TU Delft

Delft University of Technology

Human centric object perception for service robots

Alargarsamy Balasubramanian, Aswin

DOI
10.4233/uuid:aa07ed45-93bc-4e8e-a7f2-e5728b187ac2

Publication date
2016

Document Version
Final published version

Citation (APA)
Alargarsamy Balasubramanian, A. (2016). *Human centric object perception for service robots*. [Dissertation (TU Delft), Delft University of Technology]. https://doi.org/10.4233/uuid:aa07ed45-93bc-4e8e-a7f2-e5728b187ac2

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.
For technical reasons the number of authors shown on this cover page is limited to a maximum of 10.

Human Centric Object Perception for Service Robots

Aswin Chandarr Alagarsamy Balasubramanian

Human Centric Object Perception for Service Robots

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. ir. K.C.A.M. Luyben,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op Maandag, 3-Oktober- 2016 om 15:00 uur

door

Aswin Chandarr Alagarsamy Balasubramanian

BioMechanical Engineering,
Delft University of Technology, The Netherlands,
geboren te Karur, India.

Dit proefschrift is goedgekeurd door de

promotor: prof. dr. ir. P.P. Jonker
copromotor: Dr. M. Rudinac

Samenstelling promotiecommissie:

Rector Magnificus,	voorzitter
Prof. dr. ir. P.P. Jonker	Technische Universiteit Delft
Dr. M. Rudinac	Technische Universiteit Delft
Onafhankelijke leden:	
Prof. dr. ir. B.J.A. Krose	Hogeschool Amsterdam
Dr. K.V. Hindriks	Technische Universiteit Delft
Prof. dr. ir. M.Wisse	Technische Universiteit Delft
Prof. dr. ir. P.H.N. de With	Technische Universiteit Eindhoven
Prof. dr. J. Dankelman	Technische Universiteit Delft
Prof. dr. F.C.T. van der Helm	Technische Universiteit Delft, reservelid

TU Delft Delft University of Technology

Parts of this thesis have been performed under the DORA project.

Keywords: Service robots, Robot vision, Object recognition

Cover design: Revathi Pillai

Front & Back: The myriad shapes and colors juxtaposed with binary patterns depict the multiple levels at which the robot Lea perceives the world.

Copyright © 2016 by A. Chandarr

ISBN 978-90-8759-634-7

An electronic version of this dissertation is available at
http://repository.tudelft.nl/.

Author email: aswinchandarr@gmail.com

Contents

Summary

The research interests and applicability of robotics have diversified and seen a tremendous growth in recent years. There has been a shift from industrial robots operating in constrained settings to consumer robots working in dynamic environments associated closely with everyday human activities. Personal service robots to assist elderly, compliant robots with advanced perception skills for flexible manufacturing and autonomous driving vehicles for safe transportation are among the promising directions. In all these cases, robots have to work in close cooperation with human users and an intuitive higher level interaction between robots and layman users is essential for its widespread acceptability. Hence in this thesis, development of cognitive and perceptual skills in humans is studied and applied to the development of robot's perceptual skills, especially based on visual information from a user interaction point of view.

A physical robot is developed from scratch considering the aspects of affordability and user acceptability. A 9 DoF robot, LEA which incorporates a differential drive base, 4 DoF arm with a gripper and a pan-tilt neck supporting the robot's head. The entire mechanics and control electronics are custom developed leading to decreased mechanical complexity and increased flexibility in physical dimensions. All the components are well integrated with a socially appealing industrial design which has been well received by the public and media. The limitations arising from simplified mechanics and affordable hardware are compensated by advanced adaptive vision algorithms to achieve the required functionalities of a service robot. A generic human centric architecture for highly autonomous and interactive robots is proposed to integrate various capabilities of a robot that are triggered by user interaction. A specific case of object recognition is investigated, as many tasks faced by such robots involve perception and manipulation of different household objects.

An intuitive non-verbal interaction between a user and a robot for conveying objects of interest to the robot is developed. The developed spatial grounding model can detect the object of user interest independent of the relative position between the robot, the user and the object and without any prior training. This is achieved by a hybrid attention system combining bottom-up color saliency with depth image and top-down cues comprising user's pointing direction and gaze. Robustness of gaze based attention system is improved by automatically switching between a keypoint based and a color based approach depending on objects' texture.

The recognition of these objects is achieved with a three layered semantic recognition framework that can incorporate multiple modalities of information. Developed based on studies of human perception, this method achieves recognition robustness in unconstrained domestic environments while providing semantic grounding with human users. Modalities of color, shape and object location have been incorporated into this recognition model while maintaining flexibility to include additional

modalities. The first layer consists of semantic grounding modules that abstract raw sensory information into a probability distribution over meaningful semantic concepts familiar to humans. A second layer operates on these semantic features to obtain an object hypothesis based on every individual modality. The last layer performs knowledge association to estimate combined probability over known objects to obtain the final inference.

A novel algorithm to track contours of objects and persons to allow exploration from different viewpoints is developed. Visual model of the target is refined by considering only the dominant 3D cluster within the initial bounding box. A tracking-by-detection algorithm constrains the search space in the image by removing regions based on metric size constancy of the object and other structural patterns like perpendicular planes. A feature based on Color Naming System has been used with an online learning classifier to obtain a color probability map while the depth probability map is obtained by using a Gaussian model of the object's depth distribution. An optimal fusion of different object modalities using a target-background dissimilarity measure is developed and is used in a graphcut framework to continuously obtain contours of the target object.

The reliability of recognition of these objects in challenging domestic environments is enhanced using visual appearances from multiple views while incorporating the spatial relations between these viewpoints as well. A Sequence Alignment algorithm has been used with vector quantized features from each view to achieve view point correlation in object recognition. A fast Visual Odometry estimation has been used to obtain viewpoint relations in an unsupervised manner and this has been incorporated with segmentation to provide a standalone system that can be used in real world scenario. This system is made generic to be used with different feature vectors and a benchmark is created to compare the performance improvement achieved by the developed system with respect to single view object recognition using different feature vectors.

Object recognition in service robots can be augmented by incorporating the context of objects' use within the developed semantic recognition framework. The utility of an object can be understood by the actions performed by the user on the object and hence an Action Recognition system based on human skeletal tracking with a novelty detection method is developed to facilitate the incremental learning of new actions. Compact representations of skeletal structure are obtained using a Torso-PCA transform and are used as observations for a HMM based system to recognize user actions. Uncertainty in predictions, quantified as confidence measures are thresholded to detect unknown actions. These confidence measures are obtained through background models and different methods are evaluated with respect to sensitivity and specificity of recognition performance.

Various algorithms are developed to enhance the reliability of object perception overcoming challenges posed by dynamic environments and affordable hardware by incorporating different modalities of information available to a robot. The development of algorithms in this direction is significant as these concepts can be readily extended to incorporate user and environment recognition to complete the perceptual capabilities of robots …

Samenvatting

De interesse in onderzoek en naar de toepasbaarheid van robotica zijn de afgelopen jaren gevarieerder geworden en laten een enorme groei zien. Er heeft een verschuiving plaatsgevonden van industriële robots, statisch gebruikt in afgeperkte omgevingen, naar consumenten robots werkend in dynamische omgevingen dicht verbonden aan dagelijkse menselijke activiteiten. Tot de veelbelovende richtingen behoren persoonlijke service robots om ouderen te assisteren, compliante robots met geavanceerde perceptie vaardigheden voor flexibele productie en zelfrijdende voertuigen voor veilig vervoer. In al deze gevallen moeten robots nauw samenwerken met de gebruikers terwijl een intuïtieve interactie op hoog niveau tussen robots en gebruikers essentieel is voor wijdverbreide aanvaarding van robots. Vandaar dat in dit proefschrift de ontwikkeling van cognitieve en perceptuele vaardigheden bij de mens bestudeerd is en deze kennis is toegepast bij de ontwikkeling van deze vaardigheden bij robots, met een focus op visuele informatie verwerking vanuit het oogpunt van de interactie tussen robot en gebruiker.

Er is een fysieke robot ontwikkeld waarbij van de grond af aan de aspecten van betaalbaarheid en gebruikersacceptatie meegenomen zijn. Er is een 9 DoF robot LEA ontworpen die een differentiële aandrijfbasis heeft, een 4 DoF arm met een grijper en een pan-tilt nek die het hoofd van de robot ondersteunt. De hele mechanica en besturingselektronica zijn op maat ontwikkeld met slimme ontwerpen die tot een verminderde mechanische complexiteit en meer flexibiliteit in afmetingen hebben geleid en die voldoen aan eisen van de regeltechniek. Alle mechanische, elektrische en elektronische componenten zijn volledig geïntegreerd met een sociaal aantrekkelijk industrieel ontwerp dat goed is ontvangen door het publiek en de media. De beperkingen die voortvloeien uit vereenvoudigde mechanica en betaalbare hardware zijn gecompenseerd met behulp van geavanceerde adaptieve robot-vision algoritmen om de vereiste functionaliteit van een service robot te realiseren.

Er is een generieke mens-centrische architectuur voor hoog autonome interactieve robots ontworpen om de verschillende mogelijkheden van een robot die door een mens geactiveerd kan worden, te integreren. Specifiek is object herkenning onderzocht omdat veel taken van dit soort robots berust op perceptie en manipulatie van huishoudelijke voorwerpen.

Er is een efficiënte en intuïtieve non-verbale interactie tussen gebruiker en robot ontwikkeld om duidelijk te maken welke objecten door de gebruiker gewenst zijn. ”Spatial groundingömvat het door de robot te lokaliseren deel van de visuele scene waarin de gebruiker geïnteresseerd is. Het ontwikkelde model kan het object van belang voor de gebruiker detecteren, onafhankelijk van de relatieve positie tussen robot, gebruiker en object en zonder enige voorafgaande training.

Dit wordt bereikt door een hybride attentie systeem, dat bottom-up-kleur saliantie, diepte beeld, en top-down aanwijzingen combineert, het laatste bestaande

uit het met de vinger wijzen van de gebruiker en detectie van zijn blikrichting. De robuustheid van het op blikrichting gebaseerde attentie systeem is verbeterd door, gebaseerd op de mate van textuur van het object, automatisch over te schakelen tussen een key-point benadering en een benadering op basis van kleur.

Herkenning van deze objecten wordt bereikt met een drie-laags semantisch herkenningssysteem dat meerdere modaliteiten van informatie kan opnemen. Ontwikkeld op basis van studies van de menselijke perceptie, bereikt deze methode robuustheid in de herkenning van objecten in ongelimiteerde huishoudelijk omgevingen, waarbij het ook semantische informatie van gebruikers vastlegt. Modaliteiten van kleur, vorm en de locatie van het object zijn opgenomen in dit herkenningsmodel, met behoud van flexibiliteit om ook andere modaliteiten op te nemen. De eerste laag bestaat uit semantische ”grounding”modules die ruwe sensorische informatie abstraheren in een kansverdeling van zinvolle semantische concepten die bij de mens bekend zijn. Een tweede laag werkt op deze semantische functies om een object hypothese te verkrijgen op basis van elke individuele modaliteit. De laatste laag voert kennis associatie uit om de gecombineerde waarschijnlijkheid te schatten over bekende objecten, om uiteindelijke een gevolg te kunnen trekken.

Er is een nieuw algoritme ontwikkeld voor het volgen van de contouren van objecten en personen om verkenning uit te kunnen voeren startend vanuit verschillende gezichtspunten. Het visuele doel (target) model is verfijnd door alleen rekening te houden met het dominante 3D cluster binnen de initiële 3D omhullende rechthoek. Een *track by detection* algoritme limiteert de zoekruimte in de afbeelding door het verwijderen van regio’s, gebaseerd op metrische grootte constantheid van het object en andere structurele patronen zoals loodrecht vlakken. Een kenmerk op basis van het ”Color Naming Systemïs gebruikt in een online-lerend classificatie systeem om een op kleur gebaseerd waarschijnlijkheidsmap te krijgen, terwijl de dieptewaarschijnlijkheidsmap wordt verkregen met behulp van een Gaussiaanse model van de diepte distributie van het object. Er is een nieuwe methode voor het uitvoeren van de optimale fusie van verschillende object modaliteiten ontwikkeld met behulp van een target-achtergrond verschilmaat, die wordt gebruikt in het kader van een ”graphcutöm continu de contouren van het doelobject te verkrijgen.

De betrouwbaarheid van de herkenning van deze objecten in uitdagende huishoudelijk omgevingen is versterkt door zowel visuele aanzichten vanuit meerdere standpunten te gebruiken, als wel de integratie van de ruimtelijke relaties tussen deze standpunten. Een Sequence-Alignment algoritme met vector gekwantiseerde kenmerken vanuit elke gezichtspunt is gebruikt om gezichtspuntscorrelatie bij object herkenning te verkrijgen. Een snelle Visuele Odometry schatting is gebruikt voor het ”unsupervised” verkrijgen van gezichtspunt relaties, en dit is samengenomen met segmentatie om een stand-alone systeem te krijgen dat gebruikt kan worden in scenario’s in de echte wereld. Dit systeem is generiek gemaakt om met verschillende kenmerkvectoren te worden gebruikt en er is een benchmark gemaakt om de prestaties te vergelijken van het ontwikkelde systeem met enkel-zicht objectherkenningssystemen bij gebruik van verschillende kenmerkvectoren.

Object herkenning bij service robots kan binnen het ontwikkelde semantische herkenningskader worden uitgebreid door de integratie van de context van het ge-

bruik van de objecten. Het nut van een object kan worden begrepen door de acties die worden uitgevoerd door de gebruiker van het object en daarom is er, om het incrementeel leren van nieuwe acties te vergemakkelijken, een Actie-Herkennings-Systeem ontworpen dat gebaseerd is op het volgen van het menselijke skelet door middel van een nieuwheidsdetectie methode. Een compacte representatie van de skelet informatie wordt verkregen met behulp van een Torso-PCA transformatie en die wordt gebruikt als observatie voor een HMM gebaseerd systeem ter herkenning van acties van de gebruiker. Om onbekend acties te detecteren wordt de onzekerheid in voorspellingen, gekwantificeerd als vertrouwensmaat, gedrempeld. De vertrouwensmaten zijn verkregen door middel van achtergrond modellen en er zijn verschillende methoden geëvalueerd om inzicht te krijgen in de gevoeligheid en specificiteit van de herkenningsprestaties.

Er zijn verschillende algoritmen ontwikkeld voor het verbeteren van de betrouwbaarheid van object perceptie die uitdagingen overwinnen zoals dynamische omgevingen en betaalbare hardware, door middel van de integratie van verschillende modaliteiten van informatie beschikbaar voor de robot. De ontwikkeling van algoritmes in deze richting is significant omdat de concepten makkelijk kunnen worden uitgebreid met het incorporeren van gebruikers- en omgevings-herkenning om daarmee de perceptuele vermogens van robots te voltooien....

1

Introduction

1.1. Robots in modern society

Robotics is amalgamation of various disciplines spread across a wide spectrum of technology. It involves fields of pattern recognition, probabilistics, artificial intelligence, industrial design, various levels of software and engineering fields like mechanical, materials, electrical and electronics engineering. Also seemingly farther technologies such as 3D printing, smart phones and chemical engineering contribute to rapid progress in robotics by providing good appearance, better user interfaces and enhanced battery capacities respectively. With rapid advancements in all these different fields, robotics has matured from being operated in controlled industrial environments to dynamic human environments. The past decade has seen application of robotics to tackle challenges in domains which were previously unrealistic. These include fields like medical surgery, search and rescue (SAR), restoring and cleaning pipelines, construction [1], underwater and space exploration, all terrain payload carriers [2], personal assistance, rehabilitation among many others.

These new generation of robots are also no longer restricted to niche and advanced research laboratories and experimental platforms. They have been already deployed commercially in fields like agriculture [3], small households tasks [4], flexible,reconfigurable small scale assembly systems [5], automated cars [6], drones [7], warehousing [8] and tele presence [9] (Figure 1.1).

This rapid development of robotic technologies is not just fuelled by scientific curiosity of researchers, but also from economic pressure of commercial interests as well. This can be seen from the large number of startup companies in robotics and also by the amount of capital infused into this field, in both academia and industry. This can be attributed to a strong price-performance ratio of small scale robotic solutions.

This good price-performance ratio stems from rapid progress in fields of *Computing Power Density and Software sophistication*. The hardware and processing

Figure 1.1: Current commercial robots: Clockwise from Top Left, Baxter, Fetch, Neato, Amazon Prime Air

power is increasing following predictions of Moore's law. The size of processors and peripherals have become quite small and this combined with rising power efficiency, memory storage density and a large demand for smartphones have enabled deployment of high computing power hardware on standalone mobile systems that can operate without continuous external power. There have also been advances in parallel processing units such as GPUs which have allowed very high throughput computing with generic hardware without requiring specialized digital hardware designs like FPGAs. These advances in hardware have propelled development of algorithms with high computational load such as Deep learning [10], large scale probabilistic approaches to path planning (RRT, PRM [11]), localization (AMCL [12]) etc, which can now work in runtime conditions. The demand for more computational power by such algorithms, which find wide spread commercial applications, drives the development of hardware, thereby decreasing the overall cost.

With the confluence of appropriate circumstances in technology, economics and market readiness, the field of robotics contains an enormous potential, which if harnessed and directed in a constructive manner can help us tackle future challenges to enhance the quality and enrich human life.

1.2. Future of robotics applications

All these advancements have powered robots with capabilities that trigger discussions over robots being tools or companions for humans [13]. Among the plethora of fields in which robotics can be applied, the seemingly diverse domains of care, safety and space exploration provide ample scope to positively influence our lives in the coming decades.

1.2.1. Care Robotics

Currently economically developed countries of Western Europe, North America and Japan are challenged to provide quality health care to the elderly people.

The average human life expectancy is increasing due to better medical insight and technologies and a healthier life style. This combined with decreasing fertility rates over the past few decades creates a demographic shift towards an ageing population in these countries. The associated issues are two fold. Firstly, studies show that the percentage of working population contributing to the economic progress and sustenance of societies will be much less by 2050 [14]. Secondly, due to prevailing cultural norms, the elderly reside independently or in care centres. In both these cases, they need dedicated care personnel to assist with their every day living activities. These factors combined, create a large strain on the nation's economy with significant rise in government spending on increasing workforce of care givers.

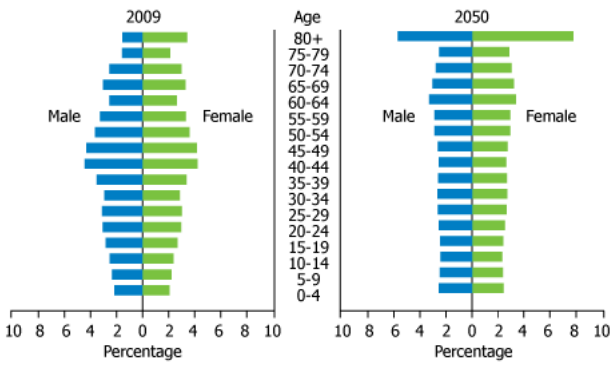


Figure 1.2: Projected demographic shift towards elderly population

These concerns are parallel to the human responsibility of creating and maintaining a society that provides a high quality life for senior citizens who in their prime years contributed to provide us the current standards of living.

Advanced robotics can be among the dominant technologies to tackle these challenges by both providing expansion of workforce and also serving as a means to care for elderly. Personal and health care robots provide us with compelling reasons to help elderly live independently and also to expand the reach of the limited

workforce of care givers. This concern has been recognized by many developed nations with initiatives such as SILVER[1] by the European Union and The Robotic Devices for Nursing Care Project in Japan [16].

These personal robots can perform simple tasks such as assisting elderly with mobility within households, actively reminding them with medications and other daily schedules. They can also assist the elderly by bringing various objects from tables and shelves which are otherwise inaccessible. These robots can support remote monitoring and assistance by family and caregivers. Additionally, they can assist elderly with dementia, sensory-motor problems and similar health issues, to lead a content life and also prevent further degradation of their capabilities.

Robots equipped with these functionalities are already being commercially approached by companies such as Robot Care Systems [17] with LEA, Toshiba with Chihira Aiko [18], Fujisoft with Parlo Robot [19], Toyota with their series of HSR robots [20], etc. (Figure 1.3).

1.2.2. Flexible manufacturing automation

Apart from helping us to care for the elderly, these new generation of robots can also assist in flexible automation in manufacturing at small industries in varied fields. Due to the development of compliant robot joints, along with advances in sensing and control, currently robots are being deployed in dynamic environments working alongside with humans [21]. Baxter [5] is an example of a collaborative robot that can be easily customized for a new task, without requiring the need for specialized personnel. There has been active interest from various traditional robot manufacturers like ABB, MotoMan, Kuka in this direction, complemented by large scale research projects such as Factory in a Day [22] and ROS Industrial [23].

1.2.3. Autonmous vehicles

Technologies derived from the field of robotics are projected to assist in enhancing safety in human lives. Recently there is an enormous interest in self driving automobiles by various players such as GoogleX, Tesla, Uber, Lyft, BMW and many other traditional car manufacturers. Statistics of trial runs of GoogleCar show 13 incidents while covering a distance of more than a million miles [24]. Successful large scale deployment of these technologies can bring an enormous increase in road safety while being more environmental friendly than prevailing systems. Apart from fully automated driving, there are many cases when AI assists in decreasing the cognitive load of users and also taking rapid actions in cases of emergency situations. These principles are also applied in other modes of transport such as aviation and shipping to enhance safety.

Additionally, robots can save many human lives in hazardous situations like nuclear disasters, natural calamities and events like fires etc. This can be achieved by robots working together with humans in tele-operative mode with varying levels of autonomy [25].

With all these applications in varied domains, robots become more and more

[1] Supporting Independant Living for the Elderly through Robotics [15]

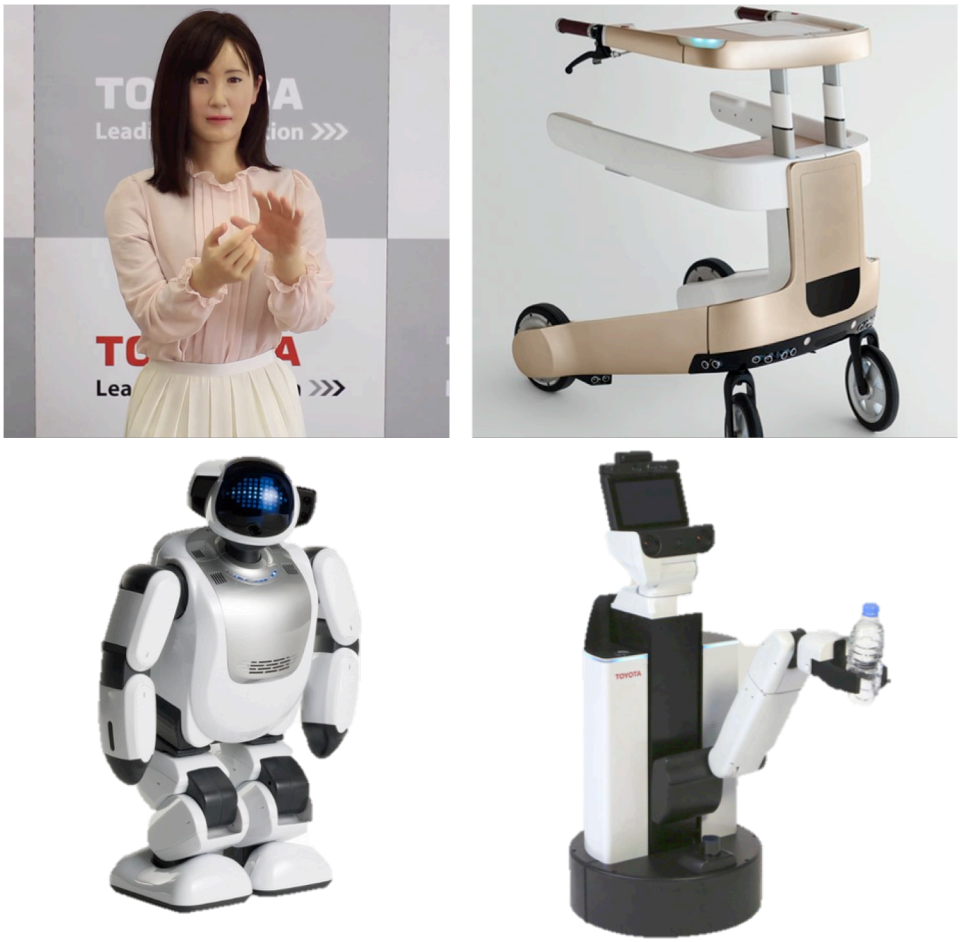


Figure 1.3: Commercial elderly care robots, Clockwise from Top Left: Chihira Aiko, LEA, Toyota HSR, Parlo

closer to every day human life. With the ability to learn from errors either automatically or with aid of a human supervision (local learning), the newly learned knowledge/skill can be updated directly into all other robots of the same class. This knowledge is also inherited by all the future robots of that class (global learning). This makes the learning process in robots massively parallel and probably enable them to outperform human capabilities at a particular task. All these factors combined, hopefully enhance the quality of human life on planet earth.

