-8887. doi: 10.5120/ijca2018917564. URL `http://www.ijcaonline.org/archives/volume182/number6/29763-2018917564`.

Michael Sapienza, Miles Hansard, and Radu Horaud. Real-time visuomotor update of an active binocular head. *Autonomous Robots*, 34(1-2):35–45, 2013.

D. Schiebener, A. Ude, J. Morimoto, T. Asfour, and R. Dillmann. 11th ieee-ras international conference on humanoid robots (humanoids). 2011.

Tanner Schmidt, Richard Newcombe, and Dieter Fox. Dart: dense articulated real-time tracking with consumer depth cameras. *Autonomous Robots*, 39(3):239–258, Oct 2015. ISSN 1573-7527. doi: 10.1007/s10514-015-9462-z. URL `https://doi.org/10.1007/s10514-015-9462-z`.

David F Shanno. On broyden-fletcher-goldfarb-shanno method. *Journal of Optimization Theory and Applications*, 46(1):87–94, 1985.

P.M. Sharkey, D.W. Murray, and J.J. Heuring. On the kinematics of robot heads. *Robotics and Automation, IEEE Transactions on*, 13(3):437 –442, jun 1997. ISSN 1042-296X. doi: 10.1109/70.585904.

JP Siebert and CW Urquhart. C3d: a novel vision-based 3-d data acquisition system. In *Image Processing for Broadcast and Video Production*, pages 170–180. Springer, 1995.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

J. Sinapov and A. Stoytchev. Grounded object individuation by a humanoid robot. In *2013 IEEE International Conference on Robotics and Automation*, pages 4981–4988, May 2013. doi: 10.1109/ICRA.2013.6631289.

J. Sinapov, C. Schenck, and A. Stoytchev. Learning relational object categories using behavioral exploration and multimodal perception. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5691–5698, May 2014a. doi: 10.1109/ICRA.2014.6907696.

Jivko Sinapov, Connor Schenck, Kerrick Staley, Vladimir Sukhoy, and Alexander Stoytchev. Grounding semantic categories in behavioral interactions: Experiments with 100 objects. *Robotics and Autonomous Systems*, 62(5):632 – 645, 2014b. ISSN 0921-8890. doi: https://doi.org/10.1016/j.robot.2012.10.007. URL `http://www.sciencedirect.com/science/article/pii/S092188901200190X`. Special Issue Semantic Perception, Mapping and Exploration.

Bastian Steder, Radu Bogdan Rusu, Kurt Konolige, and Wolfram Burgard. Narf: 3d range image features for object recognition. In *Workshop on Defining and Solving Realistic Perception Problems in Personal Robotics at the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, volume 44, 2010.

Jan Stria, Daniel Průša, Václav Hlaváč, Libor Wagner, Vladimír Petrík, Pavel Krsek, and Vladimír Smutný. Garment perception and its folding using a dual-arm robot. In *Proc. International Conference on Intelligent Robots and Systems (IROS)*, pages 61–67. IEEE, 9 2014.

L. Sun, G. Aragon-Camarasa, S. Rogers, and J. P. Siebert. Accurate garment surface analysis using an active stereo robot head with application to dual-arm flattening. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 185–192, May 2015a. doi: 10.1109/ICRA.2015.7138998.

L. Sun, G. Aragon-Camarasa, S. Rogers, and J. P. Siebert. Accurate garment surface analysis using an active stereo robot head with application to dual-arm flattening. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 185–192, May 2015b. doi: 10.1109/ICRA.2015.7138998.

L. Sun, S. Rogers, G. Aragon-Camarasa, and J. P. Siebert. Recognising the clothing categories from free-configuration using gaussian-process-based interactive perception. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2464–2470, May 2016a. doi: 10.1109/ICRA.2016.7487399.

L. Sun, Z. Yan, A. Zaganidis, C. Zhao, and T. Duckett. Recurrent-octomap: Learning state-based map refinement for long-term semantic mapping with 3d-lidar data. *IEEE Robotics and Automation Letters*, 3(4), 2018. doi: 10.1109/LRA.2018.2856268.

Li Sun. Integrated visual perception architecture for robotic clothes perception and manipulation. *phd thesis*, 2016. URL `http://theses.gla.ac.uk/id/eprint/7685`.

Li Sun, Gerardo Aragon-Camarasa, Simon Rogers, and J.Paul Siebert. Accurate garment surface analysis using an active stereo robot head with application to dual-arm flattening. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 185–192, May 2015c. doi: 10.1109/ICRA.2015.7138998.

Li Sun, Gerardo Aragon Camarasa, Aamir Khan, Simon Rogers, and Paul Siebert. A precise method for cloth configuration parsing applied to single-arm flattening. *International Journal of Advanced Robotic Systems*, 13, April 2016b. doi: 10.5772/62513. URL `http://eprints.gla.ac.uk/117227/`. CLOPEMA - 288553 (Clothes Perception and Manipulation).

Li Sun, Simon Rogers, Gerardo Aragon-Camarasa, and J. Paul Siebert. Recognising the clothing categories from free-configuration using gaussian-process-based interactive perception. In *IEEE International Conference on Robotics and Automation (ICRA*, 2016c.

Li Sun, Zhi Yan, Sergi Molina Mellado, Marc Hanheide, and Tom Duckett. 3dof pedestrian trajectory prediction learned from long-term autonomous mobile robot deployment data. *arXiv preprint arXiv:1710.00126*, 2017a.

Li Sun, Cheng Zhao, and Rustam Stolkin. Weakly-supervised DCNN for RGB-D object recognition in real-world applications which lack large-scale annotated training data. *CoRR*, abs/1703.06370, 2017b. URL `http://arxiv.org/abs/1703.06370`.