Apart from tackling existing problems on our planet, a part of humanity has always aspired to look beyond and break their limitations. In the current technological era, we have been venturing beyond our Planet Earth and pushing towards extra terrestrial exploration through manned outer space missions as well as through probes and rovers. With the entry of commercial interests in space exploration

such as deep space industries (DSI) and successful launch of reusable rockets by SpaceX [26], the prospect of a human settlement in Mars is no more only a science fiction. In all these ventures, robots will become indispensable companions of human kind, helping in variety of tasks from personal assistants to exploration, construction, mining, etc.

As discussed, robotics possesses a high potential for the future. But due to nature of the tasks they will be performing, the maximum benefit of these technologies can be achieved only by equipping robots with the ability to cooperate seamlessly with humans.

1.3. Cooperative robots

The Oxford dictionary defines cooperation as

> "The action or process of working together to the same end".

The significance and consequences of cooperative existence can be derived from the studies on history of evolution of human-kind from an insignificant species 70,000 years ago to the most dominant species having power to alter the future course of the planet [27]. The capability to function with **large scale flexible cooperation** has been deduced indispensable for rise of *Homo Sapiens*. Learning from a proven evidence, it is imperative to develop next generation robots with the ability to have a tightly knit co-existence with human beings. This entails developing perceptual and motor capabilities of robots in a human centric manner. This thesis contains work with steps in this direction, focussing on the robot's visual perception.

The flexibility and cooperation in human societies stem from two sources.

- Division of labour (Individual capability)
- Effective (abstracted) communication (Interaction)

The domain of human factors research has dealt with task allocation between machines and humans as early as 1951 when the capabilities of machines were advancing rapidly. The earliest and most widely cited work, Fitts list [28] associated machines with repetitive and deterministic tasks performed with high speed, power with short term memory requirements. Human capacity is associated with tasks in unpredictable environments which require active perception and subjective judgement based on long term memory. This is illustrated in original report as shown in Figure 1.4

Over the last 60 years, machines became sophisticated and robotics branched of as a special class of machines focussed on their autonomous capabilities enabling them to perform tasks attributed to human capacities as in Fitts list. This prompted development of human robot interaction studies in which **level of autonomy** (LOA) is used to describe the functionality of robots in a human centric manner. The most widely used model proposed by Tom Sheridan [25] describes a hierarchy from lowest LOA where computer/robot offers no assistance to very

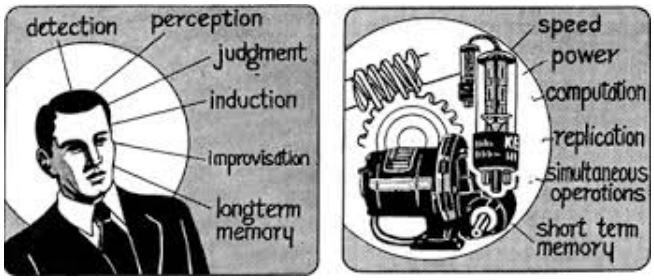


Figure 1.4: Original Illustration of Fitts list

high LOA where computer/robot decides everything and acts autonomously ignoring human inputs. This model can also be viewed as a hierarchical abstraction of information transmitted between human and robots which varies from a human providing primitive motion commands while robot streams visual feedback in a basic teleoperation scenario to a robot querying the human operator "the tomatoes are big and red, can i harvest the field (Yes/No)" in a hypothetical future scenario of an agricultural robot.

In the quest to achieve true "large scale flexible cooperation" between humans and robots as described [27], an appropriate way to consider sophistication of robots is by looking at the degree of autonomous capability and the level of human robot interaction together. This gives rise to a concept of mixed-initiative interaction [29] where any task is performed with a combined effort of agents (humans and a robots) in a "flexible interaction strategy" in which each agent contributes the best suited action at the most appropriate time. This can be visualized in a scale of LOA based on human interaction as in Figure 1.5 On the left end of the scale,

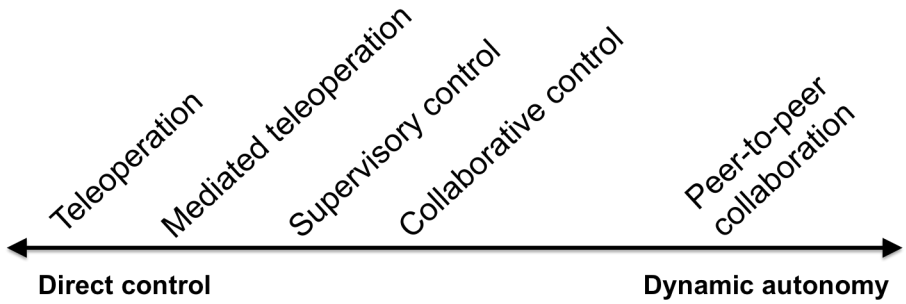


Figure 1.5: Human interaction based LOA [30]

the focus lies on developing user interface to decrease the cognitive load of human operator (direct control), while other end of dynamic autonomy requires imparting robots with cognitive skills to facilitate natural and efficient interaction with a human operator/user. It has been summarised in this survey of human robot interaction [30] that achieving peer-to-peer collaboration is more difficult than full autonomy

as it involves exhibiting fully autonomous actions at appropriate times in addition to supporting social interactions. But nevertheless, achieving seamless collaboration of robots with humans will be of immense benefit to the human race, apart from diminishing the prospects of a dystopian future controlled by autonomous agents.

In the past decade there has been an enormous progress in *Autonomy* and *Interaction*, the two main aspects of robotics. While the majority of work in this field has developed independently of each other, perception and communication in humans are tightly linked [31]. Also most of the algorithms especially in domain of perception, are used without considering the presence of multiple modalities of information available to a robot and also without incorporating the necessity of communication with human users.

> *This thesis looks at human centric perceptual capabilities of the robot, considering these requirements while taking a service robot as an example.*

Specifically, in case of visual perception, the algorithms have been borrowed from field of Pattern Recognition (PR). While the fundamental principles remain the same, the kind of resources available and challenges faced by a pure software based PR system and a Service robot are starkly different. Considering a case of recognizing objects in a household, a service robot faces challenges of noisy images with objects in arbitrary positions and with occlusions. But it has the additional advantage of having the ability to explore an object from multiple viewpoints, use its manipulators and additional modalities of information such as user preferences, location of object in a household and even a possibility to interact with human user to disambiguate and learn novel objects.

> *In this thesis, inspiration has been drawn from development of cognitive and perceptual skills in humans. With this, techniques from advancements in PR have been augmented with the multi modal sensing capability of a robot to take steps towards an efficient peer-to-peer collaboration between humans and robots.*

1.4. Spatial and Semantic Discernment

A large scale deployment of robotic agents, ranging from semi-autonomous, to autonomous to fully collaborative robots, requires an intuitive bi-directional communication between end users and robots.

Studies on social cognition [32] attribute the ability to understand the mind (internal states) of other members of the group, to previously discussed ”large scale flexible cooperation” which has led humanity to the current state of advancement. Humans have this powerful capacity to perceive the external world as psychologically meaningful representations (internal states) which then can be communicated with others through non-verbal and verbal modes. This also serves as inputs to higher order cognition. Perception, cognition and communication are tightly linked [33], [34] and are the key factors enabling peer-to-peer collaboration in human

societies. Hence, we look into the nature of human perception (specifically visual perception) and develop algorithms for imparting certain capabilities to robots.

Visual grouping is a strong aspect of human perception and has been studied extensively by early Gestalt psychologists [35]. Humans perceive their external world as a meaningful collection of various entities linked with each other. Attention mechanisms enable filtering out of background regions and focussing on important regions of interest. This ability to organize a visual scene into a graph of symbols and their relations form the fundamentals of human perception. This symbolic classification can be validated by studies in social anthropology [36]. The following quote from *Primitive Classification* by Emile Durkheim and Marcel Mauss reflects this.

> *When a person who has been blind since birth is operated upon and given sight, he does not directly see the phenomenal world which we accept as normal. Instead, he is afflicted by a painful chaos of forms and colours, a gaudy confusion of visual impressions none of which seems to bear any comprehensible relationship to the others. Only very slowly and with intense effort can he teach himself that this confusion does indeed manifest an order, and only by resolute application does he learn to distinguish and classify objects and acquire the meaning of terms such as "space" and "shape."*

This ability can be described as a part of *Spatial discernment* which also involves understanding of spatial relations implied by non verbal communication cues such as pointing at a person or an object.

Semantic discernment involves a higher level of abstraction, where a sequence of sounds (words) describe certain properties/attributes of the physical world. This becomes powerful when a group of people have the same mapping between these words and properties, and forms the fundamentals of language. According to Professor Mark Pagel [37], language is a *social technology* which is a very efficient way of communication, which has enabled an easy flow of ideas and technologies, leading to rapid progress of human race. Hence, imparting this ability to robots allows for a very efficient and intuitive way of communicating with humans in a manner humans are already familiar.

Certain aspects of Spatial and Semantic discernment have been studied in the context of robotics and pattern recognition.

In context of robotic perception, these two aspects correspond to

- Learning from environments
- Learning from human interaction

Spatial discernment is akin to learning structures in the environment from visual sensory inputs. It involves learning to segment the scene into interesting regions (objects/people) and background (redundant information) as humans inherently learn to do [36]. It also comprises of the ability of robots to distinguish objects from each other. These aspects of robot learning has been focussed earlier in the

research group of Prof. Pieter Jonker at Intelligent Vehicles and Cognitive robotics group of the TUDelft Robotics Institute where the current research was performed. The ideas and algorithms started with the PhD work of Maja Rudinac [38], described in her thesis titled "Exploration and Learning for Cognitive robots". The current work extends this research and bridge environment exploration of robots with human interaction, leading towards semantic discernment capabilities.

Semantic discernment involves a robot developing its internal states of the environment in a manner similar to humans. There has been substantial work in the AI field dedicated to this symbolic processing and cognitive architectures [39], [40] etc. These have been mainly developed in simulated environments and focussed mainly on cognition. There is a need to research algorithms for abstracting semantic symbols from the real world that can be used in these architectures and also to develop physical robots that can embody these algorithms to be tested in realistic scenarios.

There has been research into grounding certain semantic concepts for humans to direct robots actions. For instance, concepts such as "near, far, left, right" etc have been learned for robot navigation [41] Spatial prepositions such as "in front of, behind, close to" etc have been grounded to direct object manipulation by a personal robotic arm [42]. Very recently, with the advent and popularity of deep learning in combination with recurrent neural networks (RNN), there has been a significant progress in generating textual descriptions for a generic visual scene [43].

> *In this thesis, we approach the development of these discernments in context of object localization, exploration and learning, starting with development of a service robot usable in domestic environments.*

1.5. Human Centric Robot Architecture

Keeping in mind, the requirements of *Autonomy* and *Interaction* capabilities discussed in section 1.3, a Human Centric Robot Architecture that integrates various hardware and software modules is proposed in this thesis (Figure 1.6). This is derived from studying various aspects of perceptual and cognitive capabilities of humans. Such an architecture facilitates higher level interaction between robots and humans, leading to fully collaborative robots. This also allows for increased robustness in changing environments and continuous learning through user interaction. In this thesis, a complete framework is developed from a viewpoint of a Personal Service Robot (PSR), but only the perceptual part of it has been further pursued in detail.

Figure 1.6 provides an overview of internal architecture for human centric robots. Every component here is not an algorithm by itself, but an abstraction of various algorithms that can perform the required functionality. These components are generic to encompass algorithms operating on different modalities such as visual data, auditory or haptic data. Such an architecture is suitable for any robot that needs to work in cooperation with humans with a higher level of autonomy (LOA). In the case of this thesis, this is implemented on a domestic service robot (LEA).

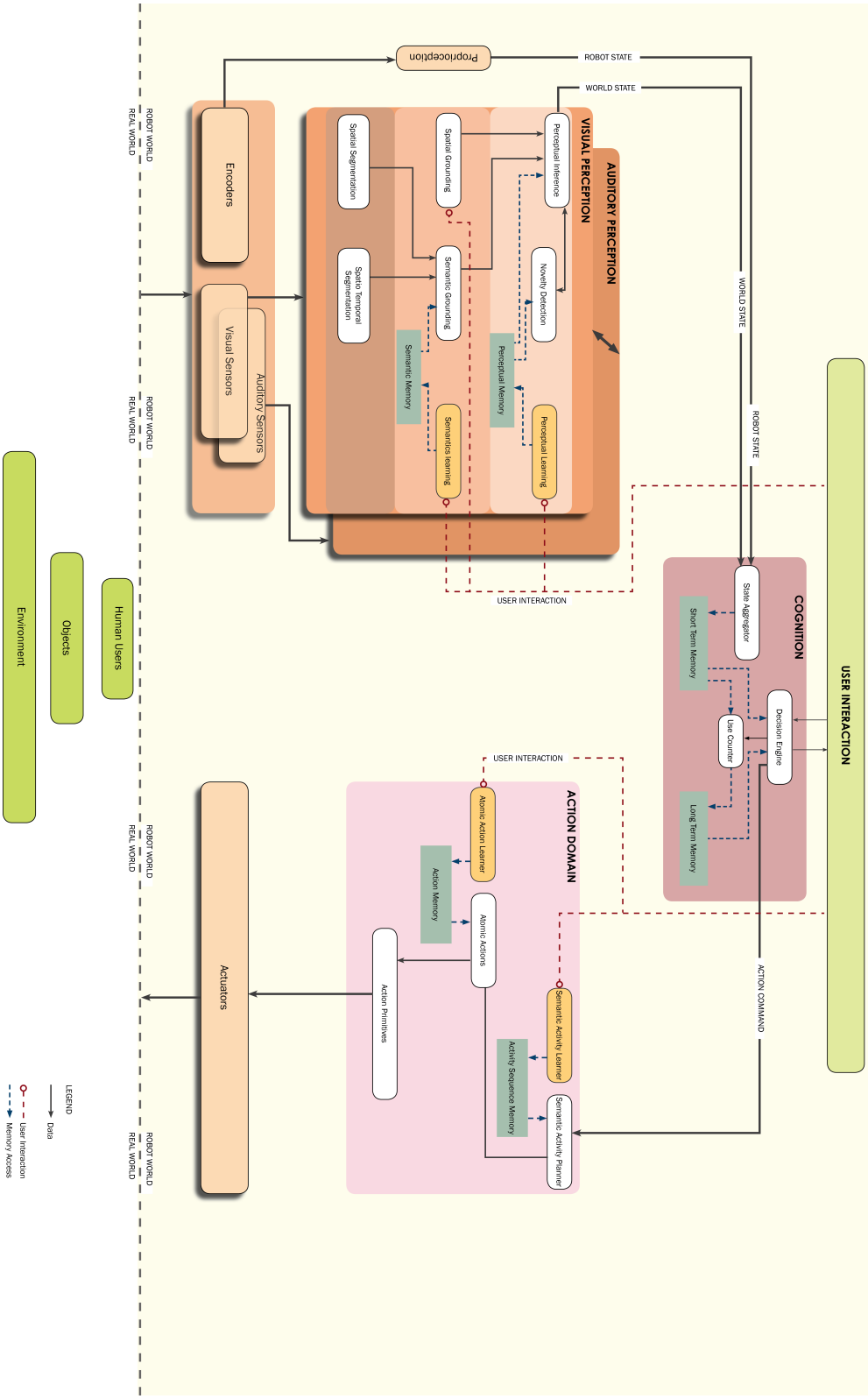


Figure 1.6: Human centric robot software architecture

At first, there is a boundary between the *physical world* and the *digital world*. The physical world has 3 entities

1. Environment
2. User
3. Robot

The digital world is embedded into the computational system of the robot. The entire proposed architecture runs here and involves the interaction between the 3 entities of the real world. The bridge between these two worlds are the sensors and actuators which form the first layer (Raw sensory sources and Basic actuators). The following table enlists few of commonly used sensors and actuators in robots.

Sensors	Actuators
3D and 2D cameras	Motors and Joints (Arm, Mobile base, Neck)
Laser scanners	Grippers
Ultrasonic and Infrared rangers	Lights
Joint encoders	Speakers
Microphone arrays	Vibration (Haptic) devices
Haptic (Force) sensors	

Table 1.1: Commonly used sensors and actuators in Personal Service Robots

There are 4 components on the highest level in this architecture

1. Perception
2. Action
3. Cognition Center
4. User Interaction

Though all these are tightly linked and algorithms in one component have to be designed satisfying the requirements for other modules, *Perception* and *Action* can be considered two distinct domains, with a little interaction between them. But *Cognition* and *User Interaction* perform on top of Action and Perception capabilities.

1.5.1. User Interaction

User Interaction directly acts on the raw sensory information to understand the intentions of the user. It involves perceptual modules that can detect and localize user hand movements (Pointing) or track their face/gaze direction, interfaces for language, understanding haptic feedback from the user. On the other hand, this also consists of *Action* modules that help the robot to communicate to the users. This can involve speech to text conversion, gestures with its arm, neck, base etc, expressing through patterns of lights and sounds etc. It is clear that, this block utilizes functions from both *Action* and *Perception* domains but solely focused on interacting with the user.

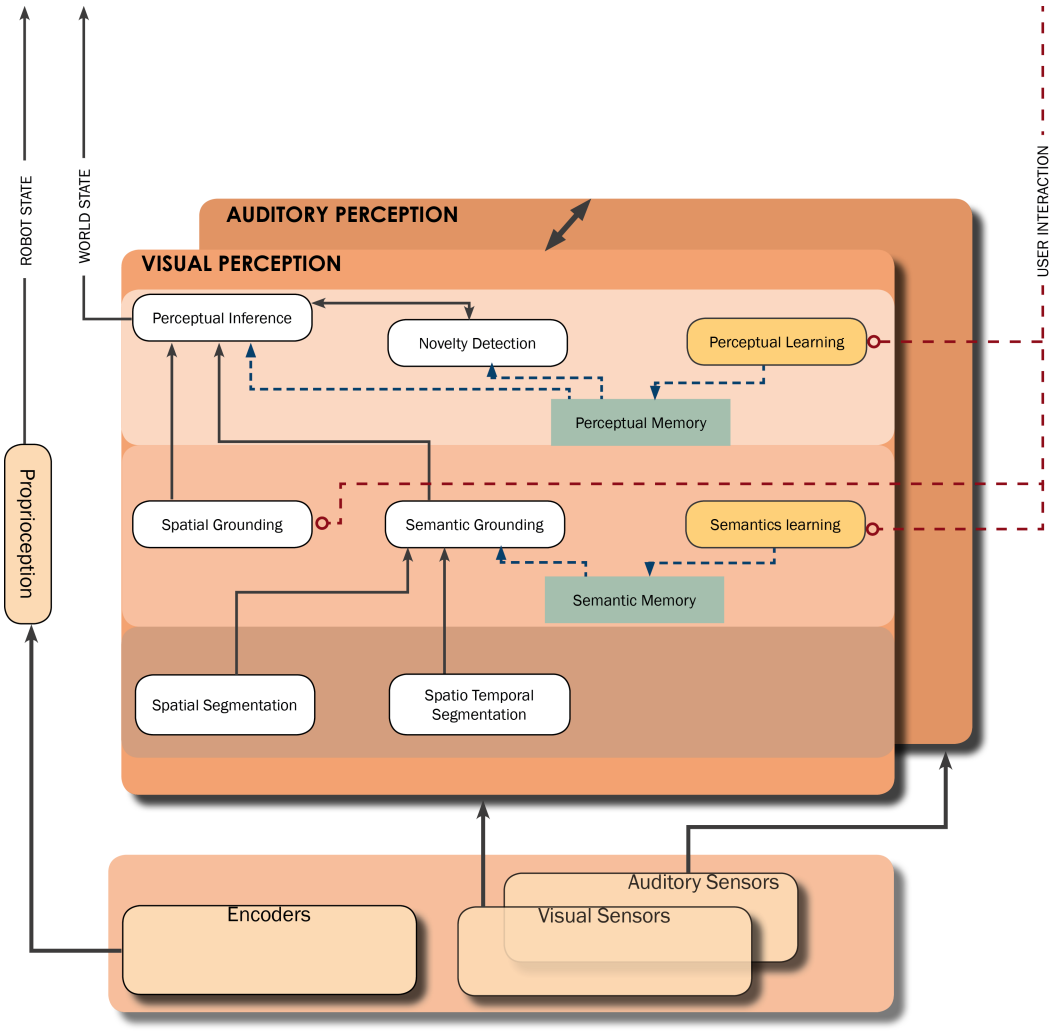


Figure 1.7: Perception module of human centric robot architecture

1.5.2. Perception Domain

Figure 1.7 shows the details of the proposed perception module. In the domain of Perception, the first step involves filtering the enormous amount of available sensory information into useful data. This is performed in *Segmentation layer* which comprises of

- Spatial segmentation involves localizing appropriate regions of interest in space relative to the robot. The algorithms that fall into this category include *Saliency* based segmentation in 2D images [38], Sound source separation and localization [44].

- Spatio-Temporal segmentation involves localizing appropriate information that vary in space and time in relation to the robot. Many algorithms such as Visual Tracking, Ego Motion estimation, SLAM, Action segmentation, Speech to Text conversion fall into this category

The results of segmentation of essential information from raw sensory information is then used by modules of *Grounding layer* with human users. This comprises of *Semantic* and *Spatial* grounding. The *Semantic grounding* abstract the filtered sensory information into semantic concepts that are compliant with human knowledge representations. For instance household objects can be described as a combination of properties such as color, shape, location and functions, while human users can be described with height, sex, physical structure, race, color of features and so on. In case of auditory information, this can involve natural language understanding.

Spatial grounding combines data from segmentation layer with the inputs from *User interaction* layer to share a common region of interest in space with human users. This can involve finding an object that a user is interested in by combining image segmentation with user's direction of pointing, gaze or voice commands that are interpreted through the *User interaction* layer. This can also comprise of understanding directions of user (right, left, up, down) or sense of distance such as near, far etc.

The next layer is the *Inference layer* which comprises of

- Perceptual Inference of abstracted data from Grounding layers to understand the state of its environment (World state) and intentions of the user (User state). This uses pre-learned information stored in the *Perceptual memory* to recognize various entities and users in the environment. For instance, recognizing a person with certain pre-trained semantic features as the robot owner. Perceptual inference includes algorithms like Object, User, Action, Voice and Location recognition.

- Novelty Detection is tightly linked with perceptual inference to detect previously unknown entities (such as objects, users and actions). This is very essential not only to avoid falsely classifying new entities into one of previously learned ones, but also to update its knowledge by learning.

- Perceptual Learning integrates tightly with *Inference* and *Novelty detection* and in combination with *User interaction* module to incrementally expand the robot's knowledge. Online learning algorithms for various perceptual inferences belong here and are responsible for updating the *Perceptual memory*

When the robot is trying to understand its environment in general without any particular objective, it is termed as *Bottom-Up* perception and these are guided by Attention mechanisms [38]

1.5.3. Action Domain (Motor Capabilities)

While the robot understands the state of *World* and *User* through its Perception modules, it performs actions on both the World and User through modules of Action

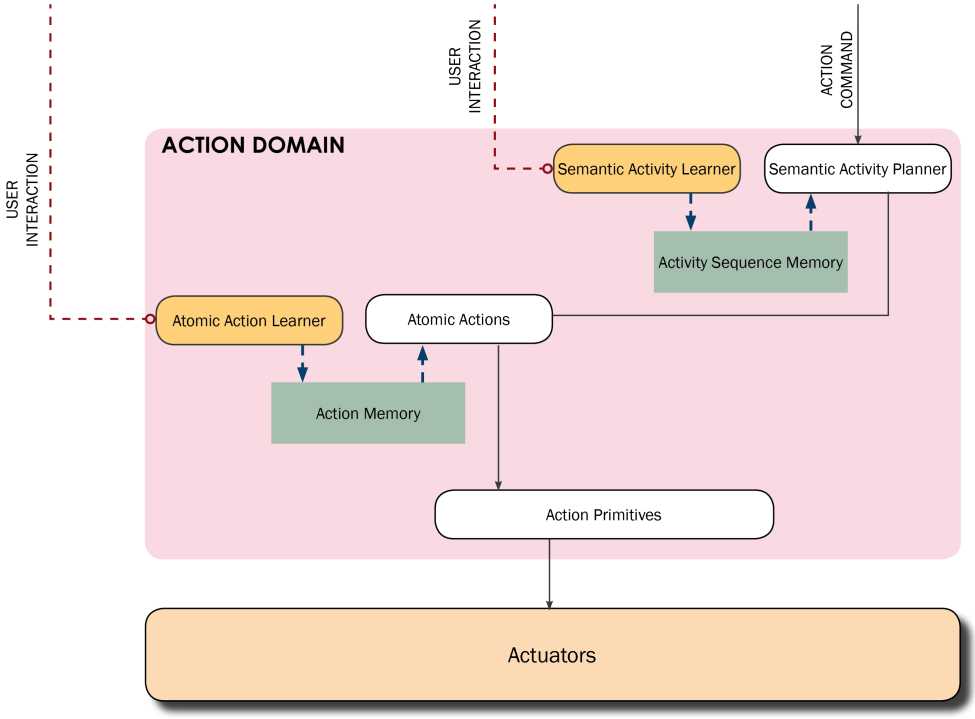


Figure 1.8: Action module of human centric robot software architecture

domain. The robots *Cognition center* as explained in 1.5.4 decides on performing tasks, which is realized through its actuators. The transformation from a decision in the *Cognition center* to actions in the physical world is accomplished through a set of modules

- Semantic Activity Planner (SAP)
- Semantic Activity Learner (SAL)
- Atomic Actions (AA)
- Atomic Action Learner (AAL)
- Action Primitives (AP)
- Proprioception (PP)

The *Semantic Activity Planner (SAP)* consists of a sequence of steps to be performed to achieve a certain task as requested by the *Cognition Center*. For instance, when the task is to ”Bring an apple to the user”, the SAP generates a sequence of subtasks like

1. Goto the kitchen
2. Find an apple
3. Grasp the detected apple
4. Goto initial location
5. Localize the user
6. Handover it to the user

All these sub-steps that the robot is capable of performing are stored in the *Action sequence memory*. The SAP derives its name as it contains appropriate sequence of subtasks which are semantically compliant with human users. The SAP can also expand the range of tasks it can do by interacting with its *Users*. This is achieved through the *Semantic Action learner* module which updates the set of subtasks and sequences in the *Action sequence memory*.

Every subtask is an *Atomic Action* like Grasp, Navigate, Handover, etc. These *Actions* interact with modules of perception that can include detection, recognition, tracking, localization, etc and control the actuators to achieve a given subtask. These actions can also be taught to the robot by the user through *Action learners*. These can involve various learning strategies like Reinforcement learning (RL), Learning by demonstration (LbD) etc.

The perception modules when initiated by these action modules perform in *Top-Down* perception modes, where the robot knows the kind of patterns that it is looking for in the sensory information stream.

Finally, the atomic actions initiate *Action primitives* which comprise of basic operations that can be performed by actuators like inverse kinematics, differential drives, text to speech etc. The actuators are directly controlled by these to interact with the *Users* and *Environment*

The internal state of the robot which includes the knowledge about relative joint locations, orientation with respect to the world, sensor-actuator calibration, power state etc are continuously estimated while actions are being performed. This is known as *Proprioception* and this estimates the *Robot state*.

1.5.4. Cognition center

At the core of the robots computational mechanism, lies the cognition center which uses the *World state, User state and Robot state* generated by *Perception* and *Action* modules to take decisions on the kind of behaviour the robot should exhibit.

The robots internal state and the external world state are collected by the *State aggregator* and stored in the short term memory, which is transferred into long term memory depending on frequency of use of particular information. This can be also influenced by decision engine based on user interaction.

This architecture comprises of modules for both perceptual, motor and cognitive capabilities of the robot as all of these are required for a completely functioning

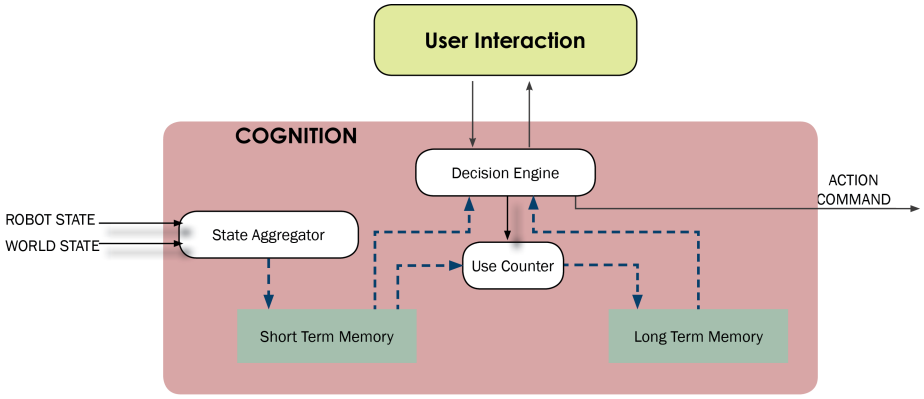


Figure 1.9: Cognition module of human centric robot software architecture

robot. It is proposed as a framework through which various algorithms can be integrated to have robots capable of peer-to-peer collaboration with humans. Naturally the scope of this architecture is very broad and applicable to any kind of robot.

The focus of this thesis lies in developing certain human centric *Perceptual* capabilities in a Personal service robot with contributions to semantic and spatial discernments as explained in Section 1.4.

1.6. Structure of Thesis

In a specific scenario of service robots operating in domestic conditions, a vast majority of its tasks involve either perceiving or acting upon objects and interacting with humans. It may be the case of searching and bringing an object that user desires, or remind the user of certain objects, or perform actions involving few different objects. For example: Picking and bringing an object requested by the user, cleaning up by depositing objects at their respective usual locations, interacting with two or more objects to learn/perform a complex task (such as cutting a vegetable, or flipping a toast). Hence object recognition can be a pertinent and readily tested use case for developing human centric perception, in this case focussed on service robots. This will be the focus of this thesis.

In summary, we would like to have the following characteristics embedded in the object perception capability of our robot

1. An intuitive (non verbal) method for humans to communicate their object of interest to a robot
2. Human centric recognition, where an object is recognized in terms of semantic concepts familiar to humans
3. Incorporate different modalities of information available to the robot to obtain a robust recognition performance to handle unconstrained human environments

We achieve these capabilities through a combination of hardware and software modules which are described further in this thesis.

A physical robot

A physical robot is necessary to embody different algorithms that make it capable of performing various tasks of a domestic service robot. A complete robot is developed in collaboration with technicians and co-researchers, that can be used to analyze and tackle the challenges of various other modules such as facial recognition, natural language understanding, navigation, manipulation, human robot interaction, etc in most realistic scenarios. The description of our robot LEA (Figure 2.6) which has been developed with the constraints of affordability and usability by humans is presented in Chapter 2.

Intuitive visual interaction

For robots to be easily used and interacted with normal users, an intuitive way of human robot interaction is needed. There are many contexts in tasks of a personal robot such as

- Picking and bringing a object requested by the user
- Cleaning up by depositing objects at their respective usual locations
- Interacting with two or more objects to learn/perform a complex task (such as cutting a vegetable, or flipping a toast) etc.

where the robot needs to localize objects which the user intends to communicate with the robot. This is the first stage in the learning pipeline where "Spatial discernment" or the structures learned only from visual data of environment (Bottom Up segmentation cues) are combined with Top-Down cues provided by user intent to segment the user's object of interest by the robot. This provides an effective *spatial grounding* in human robot interaction. Such a system which integrates Salient regions from a color image, structure segmentation from a depth image with human cues such as Pointing and Eye-gaze to perform reliable segmentation in various configurations of user-robot positions is described in Chapter 3. Subsequently, we proceed towards *Semantic discernment* with respect to understanding objects.

Semantic object recognition

Despite the availability of unlimited memory (cloud storage) and processing power, it is not possible and required to equip a service robot with ability to recognize all possible objects. Even in case of humans, every individual develops perceptual knowledge based on the kinds of objects they encounter in their environments and also from interactions from other people. But the ability to discern certain semantic characteristics of objects and ability to continuously update and expand their knowledge with experience is essential. A semantic recognition framework that abstracts concepts such as color, shape from visual data and combines that with robots location (from navigation) to understand an object is presented in Chapter 4.

Figure 1.10: Changing appearances with different viewpoints

This framework is made modular in order to integrate different modalities, and also to perform recognition even in absence of some modalities of information. With this an object eg "Apple" can be recognized as *"Red coloured object, mostly round in shape, present in kitchen or living room*.

Multiple view recognition

The recognition of objects in a unconstrained every-day human environments poses a challenge due to change in lighting conditions, orientation of the objects, viewpoint of the object seen by the robot etc. An object can appear completely different based on where it is viewed from as can be seen in the Figure 1.10. Hence a multiple view based recognition approach is used. This is performed in two steps.

1. Tracking
2. Multi-view recognition

Chapter 5 presents a novel tracking algorithm to continuously localize an object while being explored from different views by a robot or while a user demonstrates different viewpoints of an object to the robot. The developed tracking algorithm uses features that are used in object recognition as in Chapter 4 along with depth data to track complete object contours while handling challenging motions such as out of plane rotation and scale change.

With the ability to explore an object from different viewpoints, a multiple view based recognition system can be used to enhance the robustness. A novel multi-view recognition system that captures the variation of visual characteristics with respect to 3D motion between views is developed based on *Sequence Alignment* techniques borrowed from BioInformatics is described in detail in Chapter 6.

Novel action detection

Apart from visual concepts, objects can also be described based on their affordances [45] which can be integrated to augment the learning process. For example, an apple is generally used for eating, while a detergent is used for cleaning and a book is used for reading, etc. Some of these *affordances*, like their utility/function, can be learnt from actions performed by humans over the objects. The ability to recognize human actions becomes essential and since every user has a different way and diverse set of actions, we need a system to incrementally learn novel actions performed by users. Detection of new actions is the first step towards this process and we have developed a novel method to detect *"unknown actions"* which is presented in Chapter 7. With this it will be possible in future to describe an "Apple" as "Red, round object, mostly present in kitchen or living room and used for eating".

Various algorithms developed in this thesis are modular and compatible to be integrated together in an architecture presented in Figure 1.6. This provides freedom for different modalities such as natural language understanding, complete activity recognition, etc. to be incorporated into this system which can continuously evolve from experience and interaction with human users due to the presence of multiple learning loops.

References

[1] *Mx 3d bridge,* http://mx3d.com/projects/bridge/.

[2] *Bigdog, boston dynamics,* http://www.bostondynamics.com/robot_bigdog.html.

[3] *Astronaut, milking robots from lely,* http://www.lely.com/en/milking/robotic-milking-system/astronaut-a4_0.

[4] *Neato robotic vacuum cleaners,* https://www.neatorobotics.com/robot-vacuum/botvac-d-series/.

[5] *Baxter robots,* http://www.rethinkrobotics.com/baxter/.

[6] *Tesla, autopilot technology,* https://www.teslamotors.com/blog/your-autopilot-has-arrived.

[7] *Amazon prime air,* http://www.amazon.com/b?node=8037720011.

[8] *Fetch robotics, automated warehousing,* http://fetchrobotics.com/fetch-core/.

[9] *Double, tele presence robots,* http://www.doublerobotics.com/.

[10] J. Schmidhuber, *Deep learning in neural networks: An overview,* CoRR **abs/1404.7828** (2014).

[11] L. E. Kavraki and S. M. LaValle, *Motion planning,* in *Handbook of Robotics* (Springer, 2007).

[12] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)* (The MIT Press, 2005).

[13] *Robots as tools or companions,* http://www.livescience.com/27204-human-robot-relationships-turkle.html.

[14] *Projected demographic shifts in developed countries,* http://www.prb.org/publications/datasheets/2011/world-population-data-sheet/germany.aspx.

[15] *Supporting independent living for the elderly through robotics,* http://www.silverpcp.eu/.

[16] *Robotic devices for nursing care project,* http://robotcare.jp/?lang=en.

[17] *Robot care systems b.v,* http://www.robotcaresystems.com/.

[18] *Chihira aiko, android robot from toshiba,* http://www.reuters.com/article/us-japan-robot-store-idUSKBN0NB1OZ20150420.

[19] *Palro, humanoid care robot by fujisoft,* https://palro.jp/.

[20] *Hsr care robot from toyota,* http://www.toyota-global.com/innovation/partner_robot/family_2.html.

[21] *Collaborative industrial robots,* http://www.robotics.org/content-detail.cfm/Industrial-Robotics-Industry-Insights/The-End-of-Separation-Man-and-Robot.

[22] *Factory in a day project,* http://www.factory-in-a-day.eu/.

[23] S. Edwards and C. Lewis, *Ros-industrial: applying the robot operating system (ros) to industrial applications,* in *IEEE Int. Conference on Robotics and Automation, ECHORD Workshop* (2012).

[24] *Reasons for accidents in self driving google cars,* http://www.wsj.com/articles/google-reports-13-near-miss-incidents.

[25] T. B. Sheridan and W. L. Verplank, *Human and computer control of undersea teleoperators (Man-Machine Systems Laboratory Report),* (1978).

[26] *Spacex,* http://www.spacex.com/.

[27] Y. Harari, *Sapiens: A Brief History of Humankind* (Harvill Secker, 2014).

[28] P. M. Fitts, *Human engineering for an effective air-navigation and traffic-control system.* (1951).

[29] M. A. Hearst, *Trends & controversies: Mixed-initiative interaction.* IEEE Intelligent Systems **14**, 14 (1999).

[30] M. A. Goodrich and A. C. Schultz, *Human-robot interaction: A survey,* Found. Trends Hum.-Comput. Interact. **1**, 203 (2007).

[31] M. C. Tacca, *Commonalities between perception and cognition,* Linking Perception and Cognition , 7.

[32] G. V. B. K. Hugenberg, *Social cognition: The basis of human interaction,* (Taylor and Francis, 2011) Chap. Attention, perception, and social cognition, pp. 1–22.

[33] M. C. Tacca, *Commonalities between perception and cognition,* in *Frontiers in Psychology* (2011).

[34] G. Lupyan, *The centrality of language in human cognition,* Language Learning , n/a (2015).

[35] M. Wertheimer, *Laws of organization in perceptual forms* (Harcourt, Brace & Jovanovitch, London, 1938).

[36] E. Durkheim and M. Mauss, *Primitive Classification* (University of Chicago Press, 1963).

[37] M. Pagel, *Wired for Culture: Origins of the Human Social Mind* (W. W. Norton, 2012).

[38] M. Rudinac, *Exploration and Learning for Cognitive Robots*, Ph.D. thesis, Delft University of Technology (2013).

[39] I. Arel, D. Rose, and R. Coop, *Destin: A scalable deep learning architecture with application to high-dimensional robust pattern recognition,* in *in Proc. of the AAAI 2009 Fall Symposium on Biologically Inspired Cognitive Architectures (BICA* (2009).

[40] F. Bergmann and B. Fenton, *Artificial general intelligence: 8th international conference, agi 2015, agi 2015, berlin, germany, july 22-25, 2015, proceedings,* (Springer International Publishing, Cham, 2015) Chap. Scene Based Reasoning, pp. 25–34.

[41] A. Boularias, F. Duvallet, J. Oh, and A. Stentz, *Grounding spatial relations for outdoor robot navigation.* in *ICRA* (IEEE, 2015) pp. 1976–1982.

[42] S. Guadarrama, L. Riano, D. Golland, D. Göhring, Y. Jia, D. Klein, P. Abbeel, and T. Darrell, *Grounding spatial relations for human-robot interaction,* in *IEEE/RSJ International Conference on Intelligent Robots and Systems* (2013).

[43] A. Karpathy and F. Li, *Deep visual-semantic alignments for generating image descriptions,* CoRR **abs/1412.2306** (2014).

[44] Y. Salaün, E. Vincent, N. Bertin, N. Souviraa-Labastie, X. Jaureguiberry, D. T. Tran, and F. Bimbot, *The flexible audio source separation toolbox version 2.0,* in *ICASSP* (2014).

[45] J. J. Gibson, *The theory of affordances,* Hilldale, USA (1977).

2

Design of LEA: Second generation Delft personal service robot

2.1. Introduction

The first step in the direction of an ideal robot with peer-to-peer collaborative capabilities with humans is to develop a hardware infrastructure that can embody various algorithms to form a complete robot. As discussed in earlier Chapter (1) we focus on developing a Personal service robot that can perform various tasks in a domestic household environment. This poses a challenge to design a robot which apart from performing its required tasks, needs to be appealing for layman users to interact with it. Also events like Robocup@Home [1] provide avenues for benchmarking and evaluating functionalities in most realistic environments.

Currently, the most advanced robot is Asimo by Honda [2]. It is a very complex and heavy humanoid platform with 57 degrees of freedom (DOF), that can autonomously walk, jump and run both forward and backwards and manipulate small objects. However, due to its complexity it can mainly execute preprogrammed tasks and has a cost of over million Euro. Robots like Nao [3] are very good experimental platforms, but not suitable to be deployed in assistive tasks because of their limited mechanical reach and data processing capabilities. While walking bipedal robots

---

Chapter modified from articles:
Aswin Chandarr, Machiel Bruinink, Floris Gaisser, Maja Rudinac, and Pieter Jonker: Towards bringing service robots to households: Robby, Lea smart affordable interactive robots, IEEE/RSJ International Conference on Advanced Robotics (ICAR 2013), Workshop on General Purpose Service Robots, Montevideo 2013.
Machiel Bruinink, Maja Rudinac, Floris Gaisser, Aswin Chandarr, Guus Liqui Lung, Balint Szollosi-Nagy, Dennis Kaandorp, Jose Torres Mata, Martijn Wisse and Pieter Jonker: Delft Personal Robotics RocKIn@Home 2014, Team description paper.

are most ideal to work in domestic environments designed for humans, the complexity involved with basic locomotion is still computationally very intensive and are not reliable in unconstrained circumstances. Hence an intrinsically stable wheeled base is preferred so as to focus on higher level intelligence.

There exist various wheeled service robot platforms such as ARMAR [4], PR2 [5], Reem [6], care-o-bot [7], Cosero [8], etc as seen in Figure 2.1. The ARMAR-3 is a 43 DoF humanoid robot on an omni-directional wheeled platform. The robot's hardware and software were designed to embody and evaluate different aspects of its cognitive architecture and equipped the robot with various autonomous functionalities. But the increased DoFs introduce mechanical and control complexity and challenge affordability that we aim to achieve. The PR2 robot from Willow garage [5] is a commercial research platform with 27DOFs capable of performing safely and autonomously many advanced household tasks. However it's main drawback is its high cost of over half a million Euro as well as redundant DOFs for most tasks. This is also the case with other commercially available robots ([6], [7]). Cosero [8], winner of Robocup@Home Competition, has a similar configuration (23DOF) and advanced capabilities with a slightly lower costs, but this is not available to be used as a research platform.

(a) ARMAR, Care-o-bot3 and Cosero (b) PR2 and REEM

Figure 2.1: Popular wheeled service robots

To summarize, most robots developed so far have a very complex body design. Apart from suffering from decreased battery life, managing and programming their complex structure makes them too expensive for care and home market. In our research we follow a different approach and focus on building affordable robots with minimal sensor suite and a simplistic mechanical design that are still capable of performing all the required tasks. This is achieved while providing a socially appealing design for general public. The focus lies in resolving the limitations posed by inexpensive hardware through advanced and adaptive vision algorithms.

The research in this direction started with development of Robby, the first generation Delft Personal service robot (DPR) as described in the PhD dissertation of Maja Rudinac [9] with only 6 Degrees of Freedom (DoF). The capabilities, reliability and user interface have been improved to develop the second generation DPR named LEA with 9 DoF, which is presented in this thesis. These robots are shown in

Figure 2.2. The robot is developed completely in the lab [1] in order to have complete control of the robot and provide flexibility in integrating the components together. This chapter further details the mechanical, electronic and software backbone of this robot.

Figure 2.2: Robby (right) and Lea (left), 2 Generations of Delft personal service robots

2.2. Affordable mechanics

The robot consists of 4 main components

- **Mobile base and Torso**: A differential drive platform with two motorized wheels in the front and two supporting castor wheels in the rear is used as base on which the robot is built. A pair of 24V/3.6A geared DC motors is used to power the front wheels. The base platform is made generic with a mounting holes of various sizes in order to allow for flexibility to build any structure on top of it for robots of different applications. In this case, a *Torso* that can support the arm, neck and head is constructed on the front side of the base with a set of square aluminium tubes. The base also houses 4 Lead acid batteries (12V, 7Ah) to power the motors and the central computer. These batteries placed in the rear also act as a counterweight to balance the torso to prevent tipping over of the robot in normal operating conditions.

- **Arm and hand**: A 4 DoF arm with a shoulder, elbow and 2DoF wrist is constructed on the right side of the torso for object manipulation and user interaction. The arm is built from light weight materials (Acrylic) inorder to reduce the torque required from the motors. This is made possible due to a smart mechanical design to place the heavier, higher torque motors for shoulder and elbow and 1 wrist joint within the torso. These motors are coupled to the joints through a series of belt driven mechanisms as shown in Figure 2.3. The arm also comprises of a virtual four bar linkage through these belts. This allows to use low power motors at the wrist joint which only position the joint

[1] Delft Biorobotics Laboratory

at a required orientation, while the linkage maintains this configuration during the motion of elbow and shoulder joints. All these designs make it possible to use smaller motors, light weight materials while still allowing the robot to easily lift objects till 0.5Kg which is generally sufficient in households.

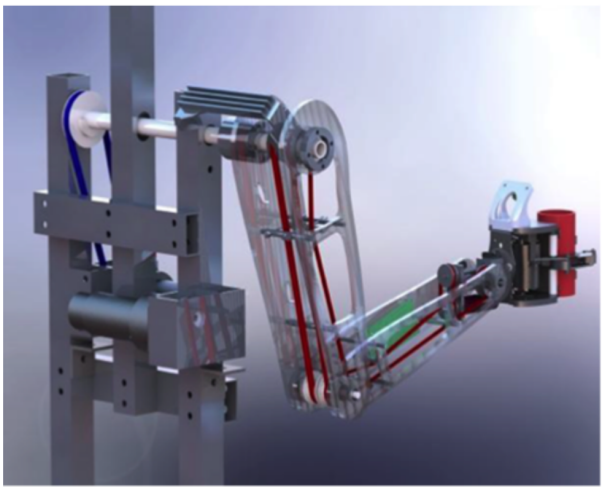


Figure 2.3: 4DoF Lea Arm design

- **Gripper**: The final object manipulation is performed by an underactuated gripper that is placed at the wrist of the arm. This is a force controlled gripper which has the ability to adapt its fingers around objects of different shapes and sizes [10] which is designed in the same laboratory. It consists of a 6 fold differential mechanism and the fingers wrap around the object until the forces are balanced. There is also an infrared sensor that serves as touch sense. Such gripper allows to perform grasping of unknown objects of various shapes and materials, since it does not require prior knowledge on grasping configuration and suitable force to grasp objects. This gripper can be seen in Figure 2.4

Figure 2.4: Underactuated gripper

- **Neck and head**: A 3D printed head structure which can house various sensors and electronics (2.3) is developed based on the appearance of its predecessor Robby [11]. The head is attached to the torso through a pan-tilt neck (2DoF). The neck had a mechanical design challenge to provide sufficient torque and speed for head motion while maintaining a compact form factor. The design [2] comprises of vertically oriented motors for each DoF. The tilt motion is achieved through a *worm gear* mechanism which prevents the head from facing down when the motors are not powered. The pan motion is achieved by a single geared DC motor whose shaft is attached to the robot torso. The body of this motor is the moving part which is coupled to the tilt mechanism through a set of bearings. The design can be seen in Figure 2.5.