Li Sun, Cheng Zhao, and Rustam Stolkin. Weakly-supervised dcnn for rgb-d object recognition in real-world applications which lack large-scale annotated training data. *arXiv preprint arXiv:1703.06370*, 2017c.

Shuai Tang, Xiaoyu Wang, Xutao Lv, Tony X Han, James Keller, Zhihai He, Marjorie Skubic, and Shihong Lao. Histogram of oriented normal vectors for object recognition with a depth sensor. In *Proceedings of 11th Asian Conference on Computer Vision (ACCV 2012)*, 2012.

T. P. Tho, N. T. Thinh, and N. H. Bich. Design and development of the vision sorting system. In *2016 3rd International Conference on Green Technology and Sustainable Development (GTSD)*, pages 217–223, Nov 2016. doi: 10.1109/GTSD.2016.57.

Michal Jilich Thuy-Hong-Loan Le, Alberto Landini, Matteo Zoppi, Dimiter Zlatanov, and Rezia Molfino. On the development of a specialized flexible gripper for garment handling. *Journal of Automation and Control Engineering Vol*, 1(3), 2013.

C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization method. *Int. J. of Computer Vision*, 9(2):137–154, 1992.

F. Tombari, S. Mattoccia, L. Di Stefano, and E. Addimanda. Classification and evaluation of cost aggregation methods for stereo correspondence. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, June 2008. doi: 10.1109/CVPR. 2008.4587677.

R.Y. Tsai and R.K. Lenz. A new technique for fully autonomous and efficient 3d robotics hand/eye calibration. *Robotics and Automation, IEEE Transactions on*, 5(3):345 –358, jun 1989. ISSN 1042-296X. doi: 10.1109/70.34770.

Raoul Tubiana, Jean-Michel Thomine, and Evelyn Mackin. *Examination of the hand and wrist*. CRC Press, 1998.

Shimon Ullman, Vidal-Naquet, Michel U, and Erez Sali. Visual features of intermediate complexity and their use in classification. *Nature*, 5, 2002.

Ulrich Viereck, Andreas ten Pas, Kate Saenko, and Robert Platt. Learning a visuomotor controller for real world robotic grasping using easily simulated depth images. *CoRR*, abs/1706.04652, 2017. URL `http://arxiv.org/abs/1706.04652`.

D. Walther, U. Rutishauser, C. Koch, and P. Perona. Selective visual attention enables learning and recognition of multiple objects in cluttered scenes. *Computer Vision and Image Understanding*, 100(1-2):41–63, October 2005.

Greg Welch and Gary Bishop. An introduction to the kalman filter. Technical report, Chapel Hill, NC, USA, 1995.

Christopher KI Williams and David Barber. Bayesian classification with gaussian processes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(12):1342–1351, 1998.

B. Willimon, I Walker, and S. Birchfield. A new approach to clothing classification using mid-level layers. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 4271–4278, May 2013. doi: 10.1109/ICRA.2013.6631181.

Benjamin A Wolfe and David Whitney. Saccadic remapping of object-selective information. *Attention, perception & psychophysics*, 77(7):2260–9, oct 2015. ISSN 1943-393X. doi: 10.3758/s13414-015-0944-z.

R. J. Woodham. Photometric method for determining surface orientation from multiple images. *Journal of Optical Engineering*, 19(1):138–144, 1980.

Changchang Wu. SiftGPU: A GPU implementation of scale invariant feature transform (SIFT). http://cs.unc.edu/ ccwu/siftgpu, 2007.

R. H. Wurtz, W. M. Joiner, and R. A. Berman. Neuronal mechanisms for visual stability: progress and problems. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 366(1564):492–503, 2011.

Wendy S. Yambor, Bruce A. Draper, and J. Ross Beveridge. Analyzing pca-based face recognition algorithm: Eigenvector selection and distance measures. empirical evaluation methods in computer vision, 2002.

Zhi Yan, Li Sun, Tom Duckett, and Nicola Bellotto. Multisensor online transfer learning for 3d lidar-based human classification with a mobile robot. *arXiv preprint arXiv:1801.04137*, 2018.

Jianchao Yang, Kai Yu, Yihong Gong, and Tingwen Huang. Linear spatial pyramid matching using sparse coding for image classification. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1794–1801. IEEE, 2009.

Qingxiong Yang. A non-local cost aggregation method for stereo matching. In *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, CVPR '12, pages 1402–1409, Washington, DC, USA, 2012. IEEE Computer Society. ISBN 978-1-4673-1226-4. URL `http://dl.acm.org/citation.cfm?id=2354409.2355045`.

A. Zaganidis, L. Sun, T. Duckett, and G. Cielniak. Integrating deep semantic segmentation into 3-d point cloud registration. *IEEE Robotics and Automation Letters*, 3(4):2942–2949, Oct 2018. doi: 10.1109/LRA.2018.2848308.

Matthew D Zeiler, Graham W Taylor, and Rob Fergus. Adaptive deconvolutional networks for mid and high level feature learning. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2018–2025. IEEE, 2011.

Cheng Zhao, Li Sun, Bing Shuai, Pulak Purkait, and Rustam Stolkin. Dense rgb-d semantic mapping with pixel-voxel neural network. *arXiv preprint arXiv:1710.00132*, 2017a.

Cheng Zhao, Li Sun, and Rustam Stolkin. A fully end-to-end deep learning approach for real-time simultaneous 3d reconstruction and material recognition. In *Advanced Robotics (ICAR), 2017 18th International Conference on*, pages 75–82. IEEE, 2017b.

Cheng Zhao, Li Sun, Pulak Purkait, Tom Duckett, and Rustam Stolkin. Learning monocular visual odometry with dense 3d mapping from dense 3d flow. *arXiv preprint arXiv:1803.02286*, 2018.