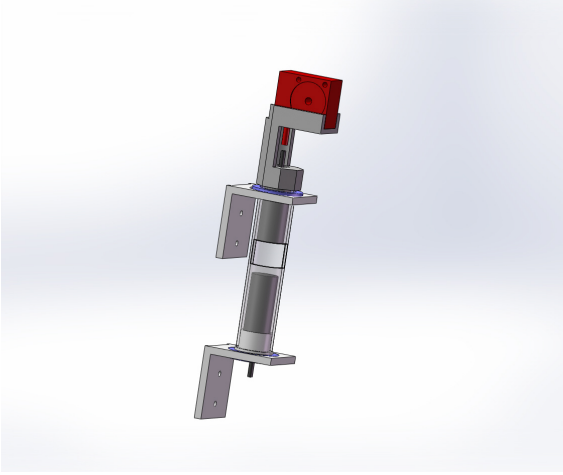


Figure 2.5: Lea Neck design

2.3. Sensor suite and Electronics

The robot is equipped with a minimal number of sensors, reducing the overall cost, while still providing sufficient information about the world to perform tasks in an autonomous/semi-autonomous manner. The electronics suite of LEA consists of the following components and the placement of the components on the robot is illustrated in Figure 2.6

1. **Microphone**: Highly sensitive directional microphone to filter background noise (Shot- gun Audiotechnica AT-8035)

2. **RGBD Camera**: Disguised as mouth of the robot, provides distance information as Pointclouds apart from color images needed for all vision processing (Microsoft Kinect)

[2] Thanks to Jan van Frankenhuyzen

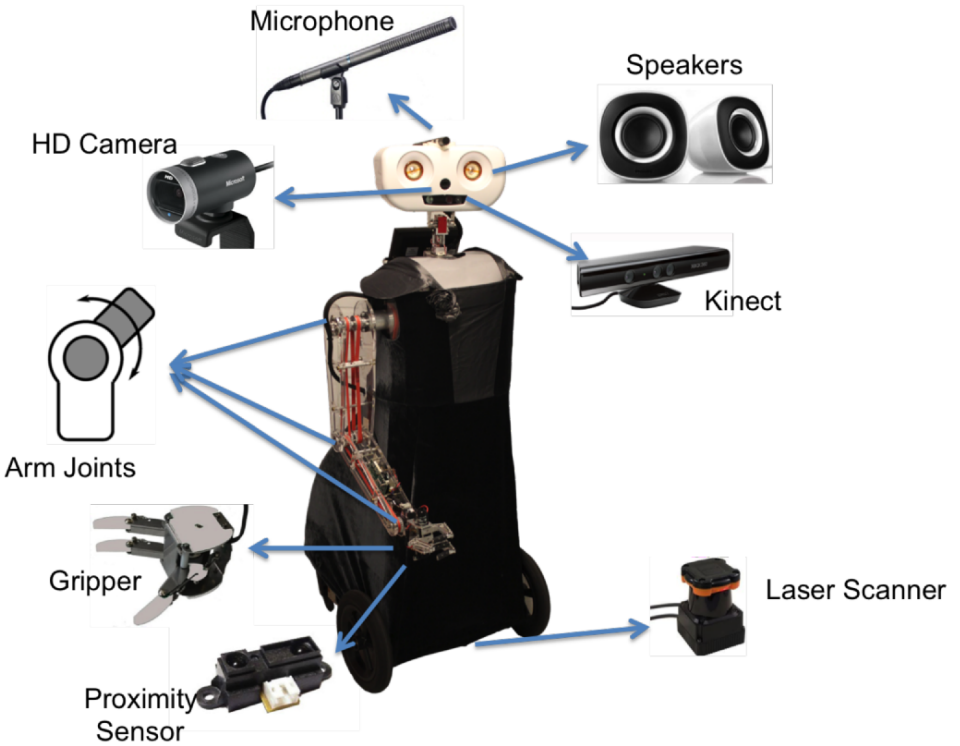


Figure 2.6: Sensor placement on robot LEA

3. **HD Camera**: Calibrated in combination with RGBD camera provides high quality images needed for specific vision algorithms (Microsoft Lifecam Cinema)

4. **Speakers**: Disguised as eyes of the robot, along with additional amplification modules, enables user communication in noisy environments. (Philips-SPA2201)

5. **RGB lights**: Embedded around the neck to provide user interaction with varying colors and frequencies

6. **IR proximity sensor**: Placed near the gripper, provides a sense of touch while manipulating objects and handing over objects with human users (SHARP 0A41SK F15)

7. **Planar Laser scanner**: Hidden under the ”dress” of robot, provides information for localization and path planning (HOKUYO URG - 04LX)

8. **Ultrasonic Rangers**: Embedded into the ”dress” in the rear of robot for local obstacle avoidance (SRF 02)

9. **Encoders**: Incremental encoders are placed in joint and wheels to provide proprioceptive data for robot state estimation

10. **Touch screen display**: Placed in the back of LEA, provides a high level GUI for choosing different operating modes of the robot. It also shows the outputs of intermediate visual processing being performed to the user.

11. **Motor Controllers**: All the motors in the robot are controlled by **3MxL**, a dual axis motor controller board developed within the lab [3]. These are smart motor controllers providing safety with *Over current protection, Heat limits, Protocol timeout detection* etc. They allow local PID controller at 1Khz for speed, position, torque and series elastic control besides offering flexibility for sophisticated model based control algorithms. The communication is based on *Dynamixel* (**todo: cite 3mxl or dynamixel**) protocol and hence many boards can be daisy chained to allow control of multiple motors through a single USB port at a high frequency. These are used to control the wheeled base, arm, gripper and neck with high speed and precision.

12. **Central Computer**: A single computer is used to integrate all sensory information and control the motors apart from decision making and user interface. A PC with an Intel i7-3770 Processor with 8M Cache, 3.9Ghz along with 8GB RAM is placed on the base between the torso. This communicates to all the peripherals through high speed USB 2.0 connection.

2.4. User friendly design

A considerable effort has been directed towards the appearance of LEA to make it socially appealing and give it a human friendly feeling. Human like features have been imbibed into the design by disguising different electronic components as eyes (speakers), nose (HD camera) and mouth (RGBD camera) in the head and torso covered by a ”velvet dress” that has been exclusively stitched for this robot. The aesthetic appeal has also been enhanced by using gradual curves in various parts of the robot and all the wires and electronics that are concealed. It has been taken care not to cross the *”Uncanny valley”* of human acceptance of human like artificial structures. The decision to leave the robot arm uncovered to expose the internal mechanics can be attributed to this. The public likeness of this robot could be gauged from its popularity in different public demonstrations and in print media (Figure 2.7) [12], [13] , [14] and [15]. LEA has also featured in a Discovery Channel documentary on Robocare [16] and even in a popular music video set in a future where robots are present in everyday life [17]

2.5. Software architecture

Intelligent software is necessary to transform an assembly of metallic components into a functional robot. LEA was initially designed to participate in RoboCup@Home competition and to evaluate its functionalities in a household setting. This requires

[3] 3MxL’s are developed by Guus Liqui Lung of Delft Biorobotics Labaratory

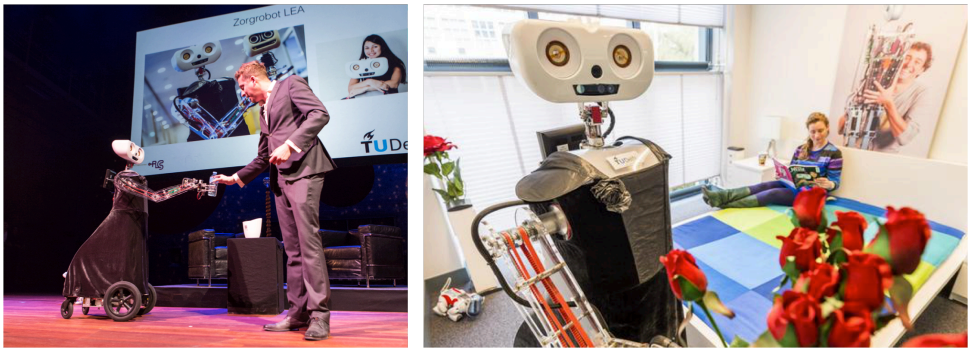


Figure 2.7: LEA in print media. Left: NewScientist [12], Right: de Volkskrant [13]

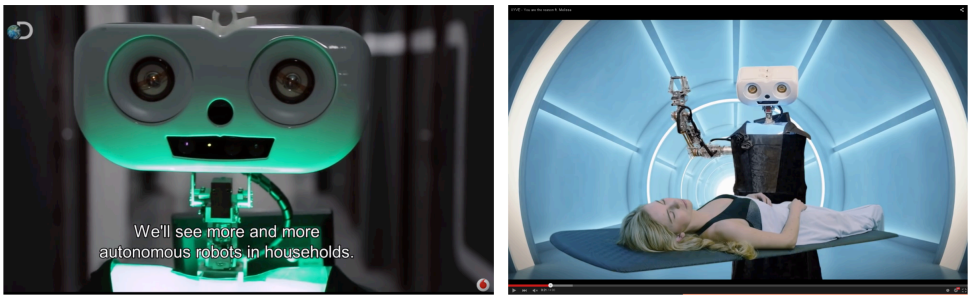


Figure 2.8: Public videos with LEA Left: Discovery Channel (Firestarters) [16], Right: XYVE music video [17]

an adequate software system inorder to accomplish its tasks. A modular architecture with loosely connected modules is developed for integrating various components of the robot. This section describes this basic software architecture, which forms the basis for the Human centric robot architecture proposed in Chapter 1.

The architecture has been divided into 2 layers the execution-layer and the robot-control-layer. The execution-layer is divided into two sub-layers the core and subcores. The robot-control-layer is also divided into two sub-layers individual modules and low-level control. All the layers of the software archi- tecture are depicted in Figure 2.9. Every layer in the system is designed to be modular and has its own specific task, allowing any part of the system to be changed or updated without affecting the other parts of the system unless the functionality of such a module has been modified. The software architecture has been designed to work with Robot Operating System (ROS) [18]. ROS provides many existing packages and drivers built specifically for robots and it provides a client-server communication interface for passing messages between different nodes running on the robot.

2.5.1. Core

The core acts as the brain of the robot, it is responsible for making the decisions and planning the requested task. The robot receives the task commands from the

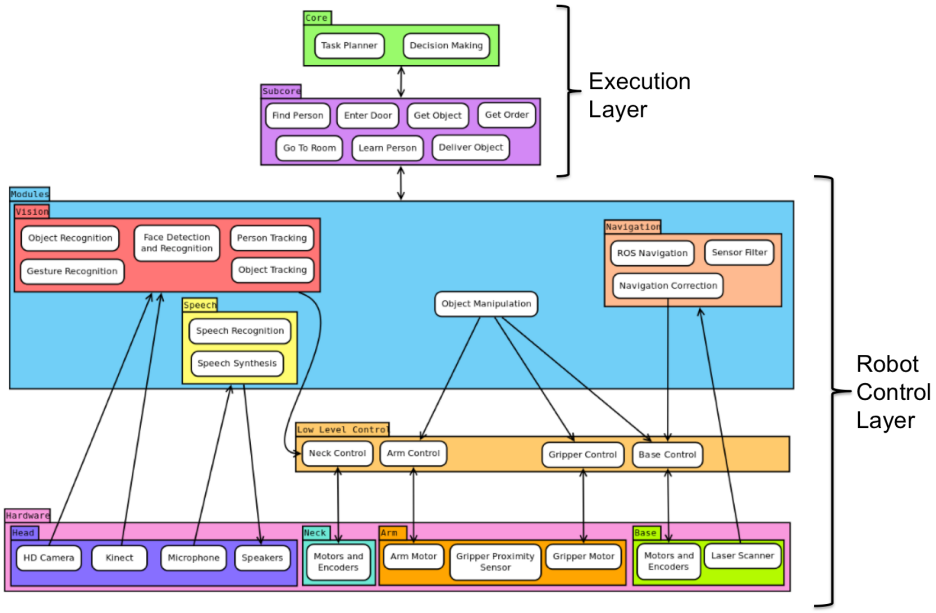


Figure 2.9: Software architecture for tasks at Robocup@Home

user through speech. When the robot receives a task from the user the robot will formulate a list of planned steps in a specific order to complete the required task. The robot must also know what its current state is in order to further formulate the steps. A state machine has been implemented to keep track of the robots current state and to initiate the subcores to perform the planned steps. Once a step is completed every module will inform the core so that it can initiate the next step. Once all steps have been completed the core informs the user of the task being completed and waits for the next user command. This has the functionality of *Semantic Activity Planner* (section 1.5). The communication between core and subcores is implemented using ROS *Topics*. All the subcores listen to these messages published by the core. The message contains an *ID* of the subcore being addressed and only the subcore corresponding to the required *ID* will respond to it. This message also has an *action state* which can be either ”Start” or ”Stop” for a subcore. There is a provision for many other parameters that provide information for the task to be performed. This can include details like ”information on the room it has to reach or the person that robot needs to find, or the object that needs to be delivered”. Currently, the core is programmed to perform a set of tasks needed in RoboCup@Home, but has the flexibility to incorporate new task sequences that can be learned from users.

2.5.2. Subcore

A subcore is a sequence of steps that are reusable for different tasks. For example, "finding a person" subcore can be used in the tasks involving bringing a person a drink or also finding a person in emergency situations. Each subcore has been created as generic as possible so that it can be used as part of many different complex tasks. A subcore receives the inputs from the core and it initiates the individual modules and provides feedback to the core. Many fallback options and timeout limitations are present to avoid the robot getting stuck in any individual state. This corresponds to the *Atomic Actions* component of section 1.5.

When a subcore completes its task, a response is sent to the core using a ROS "Topic" which contains the *ID* of the subcore responding and information about the outcome of the given task (If it was successful or some failure). This is essential for the progressing of the state machine inside the core. The messages from subcores can also contain information on data acquired such as "Name of users, objects or locations, etc."

2.5.3. Individual Modules

Individual modules are responsible for the main actions of the robot. These include navigation, manipulation, speech recognition, face recognition, object recognition and tracking as well as person tracking. Each individual module directly interacts with the low-level control layer. Many of the individual modules are implemented as ROS *Action Servers*, which provide interfaces for "Starting, Stopping, Pre-empting" the running process while also providing continuous feedback on the internal state of a particular task. The intelligence in these *Individual modules* allow us to handle the limitations arising due to affordable hardware and simplistic mechanics apart from the challenge posed by dynamic unconstrained environments. These include realtime face recognition with a high performance using CAPCA and online face learning using CertKNN [19]. Person following is achieved by using a combination of 2D HoG + 3D person detector [20] with fast tracking by using a flood fill tracker on depth data [21], augmented by a color and a motion model in a TLD framework [22]. This tracking is combined with a local navigation module to allow for safe navigation in cluttered environments and natural human activities. Reliable object manipulation is achieved through closed loop grasping using visual-servo control. The target object is continuously tracked while inverse kinematics is applied to decrease the distance error between the end-effector and the object until the object presence within the gripper is detected by its in hand IR sensor. This is an extended version from the earlier work [23] User interaction is achieved using a Frame Based Dialogue system [24] which uses speech recognition with a probabilistic grammatical parser to initiate voice interaction with users to confirm its understanding and also request missing information. The robot can also learn new actions from human demonstrations without explicit programming with a Learning from Demonstration (LfD) framework [25]. Many of these capabilities of the robot are possible due to the efforts of members of the Delft Personal Robot Project [26]

2.5.4. Low-level layer

The Low-level control layer consists of drivers for the robot hardware such as wheels, gripper, arm, cameras and distance sensors. This layer facilitates the interaction of the individual modules with the hardware. These drivers interact with the higher level *individual modules* mostly through asynchronous messaging (*Topics*) and in some cases, with "text to speech", they are implemented with a blocking *Service call* in ROS.

Such an architecture is deployed to complete the loop of perceiving the environment and users, identifying the task needed to be performed and finally acting upon the world. The system is designed to continue working (with decreased functionality) even when some of the individual modules fail and is also made modular at every level to make it readily possible to add a "new combination of tasks in a core or new functionality in a subcore or new hardware interfaces in the low level layer", without influencing existing capabilities of the system.

2.6. Conclusion

In this chapter design and functionalities of an affordable service robot with minimal degrees of freedom is presented. The mechanics was custom developed with smart designs leading to a decreased complexity and weight of the entire robot. Control electronics are custom designed to provide flexibility in mechanical dimensions and control algorithm requirements. All the mechanical, electrical and electronic components are well integrated with a socially appealing industrial design which has been well received by the public and media. The limitations arising from simplified mechanics and affordable hardware are compensated by advanced/adaptive vision algorithms to achieve the required functionalities of a service robot. Finally, all the hardware and individual software modules are integrated through a flexible software architecture that also allows for easy expansion of the robot's capabilities.

References

[1] *Robocup@home,* `http://www.robocupathome.org/`.

[2] Y. Sakagami, R. Watanabe, C. Aoyama, S. Matsunaga, N. Higaki, and K. Fujimura, *The intelligent asimo: System overview and integration,* in *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, Vol. 3 (IEEE, 2002) pp. 2478–2483.

[3] *Nao, autonomous programmable humanoid robot,* `https://www.aldebaran.com/en/cool-robots/nao`.

[4] T. Asfour, K. Regenstein, P. Azad, J. Schröder, A. Bierbaum, N. Vahrenkamp, and R. Dillmann, *Armar-iii: An integrated humanoid platform for sensory-motor control,* in *Humanoid Robots, 2006 6th IEEE-RAS International Conference on* (IEEE, 2006) pp. 169–175.

[5] *Pr2 robotics research and development platform from willow garage,* `http://www.willowgarage.com/pages/pr2/overview`.

[6] *Reem by pal robotics,* http://pal-robotics.com/en/products/reem/.

[7] *Care-o-bot 3, fraunhofer ipa,* http://www.care-o-bot.de/en/care-o-bot-3.html.

[8] J. Stuckler, D. Dr□schel, K. Gr□ve, D. Holz, M. Schreiber, and S. Behnke:, *Nimbro @home 2012 team description,* in *In RoboCup 2012 @Home League team descriptions, Mexico City* (2012).

[9] M. Rudinac, *Exploration and Learning for Cognitive Robots*, Ph.D. thesis, Delft University of Technology (2013).

[10] G. A. Kragten, *Underactuated hands: fundamentals, performance analysis and design* (TU Delft, Delft University of Technology, 2011).

[11] M. Rudinac, *Exploration and Learning for Cognitive Robots*, Ph.D. thesis, Delft University of Technology (2013).

[12] *Newscientist, met robots is het goed samenwerken,* http://www.newscientist.nl/nieuws/met-robots-is-het-goed-samenwerken/ ().

[13] *de volkskrant, is technologie de oplossing voor de vergrijzing?* http://www.volkskrant.nl/opinie/is-technologie-de-oplossing-voor-de-vergrijzing~a4213414/ ().

[14] *Zorgvisie, 500 robots in je zorgorganisatie,* https://www.zorgvisie.nl/ICT/Verdieping/2014/11/500-robots-in-je-zorgorganisatie-1651204W/ ().

[15] *Elsevier, daar komen de robots!* http://www.elsevier.nl/juist/article/2014/10/daar-komen-de-robots-1613573W/ ().

[16] *Discovery channel, firestarters, robocare,* https://www.youtube.com/watch?v=kgS2tN2HWfI.

[17] *Xyve - you are the reason, music video with lea,* https://www.youtube.com/watch?v=LWPOfmbfucc.

[18] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, *Ros: an open-source robot operating system,* in *ICRA Workshop on Open Source Software* (2009).

[19] F. Gaisser, M. Rudinac, P. P. Jonker, and D. Tax, *Online face recognition and learning for cognitive robots,* in *Advanced Robotics (ICAR), 2013 16th International Conference on* (IEEE, 2013) pp. 1–9.

[20] M. Munaro, F. Basso, and E. Menegatti, *Tracking people within groups with rgb-d data,* in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on* (IEEE, 2012) pp. 2101–2107.

[21] H. Gaiser, *Efficient Person Tracking based on Depth data*, Ph.D. thesis, Technical University of Delft (2012).

[22] J. Van Egmond, *Robust object tracking with depth data*, Ph.D. thesis, TU Delft, Delft University of Technology (2014).

[23] A. Chandarr, *Towards bootstrapping robotic perception of indoor environments*, Ph.D. thesis, Master thesis, Delft University of Technology (2012).

[24] S. Rueda, *A speech-based dialog system for household robot*, Ph.D. thesis, Master thesis, Delft University of Technology (2012).

[25] C. Rozemuller, *Action learning from human demonstrations for personal robots*, Ph.D. thesis, TU Delft, Delft University of Technology (2013).

[26] *Delft personal robot project,* `http://www.roboned.nl/en/smart-design-affordable-service-robots`.

3

Multimodal Joint Visual Attention Model

In this chapter, we introduce a non-verbal multimodal joint visual attention model for human-robot interaction in household scenarios. Our model combines the bottom-up saliency and depth-based segmentation with the top-down cues such as pointing and gaze to detect the objects of interest according to the user. For generation of the top-down saliency maps, we have introduced novel methods for object saliency, based on the pointing direction as well as the gaze direction. For gaze estimation, a hybrid model has been introduced which automatically selects keypoint-based matching or backprojection based on the textureness of the object model. The combination of different cues ensures reliable object detection and interaction independently of the relative position between the user, robot and objects. Extensive experiments show good detection results in different interaction scenarios as well as in challenging environmental conditions.

3.1. Introduction

For efficient non-verbal human robot interaction, a joint visual attention model is indispensable for naturally communicating user intentions to robots. Joint visual attention represents two humans or a robot and a human having a shared attention to the same object. This model is important in scenarios such as learning of new objects or activities, or collaborative tasks between robots and humans. The main modes of non-verbal communication by humans include gestures and gaze as well as natural cues such as salient colours and objects. We combine these cues to

Chapter modified from article: Joris Domhof, Aswin Chandarr, Maja Rudinac, Pieter P. Jonker: Multimodal joint visual attention model for natural human-robot interaction in domestic environments, IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2015, Hamburg

obtain a model of shared attention in our work, as can be seen in Figure 3.1, to achieve a natural human-robot communication.

Figure 3.1: From left to right: the view from the robot, combined saliency map and the found object of interest.

Bottom-up saliency represents naturally interesting regions that are present in the visual field of view. The most salient regions have distinct properties of colours and shapes. The saliency based on colour images has been widely studied. The biologically inspired saliency model of Itti [7] estimates the saliency by considering the distinctive local features based on colour, intensity and orientation. Other methods such as [4] estimate saliency using spectral residual models. The saliency arising from shape can be obtained from the depth maps from RGBD sensors. [15] locate distinctive regions based on the local extrema of the depth histogram whereas methods such as [16] use the depth information with colour images to obtain a combined saliency map. These methods provide a good estimation of all interesting regions in the scene. But a definite point of user's attention cannot be estimated by the robot using bottom-up saliency alone as it depends on context and the emotional state of the user. All of the obtained salient regions are equally probable to the user.

Hence, top-down visual attention models have been used to constrain the robot's estimation of the region of interest. These obtain a computational model of a user's intention based on cues such as speech, pointing gestures and head orientation. [17] combines a spoken language model along with pointing direction and colour saliency to train a CRF model to estimate objects of a user's interest ([3], [5]). The work of [1] combines colour saliency with a relative pose of the user's head based on depth information to detect the required object. The user's head is located using standard face detection algorithms and the orientation is found by modelling a cylinder around the located head. Since the actual gaze direction does not correlate with the head pose, a Gaussian Process Regression is trained to estimate the viewing angle for every user. The work of [14] trains a Bayesian inference model to track the gaze of the user and estimate the most salient object based on it.

However, these methods are limited by using only colour information. In addition they require an extensive pre-training of the system which makes them unusable in novel environments. Finally, they are restricted to be functional only in situations when the robot is facing the user. This limits the use of existing methods for natural interaction with robots when the relative position between user and robot

changes. They also lack easy adaptability to various users and previously unknown environments.

Hence, we introduce a novel multimodal joint visual attention system which does not require any training routine and adapts automatically to change in user-robot configuration. We combine bottom-up saliency models based on colour and depth along with top-down saliency cues of pointing direction and gaze tracking to achieve this. The main contributions include a method of accurately estimating a saliency map based on 3D hand tracking and an adaptive method that estimates the saliency map based on gaze obtained from a head-mounted eye tracker. The adaptiveness ensures efficient saliency estimation for both textured and uniformly coloured objects. A novel combination of all different modalities has been introduced to robustly detect the object of user interest.

The methods have been implemented on our custom developed service robot LEA with objects in a table top scenario. Experiments have been performed to show the working of the algorithm when i) the robot is opposite to the user, ii) when the robot is adjacent to the user and iii) when the robot is in an intermediate position. Experimental results have shown a very good performance with the different combinations of the four modalities. Furthermore, the influence of different backgrounds, illumination conditions and variations in distance between the objects is evaluated and these experiments show a good performance.

The rest of the chapter is organized as follows. Section 3.2 explains the computation of saliency maps from different modalities, after which Section 3.3 elaborates on the integration of the obtained maps to detect the object of interest. Section 3.4 provides details of the different experimental conditions and also the specifications of the robot being used. Extensive experimentation and discussion are presented in Section 3.5 with conclusion and future work in Section 3.6.

3.2. Modalities

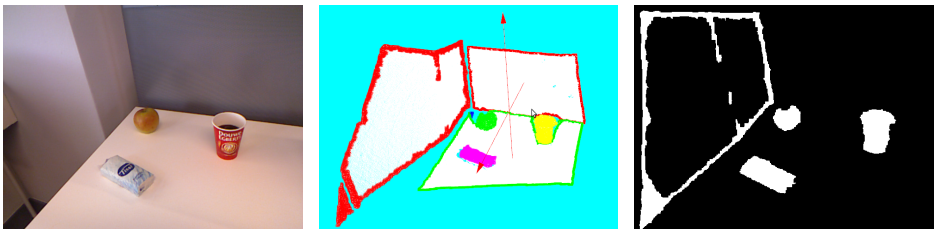
The non-verbal joint visual attention model that we propose combines bottom-up cues with top-down cues. It consists of the bottom-up cues: bottom-up saliency and depth. Furthermore, the attention is directed towards top-down salient regions that are indicated by pointing and eye-gazing. In this section, we describe how each of the modalities is computed.

3.2.1. 2D Saliency Map

Points of interest according to the user are most likely to be found in the most salient regions in the scene. Therefore we use the model of Itti, Koch and Niebur [7] to calculate bottom-up saliency. This model computes conspicuity maps by means of center-surround differences for each of the three features: colour, intensity and orientations. Combining these three conspicuity maps by averaging provides the bottom-up saliency map. However, this type of map gives several regions of interest.

3.2.2. Depth-Based Saliency Map

To restrict the number of probable object locations obtained by 2D saliency, we deploy depth-based segmentation. The first step consists of filtering the point cloud to find the region of interest. Then, the surface normals are estimated at every point in the point cloud. The Organized Multi-plane Segmentation algorithm of [8] is used to detect planes based on the normal vectors. After that, the clusters are extracted by Euclidean clustering of the remaining points after planar removal. These clusters are projected back using camera-intrinsic parameters to obtain a mask with the same dimensions of the 2D saliency map, as shown in Figure 3.2.



(a) Image (b) Point cloud (c) Depth mask

Figure 3.2: At the left hand side the image of the scene is shown and at the right hand side the image containing all objects.

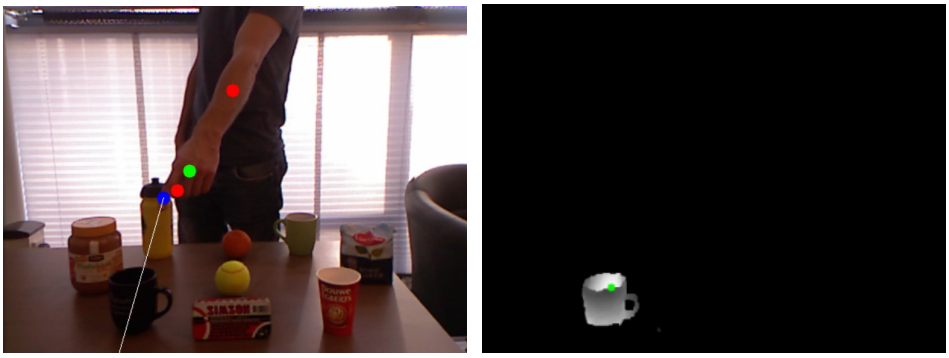
3.2.3. Pointing Map

A novel pointing-based saliency map has been developed. At first the estimated location of the hand is detected using the OpenNI-tracker [18]. After that, a more accurate and stable hand location is acquired by computing the centroid of the hand cloud, obtained by a conditional filter with spherical conditions (with a radius of 15 cm). The user arm is removed by a conditional filter with cylindrical conditions (with a radius of 25 cm) after which the palm of the hand is removed to identify the pointing finger point cloud. The eigenvector belonging to the largest eigenvalue corresponds to the pointing direction as used in [9].

Based on the pointing direction, the top-down saliency map is calculated as follows. At first, from depth-based segmentation, a point cloud with object clusters is obtained. For every point that belongs to an object, the vector from the position of the finger to this point is computed. The angle between this vector and the pointing direction vector is computed using

$$\cos\beta = \frac{x.d}{|x|.|d|} \tag{3.1}$$

where d is the pointing direction vector and x is the vector from the position of the finger to the object point. The value of the pointing map is related to the calculated angle using a normal distribution as in [3]. An illustration of 3D pointing vector detection and estimation of pointing-based saliency map is shown in Figure 3.3.



(a) Pointing subject (b) The pointing map

Figure 3.3: The subject is pointing at the blue coffee cup. At the left hand side the pointing direction is visualized by the line. The blue dot represent the location of the pointing finger. Furthermore the two red dots indicate the location of the elbow and the hand. The green dot in the left image shows the centroid of the hand point cloud, whereas the green dot in the right image shows the most salient point in the pointing map.

3.2.4. Gaze Map

Though pointing provides a good estimation of the top-down saliency for detecting the object, it is not available when the robot is adjacent to the user. In these cases, we use a saliency map based on gaze available from an eye tracker. We introduce a novel method to estimate the gaze-based saliency without using any camera calibration, to allow for free user movement. At first, the eye tracker provides the point of user attention in its head mounted camera frame. The object of interest is segmented using GrabCut [12] and the location of this object is now identified in the robot camera frame. We achieve this using a novel hybrid model that automatically selects a texture-based approach or a colour-based approach depending on textureness of the object.

Texture-based approach

Keypoints and descriptors are computed in the sub image around the gaze location and in the target image using a Speeded-Up Robust Features (SURF) Detector [10]. After that, the keypoints are matched using a fast KNN [11]. A Gaussian function with a σ which depends on the average sizes of the objects, is centered over every matched point in the camera frame. A final gaze-based saliency map is obtained by summing all Gaussian functions that are located at every keypoint in the target image:

$$G = \sum_{i=1}^{n} N(\mu_i, \sigma) \tag{3.2}$$

This process is shown in Figure 3.4. If the number of key point matches is smaller than 15, the model switches to the colour-based approach which is explained in the next section.

Figure 3.4: Gaze based saliency map obtained for a textured object. Here the user is looking at the right most object, marked with a green point

.

Colour-based approach

A histogram backprojection based approach is used to estimate the gaze map for uniformly coloured objects which do not have keypoints. A normalized histogram in the Hue, Saturation space is constructed for the segmented target. The gaze map in the robot camera frame is obtained as value of the histogram at the bin corresponding to the every pixel [13]. A histogram with 180 bins for Hue and 256 bins for Saturation is used.

The target is matched to a high resolution camera calibrated to the RGBD camera. The extrinsic and intrinsic parameters are used to obtain the final gaze map in RGBD frame to match the dimensionality of other saliency maps obtained earlier.

Once all modalities are defined, a novel hybrid integration model is used to obtain the final object of interest as described in following section.

3.3. Integration

The individual maps obtained from bottom-up and top-down cues can be seen in figure 3.5. They are combined into a single saliency map using a Hadamard product [5]. A Hadamard product for two matrices A and B with the same size is defined as

$$(A \circ B)_{i,j} = (A)_{i,j} \cdot (B)_{i,j} \tag{3.3}$$

The advantage of using the Hadamard product is that points that belong to the table and points outside the pointing direction cannot be considered salient, because these points have a zero value in the depth map and pointing map, respectively. In all cases, the bottom-up saliency map (S) and the depth map (D) are available, but the gaze (G) and the pointing (P) probability map depend on the configuration of the human, robot and object. If the human and robot are standing opposite of each other, in that case gaze is not available. In this case, the combined saliency map is obtained by:

Figure 3.5: In this figure the four saliency maps are showed. From left to right the images represent the Gaze map (G), Pointing map (P), Depth map (D) and bottom-up Saliency map (S).

$$(C)_{i,j} = (P)_{i,j} \cdot (S)_{i,j} \cdot (D)_{i,j} \tag{3.4}$$

If the human and robot are standing next to each other, the robot cannot analyse the pointing gesture of the human. This means that the pointing map is not available, and therefore the combined saliency map is equal to:

$$(C)_{i,j} = (G)_{i,j} \cdot (S)_{i,j} \cdot (D)_{i,j} \tag{3.5}$$

When all four maps are available, it is not recommended to point-wise multiply all four cues because the gaze map is not always reliable due to the accuracy of the eye tracker and the usage of two different cameras. Therefore these four maps are added, but only for points that are located in pointing direction. This means that the combined saliency map is generated by adding all maps and multiplying it by the mask $(P)_{i,j} > 0$:

$$(C)_{i,j} = ((S)_{i,j} + (D)_{i,j} + (P)_{i,j} + (G)_{i,j}) \cdot ((P)_{i,j} > 0) \tag{3.6}$$

3.3.1. Object of Interest

For every object in the scene, the point cloud has been extracted in section 3.2.2. The centroid of all objects is computed and transformed to pixel coordinates. The object cluster with the smallest euclidean distance from centroid to the location of the maximum value in the combined saliency map is the object of interest.

3.4. Experimental Setup

In order to cover different real life scenarios, the system is tested in three different configurations of user, robot and object. Furthermore, most common collaborative tasks between robot and human that could take place in household situations are limited to a table setting. Therefore we test our Joint Visual Attention (JVA) model in a table top scenario in which the objects are positioned on a table.

3.4.1. Sensors Specifications

During the test, a user is wearing the eye tracker while the robot is equipped with a Microsoft Kinect camera. On top of the Kinect sensor a HD camera is mounted.

The resolution of the Kinect sensor is 640×480 pixels at a frame rate of 30 fps. Furthermore, the Logitech camera, which is mounted on top of the Kinect, has a resolution of 1920×1080 pixels. The Kinect RGB camera and the HD camera are stereo calibrated using a chessboard pattern. The subject is wearing an eye tracker of Pupil Labs [19], which has a resolution of 640×360 pixels for the eye camera, while the world camera has a resolution of 1280×720. Both cameras record images at a frame rate of 30 fps. The location of all the cameras used in the experiments can be seen in the Figure 3.6.

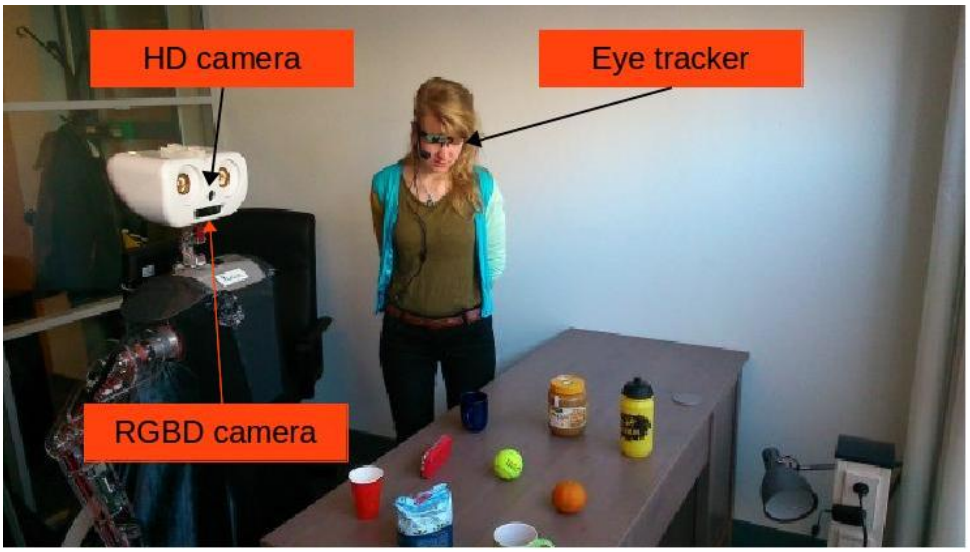


Figure 3.6: The experimental setup showing the different cameras on the user and the robot.

3.4.2. System Specifications

The system is tested on personal robot LEA, which is designed to assist humans with activities of daily living. It is a 9DOF mobile robot equipped with a set of sensors for autonomous navigation, manipulation of unknown objects, objects learning and recognition as well as user activity recognition. From all the sensors on the robot, the Microsoft Kinect and the HD camera are used. All software is running in C++ and is adapted for the Robot Operating System (ROS). All the software developed is available online[1].

3.4.3. Human-Robot Testing Configurations

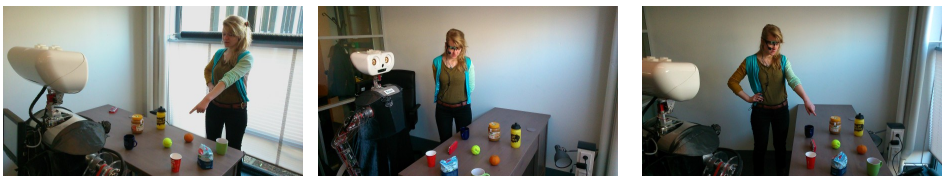
Mimicking ways how humans are interacting with each other, there are three ways how the human, robot and object are positioned. In the first case, the human and robot are standing opposite of each other and the object is located in between them.

[1] https://github.com/JDomhof/joint_visual_attention_model

This setup will be called 'opposite' and it is visualised in Figure 3.7a. In this case the human is in view of the camera of the robot, so the robot is able to determine where the human is pointing. However, the difference in viewing angle between the robot and the human is $180°$, so the view of the scene is totally different.

When the robot and human are standing next to each other, the configuration will be referred to as 'next to' (3.7b). In this case the human is not in view of the camera, it is not possible to determine the pointing direction, so the robot should be able to find the object of interest by combining the gaze with bottom-up saliency and depth.

In order to combine all four modalities, the human should be in view of the camera and the human and the robot should have approximately the same viewing angle. This means that the human, robot and object are positioned in a triangular shape, which is visualized in Figure 3.7c. Therefore it will be referred to as 'triangle'.



(a) 'opposite' (b) 'next to' (c) 'triangle'

Figure 3.7: The three setups.

3.5. Experimental Results

The joint visual attention model is tested in the three set-ups described in Section 3.4: the human and robot are standing opposite of each other, the human and robot are standing next to each other and the the human is looking and pointing to the side. The performance measure is the object detection rate. The object of interest is detected in every frame and the detection is labelled by an expert as correct or incorrect. For every scenario, 10 different users are indicating 9 different objects that are used for the experiments (see Figure 3.1). Users are pointing and looking at the object for 2 seconds in total. The objects used in all the experiments can be seen in Figures 3.12, 3.13. They include 5 textured and 4 uniformly coloured objects.

Opposite

The robot and human are standing opposite of each other with the object between them. Each of the ten subjects points at nine different objects. In Table 3.1, it can be seen that 93.3% of the objects are detected. Detection results are presented in Figure 3.8 and 3.9. Figure 3.9 shows that the tennis ball is the most salient object in the scene, because the tire repair box and the blue coffee cup have lower values in the combined saliency map. Both images also show that the method works with

two different pointing strategies. Figure 3.8 shows that it is working when the human is leaning forward to indicate the object and Figure 3.9 is showing when the human is pointing from a larger distance.

Table 3.1: Detection rate of the system in the case where the human and robot are facing each other. In this case Pointing (P), Saliency (S) and Depth (D) are combined to find the object of interest.

Setup	Cues	# Objects	Detection rate
Opposite	P, S, D	90	93.3 %

Figure 3.8: The subject is pointing at the blue coffee cup as indicated in the left image. The combined saliency map is located in the middle and at the right hand side the segmented object is found. The standard deviation of the pointing probability map is equal to 15°. The green dot in the saliency map and the RGB image correspond to the most salient location in the combined saliency map.

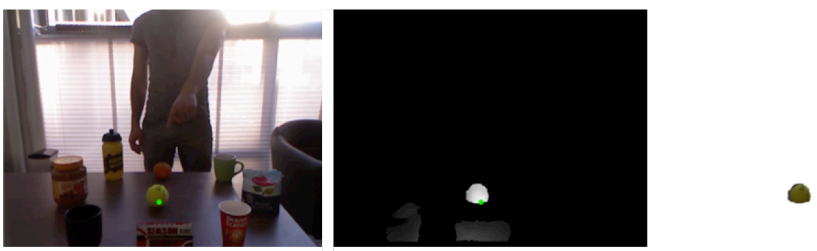


Figure 3.9: The subject is pointing at the tennis ball as indicated in the left image. The combined saliency map is located in the middle and at the right hand side the segmented object is found. The standard deviation of the pointing probability map is equal to 15°. The green dot in the saliency map and the RGB image correspond to the most salient location in the combined saliency map.

Next to

Each subject is looking for two seconds at each of the nine objects. In Figure 3.11, the subject is looking at the tennis ball (see Figure 3.10) and in this case the colour-based approach is used to generate the gaze map. When the subject is looking at the pack of sugar as can be seen in Figure 3.12, the gaze map is created by the

texture-based approach. An example gaze map can be found in Figure 3.13. Table 3.2 shows that the detection rate for textured objects is equal to 96.0% and for the uniformly coloured objects it is equal to 32.5%. This results in an overall detection rate of 67.8%. The performance in case of uniformly coloured objects is lowered due to the sensor's white balance difference between the eye tracker and the robot cameras. Use of equal cameras for the eye tracker and the robot will allow for higher performance using backprojection.

Table 3.2: Detection rate of the system in the case where the human and robot are next to each other. In this case Gaze (G), Saliency (S) and Depth (D) are combined to find the object of interest.

Setup	Cues	# Objects	Detection rate
Next to	G, S, D	50 textured objects	96.0 %
Next to	G, S, D	40 uniformly coloured objects	32.5 %
Next to	G, S, D	90	67.8 %

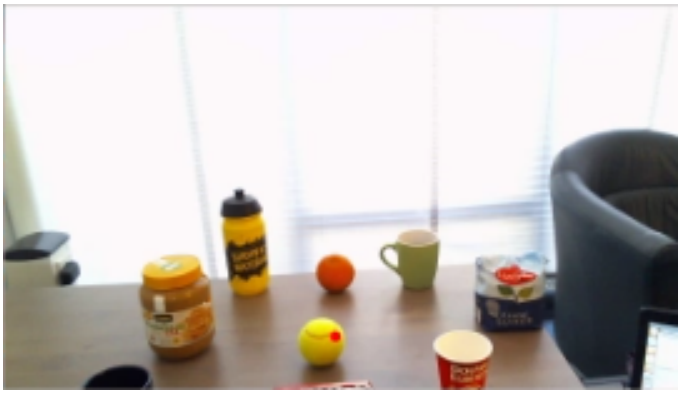


Figure 3.10: Subject is looking at the tennisball as is indicated by the red dot.

Triangle

In this experiment, each of the ten subjects is pointing and looking at the object of interest for approximately 2 seconds. In the experiment of Figure 3.14, the subject indicates the pack of sugar. The maximum point (green dot) is located at the pack of sugar, meaning that the pack of sugar is detected correctly. In figure 3.1, the subject is pointing and looking the tire repair box. The most salient point is located at the tire repair box, resulting in a correct detection of the object of interest. Table 3.3 shows the detection rates of the total system. The detection rate of the system without using the gaze map is equal to 91.1% and with the gaze map it increases to 96.7%.

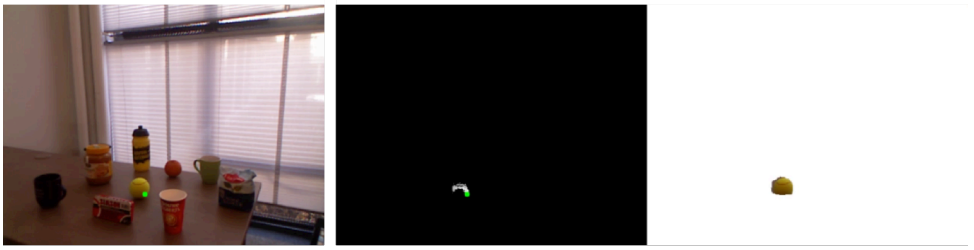


Figure 3.11: The view of the robot is presented at the left hand side. The green dot represents the most salient point in the combined saliency map. This map is located in the middle and at the right the segmented object can be found. In this case the colour-based approach is used to compute the Gaze map.

Figure 3.12: Subject is looking at the pack of sugar as is indicated by the red dot.

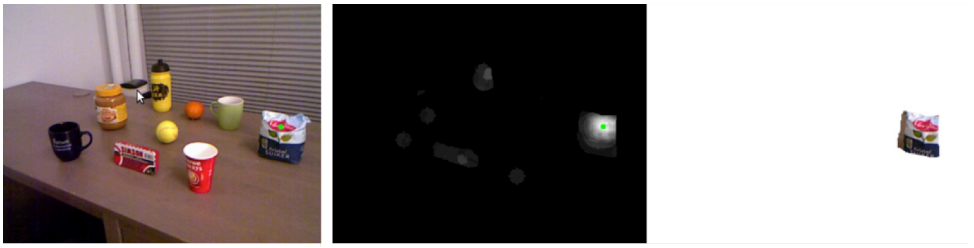


Figure 3.13: The view of the robot is presented at the left hand side. The green dot represents the most salient point in the combined saliency map. This map is located in the middle and at the right the segmented object can be found. In this case the texture-based approach is used to compute the Gaze map.

3.5.1. Influence of Distance Between Objects

In this experiment we tested the robustness of the method with respect to clutter in the scene and closely positioned objects. In this case five objects are located at

Figure 3.14: From left to right: the view from the robot,the combined saliency map and the found object of interest. The green dot indicates the most salient point in the saliency map.

Table 3.3: The detection rate of the system in the case where the human and robot are facing each other. In this case Pointing (P), Gaze (G), Saliency (S) and Depth (D) are combined to find the object of interest.

Setup	Cues	# Objects	Detection rate
Triangle	P, S, D	90	91.1 %
Triangle	P, G, S, D	90	96.7 %

1.2 m from the robot and the distance between the objects is 6 cm. This distance is varied and the experiment is repeated for 4 cm and 2 cm. For each distance, the subject indicates four times at five different objects, hence in total twenty objects should be detected per distance. Figures 3.15 and 3.16 show that the pack of sugar is detected correctly in both cases . Table 3.4 shows the results. The model performs well, since the detection rate only lowers 5% when the distance between the objects decreases to 2 cm.

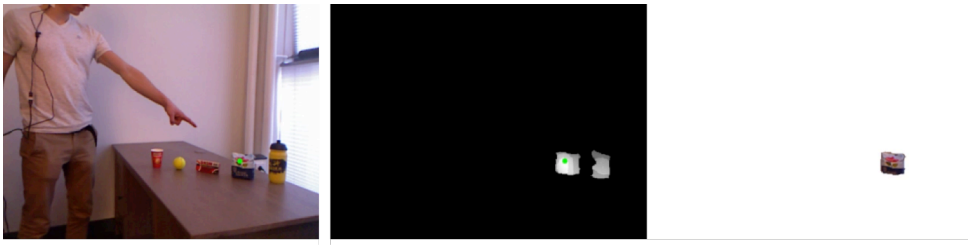


Figure 3.15: The distance between the objects is equal to 6 cm and the subject is indicating the pack of sugar. In this case, the pointing map is created with a standard deviation of 10° and the gaze map is created with a window size of 100 pixels.

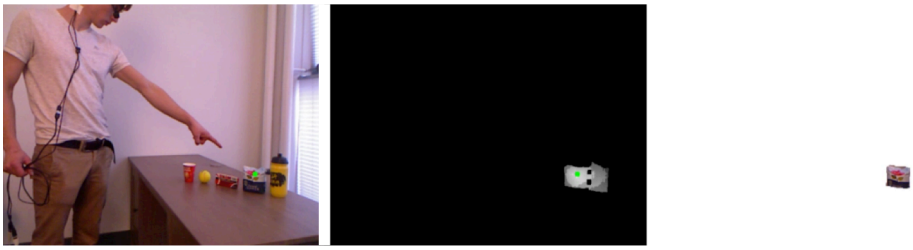


Figure 3.16: The distance between the objects is equal to 2 cm and the subject is indicating the pack of sugar. In this case, the pointing map is created with a standard deviation of 10° and the gaze map is created with a window size of 100 pixels.

Table 3.4: Detection rates at different distances between the objects.

Setup	Cues	# Objects	Detection rate
2 cm	P, G, S, D	20	95.0 %
4 cm	P, G, S, D	20	100.0 %
6 cm	P, G, S, D	20	100.0 %

3.5.2. Poor Illumination Conditions

In order to quantify the influence of light on the performance of the joint visual attention model, the model is tested in poor illumination conditions where there are many shadows and non-uniform lighting influenced by external conditions. In total six different subjects have performed a test in daylight in the 'next to' configuration. Table 3.5 shows the performance of the system. Due to poor illumination conditions the detection rate decreases from 67.8% to 48.2%. Figure 3.17 shows the result when the subject is looking at the red coffee cup. Such drop can be explained by the influence of the colour saliency map. In addition, in gaze map estimation uniformly objects are modelled by colour-based grab cut segmentation which is affected by illumination changes.

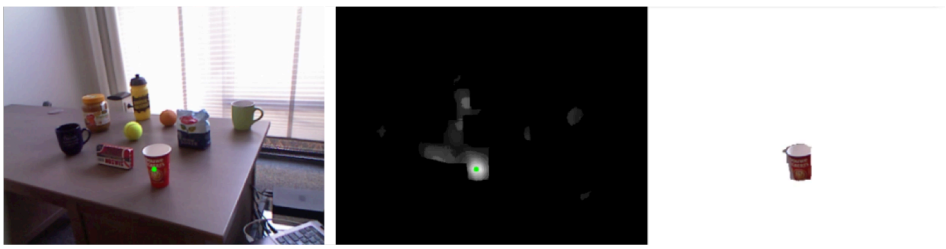


Figure 3.17: From left to right: the view from the robot, combined saliency map and the found object of interest. The green dot indicates the most salient point in the saliency map.

Table 3.5: Detection rate of the system in the case where the human and robot are next to each other. In this case Gaze (G), Saliency (S) and Depth (D) are combined to find the object of interest.

Setup	Cues	# Objects	Detection rate
Next to	G, S, D	30 textured objects	90.0 %
Next to	G, S, D	24 uniformly coloured objects	33.0 %
Next to	G, S, D	54	48.2 %

3.5.3. Influence of Different Saliency Maps

In this experiment the influence of each map is investigated by comparing four combinations of the available cues to determine the object of interest. These results are obtained in a single experiment for each of the three setups. The minimal combination of cues is the combination of the bottom-up cues, being the depth and bottom-up saliency. Table 3.6 shows that in all cases the detection rate is 11.1% for the combination of S and D, because one of the nine objects is most salient. In the 'triangle' setup, it can be seen that if the G, S and D are combined that the detection rate is 33.3%. In this case, the difference in viewing angle for the robot and subject is large. If P, S, and D are combined the result is 77.8%. However if you combine P, G, S and D the detection rate is 100.0%. This proves that integrating all four cues is a very powerful way of detecting objects of interest.

Table 3.6: Detection rates of the three different setups for four different combinations of saliency maps. The detection rate is shown in different combination of Pointing (P), Gaze (G), Saliency (S) and Depth (D).

	S,D	G, S, D	P, S, D	P, G, S, D
Opposite	11.1	×	100.0	×
Next to	11.1	77.8	×	×
Triangle	11.1	33.3	77.8	100.0

3.5.4. Influence of Cluttered Background

In this experiment, five objects are located at the table as can be seen in Figure 3.18. The background and the table are covered by a textured blanket in order to test to what extend the cluttered background is influencing the results. The subject is pointing and looking at the five objects. This experiment has been repeated five times, so in total 25 objects should be detected. It resulted in a detection rate of 100%. The gaze map is influenced, but the pointing map is also available and it is not influenced by the cluttered background because of which the object will be detected correctly. In Figure 3.18 a result can be found when the subject is pointing at the drinking bottle.

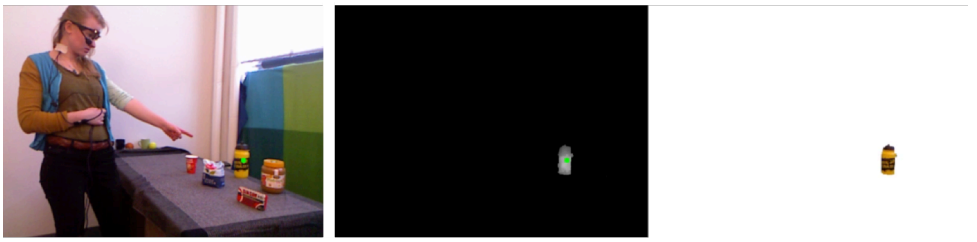


Figure 3.18: From left to right: the view from the robot, combined saliency map and the found object of interest. The green dot indicates the most salient point in the saliency map.

3.6. Conclusion

The main contribution of this work is the development of a novel flexible joint visual attention model for non-verbal human-robot interaction. The model can detect object of user interest independently of the relative position of the robot, the user and the object, without any prior training. To achieve this, both top-down and bottom-up attention cues have been combined. The attention model combines bottom-up colour saliency with a depth map, a novel user pointing direction map and a gaze map. To further improve the performance, we have developed a novel method for generating a gaze saliency map by either the texture-based approach or colour-based approach depending on textureness of the object. The system is implemented on a personal robot and it is tested in household table-top settings.

Extensive experiments show good performance results in all tested positions of user and robot; opposite, side and triangular position. Furthermore, the model works well in case of small distances between the objects as well as with a cluttered background. Lower detection rates are noticed in case of poor illumination conditions due to the influence of the colour component. This work has focussed only on static scenes, however in realistic operating conditions the robot should deal with a dynamic environment. Therefore sophisticated temporal filtering combined with additional cues such as motion have to be incorporated to robustly deal with the effects of distracted users, background motion, etc.

References

[1] Zeynep Yucel, Albert Ali Salah, Cetin Mericli, Tekin Mericli, Roberto Valenti and Theo Gever, Joint attention by gaze interpolation and saliency, IEEE Transactions on cybernetics 43, 3 (2013).

[2] Xiao, J., Moriyama, T., Kanade, T., and Cohn, J. (2003). Robust full-motion recovery of head by dynamic templates and re-registration techniques. International Journal of Imaging Systems and Technology, 13:85 - 94.

[3] B. Schauerte, J. Richarz and G. Fink, Saliency-based identification and recognition of pointed-at objects, IEEE/RSJ International Conference Robots and Systems, October 18-22, Taipei, Taiwan (2010).

[4] X. Hou and L. Zhang, Saliency detection: A spectral residual approach, IEEE Conference on Computer Vision and Pattern Recognition 23, 1 (2007).

[5] B. Schauerte and G. Fink, Focusing computational visual attention in multi-modal human-robot interaction, Proceedings of the 12th International Conference on Multimodal Interfaces (2010).

[6] C. Guo, Q. Ma, and L. Zhang, Spatio-temporal saliency detection using phase spectrum of quaternion Fourier transform, (2008).

[7] L. Itti, C. Koch, and E. Niebur, A model of saliency-based visual attention for rapid scene analysis, Pattern Analysis and Machine Intelligence, IEEE Transactions on , vol.20, no.11, pp.1254,1259, Nov 1998

[8] A. J. B. Trevor, S. Gedikli, R. B. Rusu, and H. I. Christensen, Efficient organized point cloud segmentation with connected components, Semantic Perception Mapping and Exploration, (2013).

[9] Garratt Gallagher, Finger detection, $http : //wiki.ros.org/mit - ros - pkg/KinectDemos/FingerDetection$

[10] H. Bay, T. Tuytelaars, and L. V. Gool, Surf: Speeded up robust features, Comput. Vis. Image Underst (2006).

[11] M. Muja and D. G. Lowe, Fast approximate nearest neighbors with automatic algorithm configuration, in International Conference on Computer Vision Theory and Application VISSAPP'09) (INSTICC Press, 2009) pp. 331 - 340.

[12] C. Rother, V. Kolmogorov, and A. Blake, GrabCut: Interactive foreground extraction using iterated graph cuts, ACM Trans. Graph., vol. 23, pp. 309-314, 2004.

[13] Michael J. Swain and Dana H. Ballard., Indexing via color histograms, Third international conference on computer vision, 1990.

[14] Shon, A.P. and Grimes, D.B. and Baker, C.L. and Hoffman, M.W. and Shengli Zhou and Rao, R.P.N., Probabilistic Gaze Imitation and Saliency Learning in a Robotic Head, Proceedings of IEEE International Conference on Robotics and Automation, 2005.

[15] Parvizi, E. and Wu, Q.M.J, Multiple Object Tracking Based on Adaptive Depth Segmentation, Canadian Conference on Computer and Robot Vision, 2008. CRV '08.

[16] Bleiweiss, Amit and Werman, Michael, Fusing Time-of-Flight Depth and Color for Real-Time Segmentation and Tracking, Lecture Notes in Computer Science 2009.

[17] Boris Schauerte and Rainer Stiefelhagen, Look at this! Learning to Guide Visual Saliency in Human-Robot Interaction, Proceedings of the 27th International Conference on Intelligent Robots and Systems (IROS), IEEE/RSJ, 2014

[18] OpenNI skeletal tracking, $http://wiki.ros.org/openni\_tracker$

[19] Pupil Labs open source eye tracker $https://pupil-labs.com/pupil/$

4

Multimodal human centric object recognition framework

In this chapter we focus on a perception system for cognitive interaction between robots and humans, especially for learning objects in household environments. Therefore we propose a novel three layered framework for object learning to bridge the gap between the robot's recognition capabilities with semantic knowledge of humans using the weighted fusion of multimodal sources like chromatic, structure and spatial information. In the first layer we propose the grounding of the raw sensory information into semantic concepts for each modality. We obtain a semantic color representation by using SLIC super-pixeling followed by a mapping learned from online images using a PLSA model. This results in a probability distribution over basic color names derived from cognitive linguistic studies. To represent structural information, we propose to cluster the ESF features obtained from pointcloud data into primitive shape categories. This primitive shape knowledge is learned and expanded from the robot's experience. For spatial context, a metric map from the navigation system, demarcated into landmark locations is used. All these semantic representations are compliant with a human's description of his environment and used in the second layer to generate probabilistic knowledge about the objects using random forest classifiers. In the third layer, a novel weighted fusion of the obtained object probabilities is performed, where the weights are derived from the prior experience of the robot. We evaluate our system in realistic domestic conditions provided at a Robocup@Home setting.

Chapter modified from article:
Aswin Chandarr, Maja Rudinac, Pieter P. Jonker: Multimodal human centric object recognition framework for personal robots, IEEE-RAS International Conference on Humanoid Robots, Humanoids 2014, Madrid

4.1. Introduction

In future, personal robots will share the dynamic and natural everyday living environments with humans. Though they will be equipped with basic initial knowledge about the world, they need to be able to further learn and expand their knowledge from interaction with humans and their environments. This necessitates a common cognitive understanding between robots and humans. Learning and understanding objects in household environments forms the basis of this higher level interaction, on which we focus our research.

State of the art object recognition methods [1], [2], [3] though providing a very good recognition performance on the standard benchmark datasets [4], [5], describe objects in a metric form that is difficult for human to comprehend. We illustrate this semantic gap in the Figure 4.1 where the current object recognition methods perceive a tomato as a high dimensional feature vector (metric description), whereas the human understands it with a collection of linguistic terms (an edible, red, round object, mostly found in a kitchen). A lot of attention has been given to visual scene understanding which aims to interpret the link between a collection of objects to obtain a scene label [6]. Though they provide a semantic representation of a scene, they do not describe the basic properties of individual objects which are necessary for efficient human robot interaction (HRI).

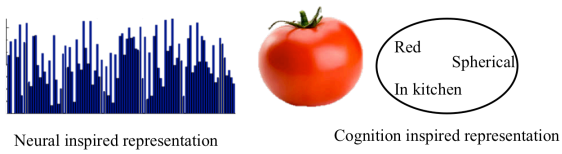


Figure 4.1: Differences in human and computer perception

To develop a human centric recognition system, we have observed the development of the perception system of infants [7]. It has been shown that perception capabilities evolve from localizing basic salient color information in the first 3 months to understanding the structure and spatial information of primitive shapes from 6 months onwards. Understanding of spatial information leads to the development of reaching and grasping capabilities [8] and further expands to semantic understanding and detailed linguistic descriptions from 2 years on. In the previous work of the group [1] [9], a robotic system capable to explore and learn new objects inspired by early cognitive development in infants till the first year was developed. Although this system provided a good recognition performance, it lacked the semantic representation. Further development of grasping and linguistic capabilities have been explored with the embodied cognitive platform, iCub [10]. Although having a very sophisticated learning process, it does not abstract the object properties into linguistic semantic concepts.

From cognitive linguistic studies [11] it has been shown that basic linguistic representation that humans use to describe the objects consists of color, shapes

[1] Intelligent Vehicles and Cognitive robotics group of the TUDelft Robotics Institute

and their attributes. In this work we will link the results of these studies with our previous work in developmental perception [9] and expand it with semantic representation of multi modal sources of information by a novel 3 layered framework for cognitive object recognition. The first layer consists of grounding the raw sensory information into semantic concepts for each modality, which are further used in the second layer to generate probabilistic knowledge about the objects which are fused in the third layer using weights derived from the prior experience of the robot. This is further used as a main recognition framework for our affordable household robot series [12]. In this chapter, we focus only on a perception system which will form a basis for future cognitive Human Robot Interaction.

4.2. Background

Our multilayered recognition framework requires fusing of multiple sources of information. The developmental perception [7] and linguistic studies [11] show that these sources are chromatic, structural and spatial information.

Chromatic information has been widely used in Content Based Information Retrieval systems where different types of color descriptors are used. Comparitive studies [13] show that best results in case of viewpoint and illumination invariance have been obtained respectively by ColorSIFT descriptors [14], histograms in different color spaces [15], color moments and moment invariants [16]. Though ColorSIFT descriptors provide the best results, due to the high number of false matches, they do not perform well in unknown object classification [9]. Hence, we have opted for a histogram based representation. However, the regular histograms provide a high dimensional feature vector that is incomprehensible for humans. Hence, we propose to use a novel histogram of semantic color descriptions. Since the robot interacts with the physical world, we use 3D structural information of the objects. Though keypoint based methods [17] provide us information to estimate the 3D pose of the objects, they do not give good results for object categorization. Hence we have opted to use a global descriptor, The Ensemble of Shape Function (ESF) [18] which showed good descriptive capacity. To extract a semantic description of the object structure we propose to cluster them into primitive shapes and allow further bootstrapping. Finally, we propose to add spatial information using the robot's navigation system.

All these semantic representations comply with a human's description of his environment and further used in the second layer to generate probabilistic knowledge about the objects. To generate this knowledge, a multi class classifier is required. Bag of words with SVM [19] are not inherently multiclass and scalability issues arise with large number of object categories. To solve this we have opted to use a random forest classifier [20] since it results in probabilities over multiple categories and allows for unknown classification.

Finally, we propose a novel online fusion method using the obtained probabilities to generate a final object hypothesis. In our previous work [21], we used statistics of object features from multiple viewpoints to enable online feature fusion from multiple sources of information. The drawback of this method is that all the sources of information need to be available. To allow object hypothesis generation in cases

of incomplete information, we propose semantical fusion where the weights are derived from prior experience.

The rest of the chapter is organized as follows. We detail the proposed recognition framework in Section 4.3. The abstraction of color semantics is explained in Section 4.4.1 while the initial knowledge and further bootstrapping of new shapes is elaborated in Section 4.4.2 and the incorporation of spatial context is briefed in Section 4.4.3. The generation of object hypothesis from the generated semantic information is detailed in Section 4.5 while the fusion of the obtained probabilities is explained in the Section 4.6. Finally we conclude with experimental results and future work in the Sections 4.7 and 8.

4.3. Cognitive learning framework

The novel three layered framework we propose is depicted in the Figure 4.2. The

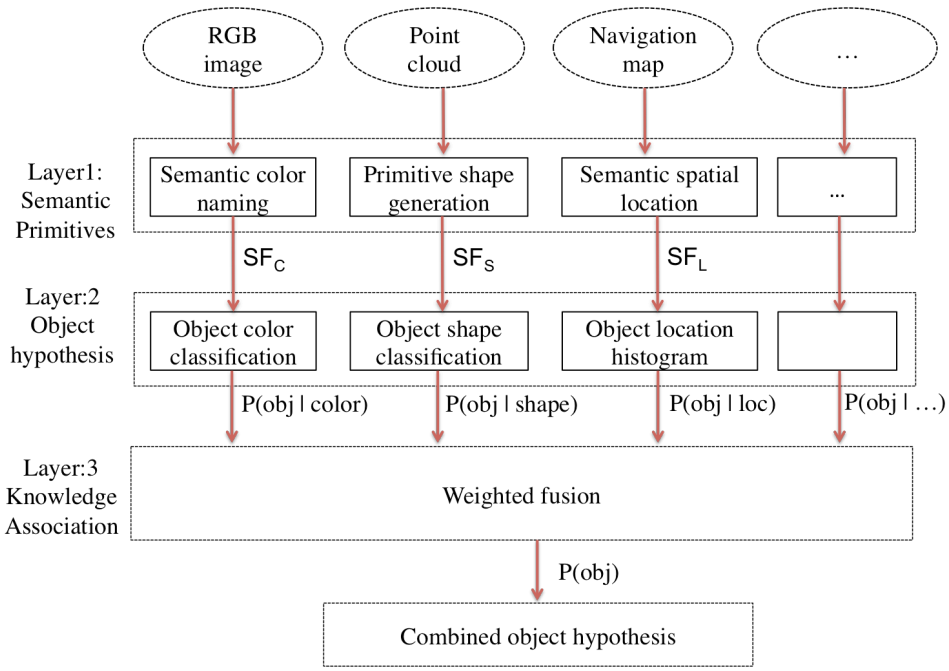


Figure 4.2: Cognitive recognition framework

raw sensory information is in the form of an RGB image, a 3D Point cloud of an object and the metric location from the robot's navigation system. In semantic primitive layer, the raw sensory information is converted into semantical features (SF). We obtain a histogram over semantic colors from raw RGB data, while for the shapes ESF features are extracted from the pointcloud and classified into a distribution over primitive shapes. New primitive shapes are also bootstrapped from experience. Metric location from the map is transformed into semantic location labels. In the object hypothesis layer, object probabilities are derived from gener-

ated semantic features. In the last knowledge association layer, the fusion of the obtained probabilities are performed by using the entropies of the distributions as the weighting factors.

The main advantage of such a framework is that it can handle incomplete or missing information of specific modalities. It is also fully modular and allows for easy addition of more sources of information. Detailed explanation of the components of the framework are provided in the subsequent sections.

4.4. Semantic primitives

4.4.1. Color semantics

The generation and use of semantic color information of an object has been well explored in the computer graphics literature. The method of assigning a linguistic $"name"$ to a pixel with a given RGB value is called Color Naming System (CNS) and it was introduced by Berk et al. [22]. This system and its many variants map the RGB value into color names with fields describing tones, shades, tints, etc and give rise to enormous different color names. To our knowledge such methods have not been used for object recognition. Previous research [23] has shown that there is a small set of basic linguistic color names that are also shared among many languages. We introduce a novel method of semantic color feature extraction which uses the entire spectrum of the perceived color data discretized into one of the following eleven color names: black, blue, brown, grey, green, orange, pink, purple, red, white, yellow which constitute the basic colors in English. Assigning the color names to the pixels cannot be a straight forward linear mapping in RGB/HSV space and hence in our novel method we use a two phased approach to tackle illumination noise and increase the speed.

Super-pixel clustering

To reduce illumination noise and increase the speed, we propose to use SLIC (Super Linear Iterative Clustering) [24] to group spatially and chromatically related nearby pixels into super pixels. This has the advantages of decreasing the noise in assigning names to each pixel and also increasing the speed of future processing by having very few data to be assigned names to. Each super-pixel is assigned the mean of the color values of all the pixels constituting the super-pixel. A sample image from our data and its corresponding super-pixelated image are shown in the Figure 4.3.

Color learning using online search

To obtain the mapping between super-pixelated RGB information and basic color names most studies [25] manually label the colors. To automatically obtain the linguistically invariant color labels, we use the system developed by Weijer et al [26]. They developed methods to learn the linguistic color mapping from a vast data of real world images obtained from an online search engine like Google. Given the noisy results obtained from the search engine, a specific PLSA (Probabilistic Latent Semantic Analysis) model was adapted to estimate the mapping. The results are available as a probability distribution of the given pixel over the previously defined

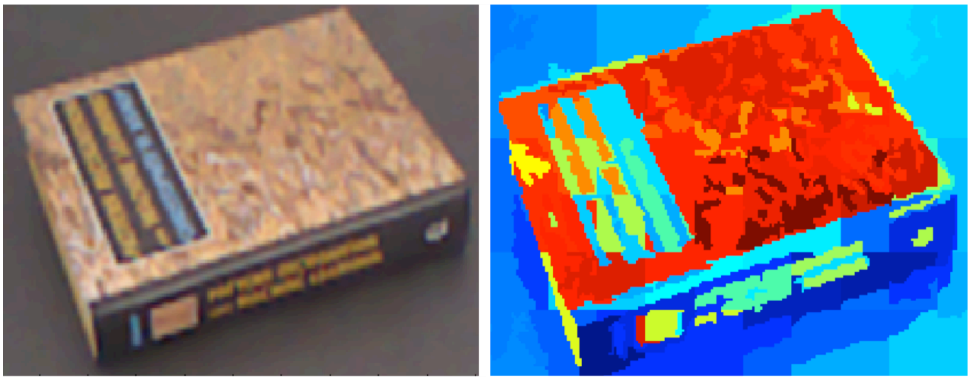


Figure 4.3: Sample image and its super pixelated image: Each color is a superpixel index

11 basic color names. The final color name is obtained by the maximum likelihood of the distribution for each pixel.

Once we have the color name assigned to each super-pixel, we obtain the color semantic feature (SF_c) of the object in a visual Bag of Words fashion. To achieve scale invariance we take the normalized histogram of the assigned color names as the global color semantic feature vector of the image. A sample image from our dataset and the corresponding color semantic feature are shown in the Figure 4.4.

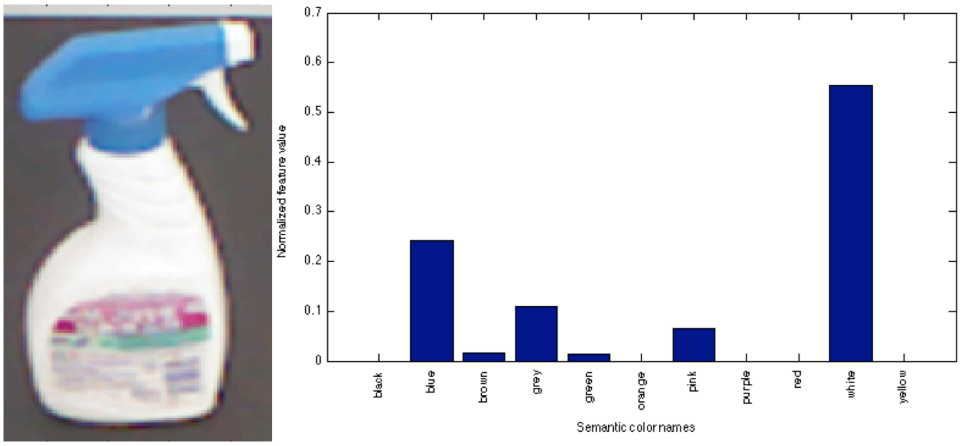


Figure 4.4: Sample image and its color semantic feature

This condensed feature can be considered as a low dimensional representation of the chromatic appearance of the object. Hence with our method we simultaneously achieve both the lower dimensionality and semantic grounding with human knowledge. Learning of the mapping between color values to names from real world images available online, can be seen as a process of robot bootstrapping its

knowledge of color from the environment. Since the entire spectrum of the color values is spanned by color names, it is possible to obtain this color semantic feature vector. However, this is not straightforward for shape semantics and will be discussed in the next section.

4.4.2. Shape semantics

It was seen in the previous section that the entire color spectrum can be described into a set of basic colors that are also common in different human languages. However, there are no unique semantic labels of shapes and the distinction between them is not well defined. Most prominent studies of the shape representations in humans have been done in the field of neuro-science, where [27], [28] identify 36 basic shapes (geons) which constitute all the shapes found in the physical world. However cognitive psychology studies [11] show that few good forms are perceptually salient among the geometrical shapes. Out of these, most primitive forms encountered in every day households are spherical, cylindrical and cubical. Hence we propose a novel method which equips the robot with the initial knowledge of these three forms and further allows to expand its knowledge of more sophisticated shapes with experience (till a maximum of 36). In later text we explain how the shape knowledge is being acquired and described.

Shape description

As the first step of our introduced method, to obtain a description of shape from the pointcloud data, we use an Ensemble of Shape Functions (ESF) [18] feature. This feature has also shown to provide good recognition performance to distinguish objects of varied shapes with scale and viewpoint invariance. ESF is a 640 dimensional feature which is a concatenation of 64 bin histograms of 10 different shape functions comprising of 3 angle, 3 area and 3 distance functions followed by a distance ratio shape function. Additionally, we use a moving least squares smoothing over the segmented object to reduce the impact of noise in 3D image acquisition. This provides local consistency to the points before the ESF features are obtained. Two different objects and their ESF features are shown in the Figure 4.5.

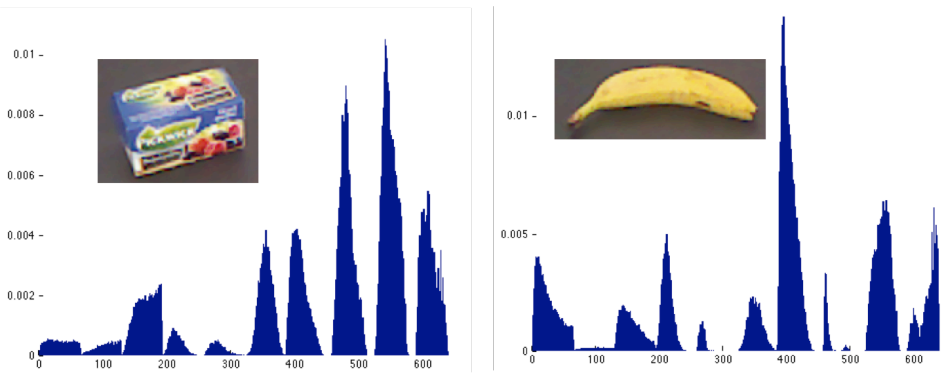


Figure 4.5: Sample objects and their ESF features

Initial shape knowledge and online learning of new shapes

As a second step of our method we equip the robot with the knowledge of the three semantic shapes. We do this by obtaining synthetic 3D models of objects of these shapes from online databases such as 3DNet. We perform supervised learning where a Random Forest classifier learns the mapping of the obtained ESF features into a probability distribution over these semantic shapes. Now this classifier can be used to cluster any newly seen object into these three shapes.

When we encounter a novel object, we first extract the ESF features. Then we use the previously learned classifier to estimate the probability of this object belonging to the three semantic shapes. Unknown object forms are detected by thresholding the probability distribution over the semantic shapes. This unknown object form can be incrementally added as a new shape to the existing semantic set and the classifier is retrained. Example objects and their respective semantic shapes are shown in the Figure 4.6

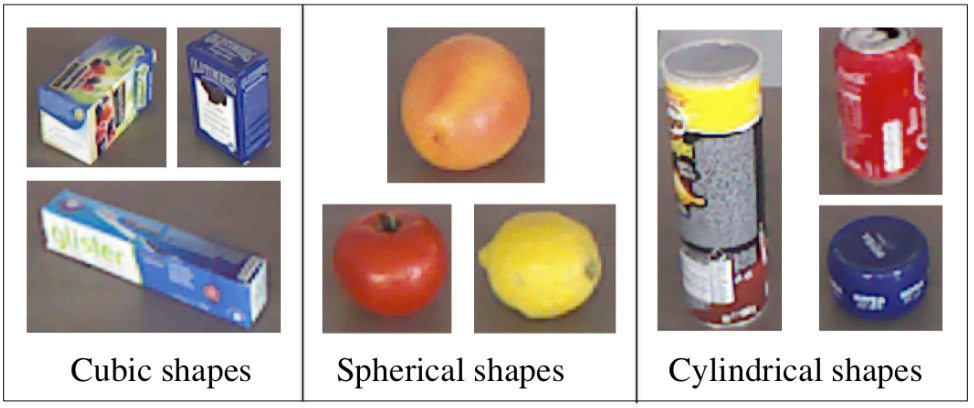


Figure 4.6: Sample objects with standard semantic shapes

4.4.3. Semantic spatial context

The location of the robot inside a household gives some powerful cues regarding the object classes. The location of the robot is determined based on its position the metric map it uses for its navigation. The different rooms are demarcated in the map and the robot looks up its current metric position to the spatial category. The sample metric map used and the room labels in it are shown in the Figure 4.7. During the training phase, the information about the most commonly present locations for each object is acquired. Such information is also available in the benchmark for personal robots such as RoboCup@Home and RocKiN competitions.

4.5. Object hypotheses

The goal of this layer is to learn the mapping from the distribution over the semantic labels for each modality to the object class labels. This is done independently for

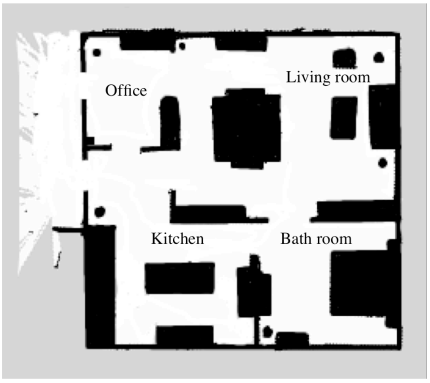


Figure 4.7: Robot navigation map with demarcated locations

each of the modalities which are currently limited to color, shape of the object and the location of the robot. The initial knowledge generation of the robot is done in a fully supervised manner using a set of random forest classifiers.

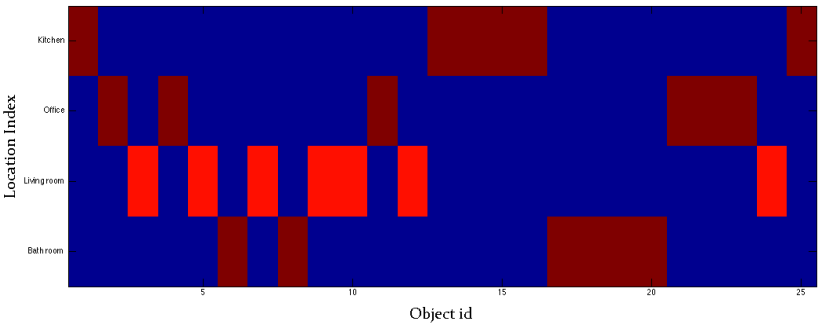
In the previous layer, a color semantic feature is obtained as a histogram of the basic semantic colors. A classifier is trained to learn the mapping $P(obj|color)$ between the semantic color features and the object labels.

For the object shape classification, we use the shape semantics as well as the ESF features itself generated from the previous layer to train a classifier to obtain the probabilities $P(obj|shape)$ of object labels.

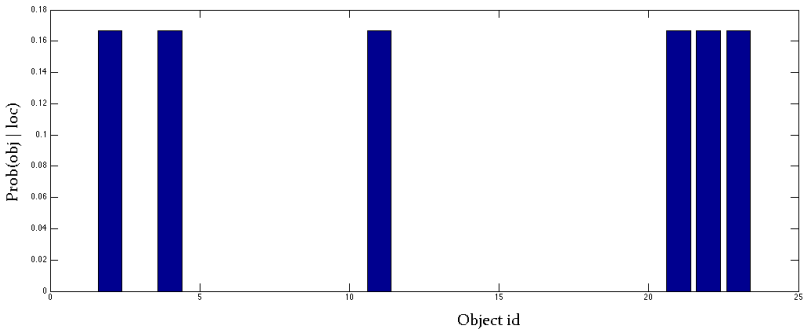
For the object location classification, during the training phase an Object Location matrix is obtained as a probability distribution of presence of each object in a different location, as shown in Figure 4.8a. Once a new object is encountered, its semantic location is determined from the position of the robot in the navigation map and the probabilities of different objects being present in that specific location $P(obj|loc)$ are obtained from the generated Object Location matrix, as shown in Figure 4.8b.

4.6. Knowledge Association

Once the object probabilities from different modalities are obtained, they have to be automatically fused to generate the final object hypothesis. This fusion needs to provide an optimal combination of different modalities as well as handle the conditions where information from certain modalities are missing. One solution would be the use of late fusion techniques [29], where the individual probabilities are concatenated and a new classifier is learned from them. However, this gives problems when one of the sources is missing as well as a new modality has to be added. To simultaneously solve these challenges, we introduce a novel method for weighted fusion based on information theory. The combined object probability is



(a) Object Location matrix



(b) Object probabilities for given location: Office

Figure 4.8: Incorporating spatial information

then obtained using Equation (4.1)

$$P_{obj} = W_{color} * P(obj|color) + W_{shape} * P(obj|shape) + W_{loc} * P(obj|loc) \tag{4.1}$$

where the weights are obtained as

$$W_x = e^{-S_x} \tag{4.2}$$

$$S_x = \Sigma(-P_x * log_2 P_x) \tag{4.3}$$

where P_x is the discrete probability of the object based on color, shape or location. We propose to use weights which quantify the uncertainty of each modality. These weights are obtained as the entropy of the object hypothesis from each modality. If robot has very little or no encounters with the object, it will be highly uncertain about the object, which is reflected in higher entropy value and therefore leading to lower weights as described in Equation 4.3. One advantage of this method is that the weights will automatically become zero if the data from one of the modalities

is not available. Secondly, this allows for automatic selection of dominant features of the object.

To ensure that the combined probabilities of all the objects fall in the same range, the final object probability is normalized over the maximum probability value, Equation 4.4

$$\bar{P_{obj}} = \frac{P_{obj}}{\max_{1:N} P_{obj}(n)} \tag{4.4}$$

Lastly the object label is obtained as the maximum likelihood estimate of the normalized distribution $\bar{P_{obj}}$. This is performed also to provide compatibility for future multi-view learning systems.

4.7. Experimental Setup

Our introduced recognition system is designed for personal robots in a household scenario. In order to realistically evaluate our system, we have mimiced the Robocup@Home competition settings where a furnished apartment with commonly used objects is used to test the robot performance. In our lab, we have constructed such an apartment comprised of different rooms which is used to validate the robots capabilities as can be seen from Figures 4.7, 4.9. Our robot LEA is used

Figure 4.9: Constructed robot apartment

for evaluation of the developed algorithm. In order to make a realistic dataset, we have positioned the robot at different locations within the apartment where various household objects were presented to the robot. We used a total of 25 objects varying in color and shape, some of them can be seen in Figure 4.6. The objects are grouped based on the locations where they are generally present, such as kitchen, living room, office and bathroom. The objects are presented in front of the robot at the respective locations and the pointclouds are obtained at two different tilt angles of the neck at 15^o and 45^o as these two viewpoints represent the most common scenarios in household conditions. The objects are automatically segmented from the pointcloud data by extracting the planes using RANSAC and followed by Euclidean clustering as in [5]. To test the robustness towards illumination changes, we obtained the data at 3 different times of the day as seen in Figure 4.10. In total

Figure 4.10: Illumination and viewpoint variations in the dataset

our dataset contains 1000 object images. We opted not to use the state of the art RGBD dataset [5] as there are no illumination changes nor information on spatial location available in this dataset. In our dataset, such information is automatically acquired from the robot's navigation system as explained in section 4.4.3. Detailed experimental results are presented in the next section.

4.8. Experimental results

In the first experiment we tested the overall accuracy of the system. All 1000 images of the database are used for evaluation. For testing a leave one out cross-validation method is used in each of the 3 illumination and 2 viewpoint settings. The results in Table 4.1 show the overall accuracy for different modalities as well as the combined accuracy. The accuracy increases from 73% in case of using only color, to 88 % while using only shape, and to 92% for the combination of shape and color and providing a good accuracy of 96 % while using all three modalities combined. It can be seen that using only color information provides the lowest accuracy. The common confused objects for different modalities are listed in Figure 4.11. It can be seen that objects with semantically identical colors such as $banana$ and $lemon$, $bellpepper$ and $tomato$ are commonly misclassified based on color. Also just based on the shape, the objects such as $cream$ and $stapler$ and two different cups are confused as shown in the Figure 4.11. This is because we obtain the object class by the maximum likelihood estimate of the generated object

probability density. However, if hypotheses from other modalities are giving higher probability for the correct class as well, the combined weighted fusion provides a very high recognition performance, as reflected in Table 4.1. This provides an accuracy improvement from 73% to 96 %

	Color	Shape	Color + Shape	Color+Shape+Loc
Accuracy(%)	73	88	92	96

Table 4.1: Overall recognition accuracy

Figure 4.11: Commonly confused objects

In the second experiment we tested the robustness of the system to shape variations due to viewpoint change. The training is performed on images from a single viewpoint (15^o) and are tested on images from another viewpoint ($45\ ^o$). Figure 4.10 shows the shape variations between the viewpoints. This affects the performance of the ESF feature, which is a global descriptor, as is also depicted in the results from Table 4.2, where the accuracy of the shape decreased to 65%. In such cases the combination with color and location information significantly improves the recognition till 89 %.

Accuracy	Color	Shape	Color+Shape	Color+Shape+Loc
Viewpoint 15^o	66	65	82	89
Viewpoint 45^o	63	70	76	85

Table 4.2: Influence of viewpoint change

In the third experiment the influence of illumination variation is evaluated. We perform training on images from two illumination conditions and test on images from a third illumination setting. Images from different illumination conditions significantly vary as shown in Figure 4.10. However, these conditions are expected

in household environments and good recognition performance is required despite these severe changes. The results in Table 4.3 show that the performance of the color modality drops to 65 %, which is due to the color descriptors based on RGB data. We have also tested other color descriptors [15] which provide only a slight increase in performance but do not allow semantic inference that is achieved here. In addition in combination with other modalities, the obtained accuracy is significantly increased to 92% In the final experiment we tested the influence of the

	Color	Shape	Color + Shape	Color+Shape+Loc
Accuracy (%)	65	84	88	92

Table 4.3: Influence of illumination change

amount of training samples. The size of the training set was varied from 20, 40, 60 to 80 % of the available samples while the rest of the images were used for testing. The learning curves for different modalities are shown in Figure 4.12 which show that the combined modalities provide a high increase in accuracy for all sizes of the training dataset as well as an accuracy of over 85 % even with a training size of 20%. The smaller training dataset allows for a compact representation and small memory usage as well as faster learning time.

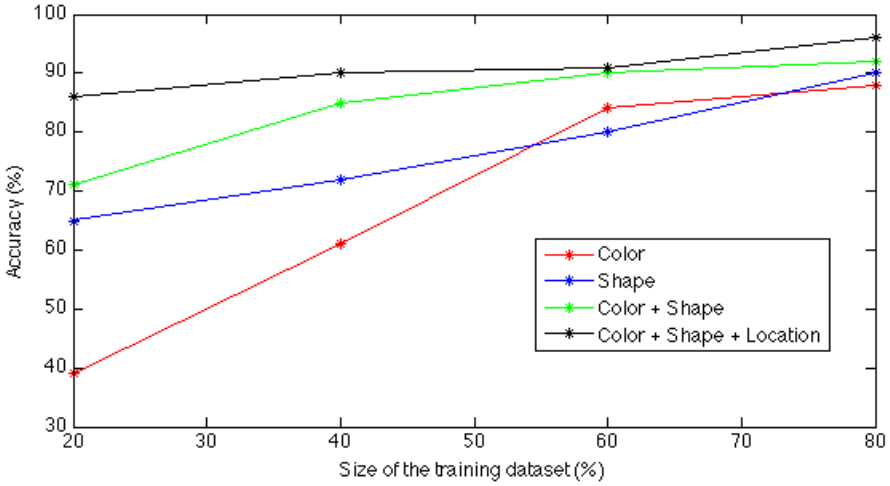


Figure 4.12: Influence of size of training set

4.9. Conclusions

In this chapter we have presented a recognition framework that can provide a basis for cognitive interaction between humans and robots. We have introduced a 3 layered system that uses multimodal sources of chromatic, structural and spatial

information available to the robots to generate a probabilistic object hypothesis. The color was represented by a histogram of color named super-pixels, with the color names obtained from cognitive linguistic studies. For shape description ESF features of the 3D point cloud where used while the shape semantics was initialized with primitive shape forms and learned further with experience. The spatial context was obtained through the robot's navigation system and incorporated using an object-location histogram. To form object hypotheses, a weighted fusion of the obtained probabilities was introduced, where the weights were functions of entropies that quantify the robot's prior experience and the confidence of correct classification of each modality. We have also shown the robustness of the system by evaluating it on a realistic robotic setting in a home environment with illumination, location and viewpoint changes. Our system gives good object recognition results despite viewpoint and illumination changes as well as in the case of small training datasets while at the same time, allowing object description coherent with human linguistics. This human centric recognition framework opens a window of possibilities for higher cognitive level interaction between humans and robots.

References

[1] Richard Socher and Brody Huval and Bharath Bhat and Christopher D. Manning and Andrew Y. Ng, *Convolutional-Recursive Deep Learning for 3D Object Classification, Advances in Neural Information Processing Systems 25*, (2012).

[2] D. G. Lowe, *Distinctive image features from scale-invariant keypoints,* Int. J. Comput. Vision **60**, 91 (2004).

[3] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, *Speeded-up robust features (surf),* Comput. Vis. Image Underst. **110**, 346 (2008).

[4] M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, *The pascal visual object classes challenge–a retrospective,* .

[5] K. Lai, L. Bo, X. Ren, and D. Fox, *A large-scale hierarchical multi-view rgb-d object dataset,* in *ICRA* (IEEE, 2011) pp. 1817–1824.

[6] L.-J. Li, H. Su, Y. Lim, and F.-F. Li, *Object bank: An object-level image representation for high-level visual recognition,* International Journal of Computer Vision **107**, 20 (2014).

[7] K. D. Kinzler and E. S. Spelke, *Core systems in human cognition,* Progress in Brain Research **164**, 257 (2007).

[8] K. Newell, D. Scully, P. McDonald, and R. Baillargeon, *Task constraints and infant grip configurations,* Developmental Psychobiology **22**, 817?832 (1989).

[9] M. Rudinac, *Exploration and Learning for Cognitive Robots*, Ph.D. thesis, Delft University of Technology (2013).

[10] D. Vernon, C. von Hofsten, and L. Fadiga, *A Roadmap for Cognitive Development in Humanoid Robots*, 1st ed., Cognitive Systems Monographs, Vol. 11 (Springer-Verlag Berlin Heidelberg, 2011) p. 250.

[11] F. Ungerer and H.-J. Schmid, *An introduction to cognitive linguistics* (Routledge, 2013).

[12] A. Chandarr, M. Bruinink, F. Gaisser, M. Rudinac, and P. Jonker, *Towards bringing service robots to households: Robby ,lea smart affordable interactive robots,* in *IEEE/RSJ International Conference on Advanced Robotics (ICAR 2013)* (2013).

[13] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek, *Evaluating color descriptors for object and scene recognition,* IEEE Transactions on Pattern Analysis and Machine Intelligence **32**, 1582 (2010).

[14] G. J. Burghouts and J. M. Geusebroek, *Performance evaluation of local colour invariants,* Computer Vision and Image Understanding **113**, 48 (2009).

[15] J.-M. Geusebroek, R. van den Boomgaard, A. W. Smeulders, and H. Geerts, *Color invariance,* IEEE Trans. Pattern Anal. Mach. Intell. **23**, 1338 (2001).

[16] F. Mindru, T. Tuytelaars, L. Van Gool, and T. Moons, *Moment invariants for recognition under changing viewpoint and illumination,* Comput. Vis. Image Underst. **94**, 3 (2004).

[17] B. Steder, R. Rusu, K. Konolige, and W. Burgard, *Point feature extraction on 3d range scans taking into account object boundaries,* in *Robotics and Automation (ICRA), 2011 IEEE International Conference on* (2011) pp. 2601–2608.

[18] W. Wohlkinger and M. Vincze, *Ensemble of shape functions for 3d object classification.* in *ROBIO* (IEEE, 2011) pp. 2987–2992.

[19] J. Zhang, M. Marszałek, S. Lazebnik, and C. Schmid, *Local features and kernels for classification of texture and object categories: A comprehensive study,* International journal of computer vision **73**, 213 (2007).

[20] L. Breiman, *Random forests,* Machine Learning **45**, 5 (2001).

[21] M. Rudinac and P. Jonker, *A fast and robust descriptor for multiple-view object recognition,* in *(ICARCV), 2010 11th International Conference on Control Automation Robotics and Vision* (2010) pp. 2166–2171.

[22] T. Berk, L. Brownston, and A. Kaufman, *A New Color-Namiing System for Graphics Languages,* IEEE Computer Graphics and Applications **2**, 37 (1982).

[23] B. Berlin and P. Kay, *Basic color terms: Their universality and evolution,* (1969).

[24] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Su, *Slic superpixels compared to state-of-the-art superpixel methods,* IEEE Transactions on Pattern Analysis and Machine Intelligence **34**, 2274 (2012).

[25] Y. Liu, D. Zhang, G. Lu, and W.-Y. Ma, *Region-based image retrieval with high-level semantic color names,* in *Multimedia Modelling Conference, 2005. MMM 2005. Proceedings of the 11th International* (IEEE, 2005) pp. 180–187.

[26] J. Van De Weijer, C. Schmid, J. Verbeek, and D. Larlus, *Learning color names for real-world applications,* Image Processing, IEEE Transactions on **18**, 1512 (2009).

[27] I. Biederman, *Recognition-by-components: A theory of human image understanding,* Psychological Review **94**, 115 (1987).

[28] S. O. Murray, B. A. Olshausen, and D. L. Woods, *Processing shape, motion and three-dimensional shape-from-motion in the human cortex,* Cerebral Cortex **13**, 508 (2003).

[29] Y. Dong, S. Gao, K. Tao, J. Liu, and H. Wang, *Performance evaluation of early and late fusion methods for generic semantics indexing,* Pattern Analysis and Applications **17**, 37 (2014).

5

Contour tracking based on depth and semantic color features

In this chapter we tackle the challenges of visual tracking for personal robots. We have proposed a novel track-by-detection method that combines a semantic object model with depth properties to obtain target contours. The tracking can be initialized by either 2D or 3D inputs, which are further refined using clustering based background removal to obtain an initial object model. During tracking, we propose to refine the search space by metric constancy and removal of the support plane. Further more, the target appearance is modelled using a semantic human centric color descriptor, continuously updated by an online learning algorithm. The spatial compactness of the target is described using a Gaussian model with an initially determined variance. A fusion of the obtained color and depth models based on a target-background dissimilarity measure, is used to perform segmentation based tracking using graph cuts to obtain object contours. The experimental results in a household scenario show a good performance of the algorithm in challenging conditions such as scale, viewpoint change and out of plane rotations.

5.1. Introduction

Among the major challenges that personal robots face are dynamic learning and exploration of the human environments. To be able to adapt to new environments and users, robots have to learn about objects and human activities and also act in

Chapter modified from article:
Aswin Chandarr, Maja Rudinac, Pieter P. Jonker: A novel multi modal tracking method based on depth and semantic color features for human robot interaction, 14th IAPR International Conference on Machine Vision Applications MVA 2015, Tokyo

continuously changing conditions. These require a robust method for object and person tracking, invariant to scale, viewpoint changes and out of plane rotations. Effective manipulation and learning of objects requires acquiring continuous object contours in addition to a target position provided by tracking methods. The challenges posed by these conditions on current state of the art tracking methods are elaborated below.

There are several state of the art 3D tracking methods using pointclouds [1], but they require an a-priori 3D mesh with particle filter which is not suitable for a robot's exploration of unknown objects. In addition, methods based on particle filters do not scale with larger and deformable objects.

The most efficient 2D state of the art tracking algorithms are based on a closed loop track, detect and learn framework [2]. They use local features and optical flow which leads to failure and drift in case of uniformly colored objects and out of plane rotations, commonly encountered by personal robots.

Algorithms like [3] overcome these problems by using an incrementally learning structured output SVM and part based features. Apart from not utilizing the additional depth information available, they only track bounding boxes over the targets and do not provide contours, required for manipulation tasks. Several methods have been proposed that utilize depth in addition to 2D models [4], however, they do not provide contour information and do not tackle the problems of out of plane rotations and large viewpoint change very well.

To obtain object contours several methods have been proposed ([5], [6]). Deformable part based models have been utilized by [5], while [6] uses a graph cut method fusing high level (object detection) and low level (color/motion) features. However, these methods consider only the properties of the target and are not adaptive to varying target background visual characteristics.

In this chapter we introduce a robust tracking algorithm that provides contours with multi modal features that can also be used for semantic object recognition [7]. The proposed *track by detection* algorithm shown in Figure 5.1, uses online learning, semantic color feature description combined with depth to provide target contours while tackling challenging conditions. We propose a background elimination and search space refining step followed by a novel optimal fusion of color and depth information in a graph cut methodology to obtain segmentation. Continuous learning and update of target models decreases drift and results in the generation of a complete feature space of the target.

Our algorithm tackles the common conditions encountered by personal robots such as scale, viewpoint change and out of plane rotation with an algorithm invariant to visual object attributes. The used semantic human centric color description and the generation of target contours can be used for object recognition, manipulation and higher level human robot interaction. The rest of the chapter is organized as follows. Section 5.2 describes target initialization followed by search space refining in section 5.3. Section 5.4 explains the color and depth target modeling and segmentation and learning using the optimal fusion of these modalities is detailed in Section 5.5. The experimental setup and the results are shown in Section 5.6 followed by conclusions and future work.

5.2. Initial target selection

Target initialization is the first and critical step as it provides the base model for the entire tracking process. We use a 2D bounding box for initial target selection as required by human robot interaction applications such as user initiated grasping as well autonomous grasping based on 2D object detection methods. However, the area within the bounding box can include parts of the background, which can cause drift during the tracking process. Given that tracking targets for robotic applications consist of spatially coherent objects, we minimize drift and obtain an accurate initial model. An Euclidean clustering is perfromed on the points in the cloud within the initial bounding box and the dominant cluster is modeled as the initial target.

5.3. Search space refining

Once the initial model is selected, the object position has to be estimated over the consecutive frames. The knowledge of the target position in the previous frame is used to restrict the search area for detection in the next frame. Additionally, certain geometric properties of the scene enables masking of parts of the scene as definite background. We further explain how these two properties are used to refine the search space of the object for the proposed tracking by detection mechanism.

While the size of the target in pixels can vary based on scale, the metric size of the target (m) is a constant. We exploit the variation of the pixel boundaries based on the metric size, distance and the camera intrinsics as described in the (5.1). This relates spread of the search space in x, y directions with the focal length (f) with the mean depth of the last frame (μ_{depth}), enlarged by a scale factor s. A scale factor of 2 has been used as the object size cannot increase more than twice between two frames.

$$spread_{new} = \frac{s \times f \times target_{size}}{\mu_{depth}} \tag{5.1}$$

The next image frame is cropped to $spread_{new}$ to restrict the search space, which is further refined as follows.

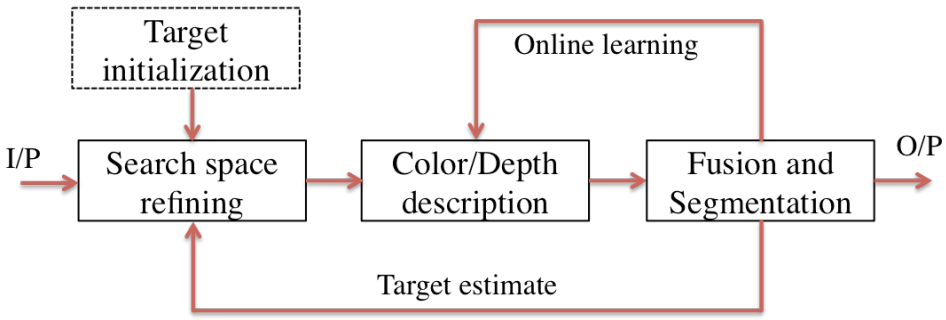


Figure 5.1: Tracking framework

It can also be observed that the targets we are interested in are mostly placed on support surfaces such as a table, floor etc. In human environments, these are always planar regions, specifically perpendicular to the target. We model the target orientation with a surface normal (N_{target}) perpendicular to the dominant planar component of the target. Large planar regions in the scene oriented normally to the N_{target} are localized using a RANSAC based plane fitting. They are removed from the source image to obtain a final search space mask. An example result of this process can be seen in Figure 5.2. Once the search space is obtained, each point here should be assigned a probability of belonging to the target. This is explained in the next section.

5.4. Target description

Given the initial target region, dense maps can be obtained for the consecutive frames that describe the probability of each point belonging to the object based on different modalities. In order to track both uniform and textured objects, we have combined semantic color appearance with depth properties.

5.4.1. Color description

A semantic color appearance model provides robots with perception of colors, similar to human understanding. Color Naming System (CNS) derived from cognitive psychology studies provides a set of 11 basic colors, common to most languages. In a previous research [8], a pixel based probabilistic mapping from the RGB to CNS has been obtained from natural images obtained from online search using a PLSA method. This has been used with reduced dimensions as a feature for tracking in [9] and also for object recognition in [7]. We use this as a color descriptor to obtain an illumination invariant semantic representation of the object. This is significant as it allows for multiple view semantic object recognition and higher level cognitive human robot interaction in the future. We obtain a normalized 11 dimensional feature at each point. A set of positive and negative feature samples are obtained by sampling from the known target and background regions. We learn a discriminative model separating these regions on an 11 dimensional manifold using AROW (Adap-

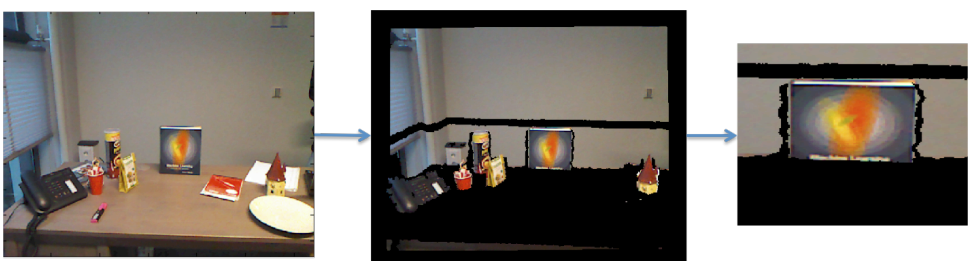


Figure 5.2: Refining the search space. The planar regions perpendicular to target are removed and search space is restricted based on metric consistency from previous frame

tive Regularization of Weights) a fast second order online learning algorithm [10]. This provides a confidence value for each point belonging to the target model. This confidence output is normalized as in (5.2) to describe the probability of the point belonging to the target (Pr_c).

$$Pr_c = \frac{c - min(C)}{max(C) - min(C)} \tag{5.2}$$

where, c is the classifier output at a particular pixel. This is further combined with the depth properties of the target.

5.4.2. Depth description

Since the whole object is considered as a single spatially coherent entity, we use only depth data to represent the spatial compactness of a target. Given the initial region, the target depth is modeled as a single dimensional Gaussian parametrized by μ and σ obtained from the initial model. With this, a dense probability map (Pr_d) of the search space is obtained using (5.3)

$$Pr_d = e^{K\left(\frac{D-\mu}{2\sigma}\right)^2} \tag{5.3}$$

where D is the depth at a given point K describes the steepness of the target background boundary. Hence Pr_d represents the probability of a point belonging to the target model based on its current depth properties. This kind of depth model assigns higher costs to any other object occluding the target, thereby making the tracking robust to occlusion. Having obtained a probability map based on color and depth, we combine them optimally and use it for segmentation of the search space as explained in the next section.

5.5. Object detection and learning

The obtained probability maps have been used to obtain a segmentation of the target and the background using a graph-cut formulation [11]. In this approach, the search space is modeled as a Markov random field and the segmentation is the configuration of the latent variable (target or background) corresponding to the minimal energy. The solution provides an optimal segmentation considering the data cost (D_c) which represents the cost of pixel (i) labeled to its components (l_i: target/background) based on the observations at the pixel (x_i: color/depth) and a smoothness cost (S_c) which enforces neighboring pixels (Nbr_i) have the same labeling. The solution is the minimization of the total energy $E(x)$ in the Equation (5.4).

$$E(x) = \sum_{i \in S_s} \left(D_c(l_i|x_i) + \sum_{j \in Nbr_i} S_c(x_i, x_j) \right) \tag{5.4}$$

The smoothness cost is assigned to a 4-connected neighborhood based on their relative euclidean distances which is proportional to their depths. This cost between

neighboring pixels (x, y) defined in (5.5) enforces spatial coherence in the segmentation process. $Depth_x$ is the distance value of the pixel (x) from the camera.

$$S_c = \frac{1}{|Depth_x - Depth_y| + \varepsilon} \tag{5.5}$$

The D_c is obtained by an optimal combination of the color and depth probabilities. A target-background dissimilarity measure (Ds) is used for adaptability to different kinds of targets in any environmental conditions,. This is obtained as distances between the normalized histograms representing visual characteristics of target and background. We use the Bhattacharyya distance [12] which provides a measure in the range $[0, 1]$. The spatial distribution(DH_{tg}, DH_{bg}) of the components is represented using histograms of the depth map of target and background quantized into 10cm bins. The chromatic distribution (CH_{tg}, CH_{bg}) is obtained using a 168 dimensional histogram of the image quantized in HSV space as used in [13]. The dissimilarity Ds_c, Ds_d is obtained using a Bhattacharyya distance between target and background histograms. For the compatibility with graph cut, the color and depth probability maps are inverted and scaled to penalties (Pen_c, Pen_d) and fused into a single data cost (D_c) by using normalized weights (w_c, w_d) which are obtained as in (5.6)

$$\begin{aligned} Pen &= A\,(1 - Prb) \\ D_c &= w_c Pen_c + w_d Pen_d \\ w_c &= \frac{Ds_c}{Ds_c + Ds_d},\ w_d = \frac{Ds_d}{Ds_c + Ds_d} \end{aligned} \tag{5.6}$$

where, A is a scale to achieve uniformity between penalties for different modalities. Now, having obtained both D_c and S_c, we obtain the segmentation T_{new} of the target by applying the min-cut [11] followed by a few morphological operations. An illustration of the different potentials over the entire frame is shown in Figure 5.3 where the penalty value increases from blue to red color. To incrementally learn the color model, the regions of the color probability map (Pr_c) that have a low value, but still are present in the new segmentation (T_{new}) are updated to

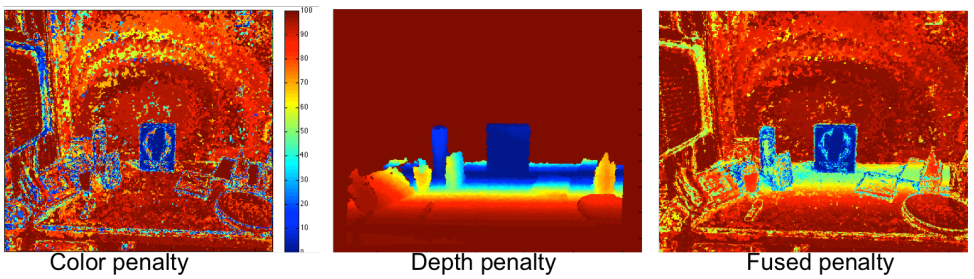


Figure 5.3: Color, depth and fused potentials

the online learner. This ensures that we have a complete model during object exploration applications. To minimize the drift, the μ of the depth model is updated while the σ is kept constant at the initial value. All these components connected together in the framework of Figure 5.1 result in continuous target tracking. The performance of the algorithm is evaluated as explained in the next section.

5.6. Experimental setup and Results

The tracking system is targeted for personal/service robots operating in indoor environments using RGB-D sensors. Currently there does not exist a dataset for scenarios encountered by personal robots for object exploration, learning and manipulation. We have implemented the tracking system on a service robot [13] and in order to realistically evaluate the performance of the system, we tested it on a challenging dataset consisting of 7582 frames divided into 15 categories of household objects in various conditions. The data was created using a Microsoft Kinect with a VGA resolution at 15fps. Some of the objects used in the dataset are shown in Figure 5.4. We tested the tracking performance during conditions of scale change, viewpoint variation and out-of-plane rotations taken in a real household setting with varying illumination conditions. An experiment was performed with a relative scale change between robot and target during the tracking and a viewpoint change from -60^o to +60^o.

5.6.1. Results

The tracking performance is evaluated by using the overlap criteria defined by (5.7) as used in [14]

$$precision = \frac{match_{area}(bb_{truth}, bb_{tracked})}{area(bb_{tracked})} \tag{5.7}$$

The performance measures reported are the percentage of frames that have a precision greater than 0.5 when using different modalities. Table 5.1 shows the general (overall) tracking performance. Using only color provides a mean precision of 78 % and using only depth has a precision of 72%. When only depth is used, the performance decreases due to the drift of the tracked target into nearby regions with similar depth. But when coupled with color information it can be seen that this drift is minimized and we obtain an enhanced tracking performance of 84% which is comparable with state of the art tracking methods [4]. Table 5.2 shows the mean

Table 5.1: Overall tracking performance

	Color	Depth	Color + Depth
Precision (%)	78	72	84

precision at different viewpoints which also accounts for out of plane rotations. It can be seen that the performance when using only color decreases with a large viewpoint change. This is due to the large variance of illumination conditions as well

different color distributions over the target periphery. Since the depth distribution remains invariant to the change in viewpoints, attributed to the spatial coherence of the targets, combining color and depth provides a good tracking performance inspite of the challenging viewpoint changes.

Table 5.2: Viewpoint change performance

Viewpoint	-60^o	-30^o	0^o	30^o	60^o
Color (%)	68	72.5	78	74	69
Color + Depth (%)	73	76	85	77.5	72

The influence of scale change on the tracking performance is quantified in the Table 5.3. When the object is very far from the robot, the target has only very few pixels which are not discriminative enough from the background in the 11 dimensional feature space. This results in lower tracking precision at farther scales while using only a color model. It can also be seen that addition of depth properties considerably increases the performance, validating the robustness of the algorithm.

Table 5.3: Scale change performance

Scale	1 (Far)	2	3	4 (Near)
Color (%)	63	67	73	76
Color + Depth (%)	70	73	78	83

The algorithm has been successfully implemented and tested on our personal robot LEA.

5.7. Conclusions

In this chapter a novel algorithm for object and person tracking is introduced for human robot interaction tasks. After the target is initialized using a bounding box over RGBD data the final target model is obtained by using a clustering algorithm. The search space of the consecutive frames is refined using the metric size constancy of the objects and also by removing the planar support surfaces. The object appearance has been modelled using both color and depth modalities. A color feature based on CNS has been used with a classifier to obtain a color probability map while the depth probability map is obtained by using a Gaussian model of the object with fixed initial variance. A novel method to perform optimal fusion of different object modalities using a target-background dissimilarity measure has been introduced. This has been used in a graphcut framework to obtain the object contour. Extensive experiments have been performed in a household environment showing good performance under challenging conditions of viewpoint and scale change as well as out of plane rotation.

Figure 5.4: Sample tracked contours

References

[1] C. Choi and H. Christensen, *Rgb-d object tracking: A particle filter approach on gpu,* in *Intelligent Robots and Systems (IROS)* (Tokyo, 2013) pp. 1084–1091.

[2] Z. Kalal, K. Mikolajczyk, and J. Matas, *Tracking-learning-detection,* Pattern Analysis and Machine Intelligence (2012).

[3] S. Hare, A. Saffari, and P. H. S. Torr, *Struck: Structured output tracking with kernels.* in *ICCV* (IEEE, 2011) pp. 263–270.

[4] S. Song and J. Xiao, *Tracking revisited using rgbd camera: Unified benchmark and baselines,* in *Proceedings of the 2013 IEEE ICCV 2013*.

[5] M. Godec, P. M. Roth, and H. Bischof, *Hough-based tracking of non-rigid objects,* in *Proc. International Conference on Computer Vision (ICCV)* (2011).

[6] A. Bugeau and P. Pérezz, *Track and cut: Simultaneous tracking and segmentation of multiple objects with graph cuts,* J. Image Video Process. (2008).

[7] A. Chandarr, M. Rudinac, and P. Jonker, *Multimodal human centric object recognition framework for personal robots,* in *IEEE-RAS International Conference on Humanoid Robots* (2014).

[8] J. Van De Weijer, C. Schmid, J. Verbeek, and D. Larlus, *Learning color names for real-world applications,* Image Processing, IEEE Transactions on (2009).

[9] M. Danelljan, F. Shahbaz Khan, M. Felsberg, and J. van de Weijer, *Adaptive Color Attributes for Real-Time Visual Tracking,* in *Proceedings of IEEE CVPR 2014*.

[10] S. C. Hoi, J. Wang, and P. Zhao, *Libol: A library for online learning algorithms,* Journal of Machine Learning Research **15**, 495 (2014).

[11] Y. Y. Boykov and M.-P. Jolly, *Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images,* (2001).

[12] A. Bhattacharyya, *On a measure of divergence between two statistical populations defined by their probability distributions,* Bulletin of the Calcutta Mathematical Society **35**, 99 (1943).

[13] M. Rudinac, *Exploration and Learning for Cognitive Robots*, Ph.D. thesis, Delft University of Technology (2013).

[14] X. Wang, M. Rudinac, and P. Jonker, *A robust real-time tracking system based on an adaptive selection mechanism for mobile robots,* in *Control Automation Robotics Vision (ICARCV), 2012 12th International Conference on* (2012) pp. 1065–1070.

6

Multiview object recognition with viewpoint correlation

In this chapter, we have developed a novel multi-view object recognition system that incorporates correlation of change in appearance of an object with its associated viewpoint change. Firstly, a benchmark is developed to compare feature descriptors describing different visual properties, without considering viewpoint correlation. Two standard datasets, RGB-D and SOIL-47 are used in this benchmark. A KdTree is trained to speed up the recognition process. A sequence of views obtained while moving a camera around an object is modelled as a string of characters in a specific order. This is achieved by a Vector Quantization algorithm using the trained KdTree. A view-discretization is performed which generates a new character in a string with every 20° of motion around an object. The generated string during training is used to create an object database to which smaller sub strings obtained in the testing phase are matched using Sequence Alignment techniques frequently used in BioInformatics. Experiments on datasets with varying sequence length show the improvement achieved by this algorithm over single view based systems using the previously developed benchmark. An unsupervised estimation of spatial relationship between different viewpoints is obtained using a fast Visual Odometry system. This is integrated with online segmentation and ground plane alignment to create a standalone system that can be applicable in real-world scenarios. Experiments on a custom developed dataset reflect the performance improvements obtained while using reference datasets.

Chapter modified from article:
Leon de Lange, Aswin Chandarr, Pieter P. Jonker, Multiview object recognition with viewpoint correlation for service robots, A bioinformatics inspired approach, submitted to Journal of Intelligent Service Robots

6.1. Introduction

An object is a 3D structure which has different geometrical and visual appearance from different viewpoints. Most methods for object recognition as discussed in chapter 4, consider an object to be collection of images which belong to same class. With this, the spatial relationship between different views are lost. Because of a robot's ability to observe an object from different viewpoints, this spatial relationship can be used as an additional modality to augment the recognition process and thereby improving recognition performance in challenging domestic environments. We term this as viewpoint correlated object recognition and Figure 6.1 illustrates this principle, where a sequence of images a, b, c is observed only when the camera is moved counter-clockwise around the object.

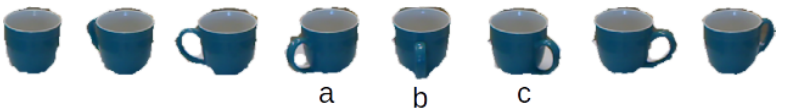


Figure 6.1: Illustration of a viewpoint correlation in an objects visual appearance

There are a limited number of algorithms in the literature that tackle the multi-view object recognition system. The *Potemkin model* [1] approximates the geometry of an object using basic geometrical shapes. Synthetic views of basic geometrical shapes are obtained from different viewpoints through projection to obtain shape primitives. Multiple views of an object are manually labelled using the generated primitives to obtain transformations of individual parts between views. An object skeleton is estimated as a combination of primitive shapes and their transformations. Any new object is projected into a skeleton to perform recognition. The Potemkin model needs training data with labelled views in order to learn an object category. This is an intensive and time consuming process and needs manual annotation of viewpoints. A *Canonical parts* based model [2] extracts unique parts which are consistent and repeat across different object views. The object model is constructed from local relations between such canonical parts that constitute an object. The extraction of canonical parts is based on clustering of repeating keypoints and this has drawbacks of inability to handle objects with low-texture and the computation and memory requirements become very high with more learned objects. A *Hypergraph* model [3] to incorporate pairwise relationships in multi-view recognition. There are set of vertices which are different views of an object and they are connected with edges comprised of clustered features from each view. A spectral clustering and transductive inference was used to apply hypergraphs for this particular problem. Though this approach models the visual features shared between different views, the spatial relationship is not considered.

There does not exist a model which explicitly considers spatial relations between viewpoints as an additional modality for recognition which performs unsupervised viewpoint relation estimation. We have developed a novel view correlated multi-view object recognition method which uses *Vector Quantization* of features combined with *Sequence Alignment* techniques. This has been integrated with *Visual*

odometry and object segmentation in-order to be used as a standalone system. Usability of such a system in run-time is considered at every step of the algorithm and care has been taken to minimize the hyper-parameters to make it applicable in different scenarios without requiring elaborate tuning. In this chapter we present such a framework that can be used with any global feature descriptors and their combinations.

6.2. Benchmarking object recognition

Given that there does not currently exist a system which explicitly correlates visual appearance with viewpoints, we have proposed to develop such a system. Since there are a multitude of feature descriptors that are suitable for different conditions, the focus lies in making a framework where any different feature descriptors and their combinations can be incorporated. In order to evaluate the performance improvements from the new multi-view system over the single view based system, we created a benchmark, where various feature descriptors are also compared.

6.2.1. Datasets

The benchmark describes a method, which evaluates the object recognition performance of different object models. In order to achieve repeatability and to reduce influences of external variables during image acquisition, we use standard object datasets. Though there exist many different datasets for evaluation object recognition, only two datasets are found to be relevant. These are SOLI-47 by Burianek *et al.* [4] and the RGB-D by Lai *et al.* [5]. We choose these specific datasets as they have systematically captured images for various commonly used household objects, along with associated viewpoint information. The use of these datasets also minimizes the influence of image noise and unequal distribution of views. The specifications of these datasets are listed in table 7.4 and sample images are shown in Figure 6.3

Table 6.1: Dataset specifications

SOIL-47 dataset

- Dataset contains cereal boxes
- 20 objects
- 21 views per object
- Fixed elevation angle
- Captures 180 degrees
- Dataset captured twice under different lighting condition

RGB-D Kinect dataset

- Dataset contains household objects
- 51 objects
- 3 times 40 - 50 views per object
- Elevation angles at 40, 45 and 60 degrees
- Captures 360 degrees

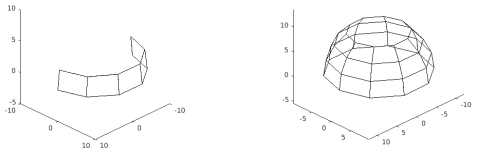


Figure 6.2: Schematic object viewpoints for SOIL-47 (left) and RGB-D dataset (right)

Both these datasets provide segmentation masks to remove the background from the images. While SOIL-47 provides good quality RGB images with higher resolution (576x720), The RGB-D dataset provides data obtained from a Kinect with VGA resolution for both RGB and Depth images, leading to an average image size of 160x100 per object. Data normalization is not applied as all data is captured consistently under equal conditions. Each object is placed at the center of a blank canvas. The center of an object is calculated from the segmentation mask boundaries (equation 6.1).

$$c_x = \frac{x_{left} - x_{right}}{2} \quad \text{and} \quad c_y = \frac{y_{top} - y_{bottom}}{2} \tag{6.1}$$

Figure 6.3: Object sample views from SOIL-47 (top) and RGB-D (bottom)

6.2.2. Compared feature descriptors

There exist a multitude of feature descriptors in image processing literature and its not practical to compare them directly. Hence, they are classified into different categories based on the kind of features they describe such as

1. Edge/Corner
2. Texture
3. Shape
4. Colour
5. Multi scale abstraction

One descriptor from each of these categories are used for our object recognition benchmark. All these descriptors are obtained from *vlfeat* MATLAB toolbox [6] and are listed below

1. Scale Invariant Feature Transform (SIFT)
2. Histogram of Oriented Gradients (HoG)
3. Local Binary Patterns (LBP)
4. Hue, Saturation and Value (HSV)
5. Deep Convolutional Neural Network (CNN)

SIFT [7] is a local keypoint detector and descriptor based on a local gradient distribution. Keypoints are described using histograms of magnitude and orientations of gradients in a local neighbourhood. Each keypoint is described by a vector of dimension 128. An object is then described as a collection of many different keypoints and this number varies for every image. The rest of the features used are global descriptors where entire image is represented with a single vector.

HoG [8] is a global operator that describes gradient distribution over the entire image. The bins are concatenated leading to vector of dimension 279. LBP [9] describes texture of an image with illumination and rotation invariance and with 58 uniform binary patterns, it generates a feature vector of size 522. HSV histograms are used to describe color distribution in an image. Hue, Saturation and Value are quantized and a histogram is constructed and concatenated with mean and standard deviation as in [10] to obtain a feature vector of 164 dimensions.

Deep convolutional neural networks have recently become quite popular for image understanding. These deep neural networks require a very large training dataset to learn the underlying features, restricting their use in domains having less training data. It has been shown with *transfer learning* [11] that a deep neural network trained on extensive data, can be used as feature descriptor in different applications. In this case, we use one of the popular, pre-trained networks from Simonyan and Zisserman, *imagenet-vgg-verdeep-16* [12] whose structure is shown in Figure 6.4

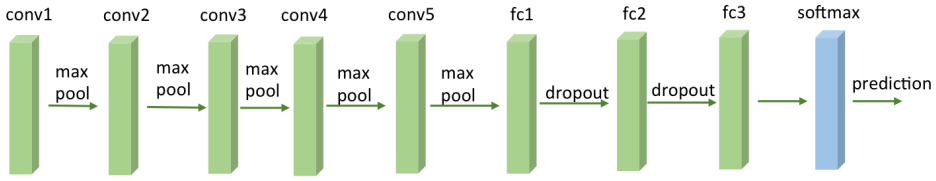


Figure 6.4: Schematic layer overview of imagenet neural network

This neural network consists of 16-layers. A stack of convolutional layers with varying depth is followed by three fully connected layers. Two of those layers have 4096 channels each and the third performs a 1000-way ILSVRC classification. The

final layer is a soft-max layer, which assigns a class to an input image. A feature vector is obtained from the second fully connected layer as a 1000 dimensional vector. The neural network input layer requires a 224×224 RGB image.

6.2.3. Fast matching with Kd-tree

After obtaining object data and extracting feature vectors, the final step is classification which assigns every image to a previously trained class. There exist many different algorithms to perform classification, ranging from *K-Nearest neighbour (KNN), Support Vector Machines (SVM), Decision trees, Neural network,* etc. The objective of the benchmark is not to obtain the best recognition performance on a particular dataset, but to compare the performance of a multi-view based recognition model with a single-view model that can applied in varying circumstances. Hence a *Nearest Neighbour* search based on an Euclidean distance measure (Equation 6.2) is used as a classifier to avoid performing hyper-parameter tuning.

$$d_{eucl}(p,q) = \sqrt{\sum_{i=1}^{n}(q_i - p_i)^2} \tag{6.2}$$

Where p and q are feature vectors.

In the first step (training) an object database is created by stacking feature vectors of all the available images and tagging them with their object class names. During the testing phase, a feature vector is extracted from a query image and is compared to all the vectors in the dataset. The class of the vector with the least distance is assigned to the query image.

Finding the closest feature vector in a large database is computationally demanding. In order to quickly solve this nearest-neighbour problem a Kd-tree is used. This is a hierarchical structure built by partitioning the data recursively along the dimension of maximum variance.

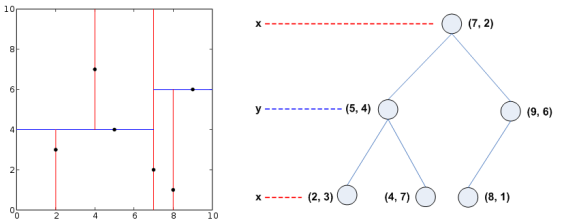


Figure 6.5: Kd-tree decomposition for the pointset (2,3), (5,4), (9,6), (4,7), (8,1), (7,2)

Figure 6.5 illustrates an example Kd-tree. The median along the first dimension splits the data in two partitions at the point (7,2). From both remaining partitions median points are selected along the second dimension. A horizontal line is drawn through the points (5,4) and (9,6). This process is repeated until no points are left. A new query point can be classified in a maximum of 3 steps by this Kd-tree, which

is more efficient than evaluating each of the 8 points. Figure 6.6 shows the execution time of HOG feature matching with varying size of the database. The speed improvement becomes more pronounced also with the increasing dimensionality of matched feature vectors.

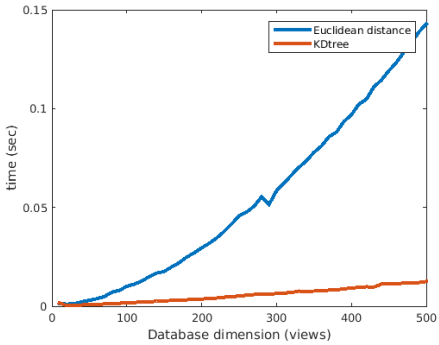


Figure 6.6: Euclidean vs Kd-tree features matching time with varying database size for a 256 dimensional feature vector

SIFT is a local descriptor, which describes (multiple) edges and corners within an image. Matching is done by comparing all features from one image to the features of another image. Each feature vector of an image is matched against all feature vectors from another image using Euclidean the distance. A distance threshold determines if there is a true match between two vectors.

6.2.4. Benchmark performance

To evaluate the recognition performance, variability in the visual appearance is introduced through Gaussian noise, affine transformations and illumination changes as used in [10]. These are comparable to the conditions encountered during object recognition in real world circumstances. A Gaussian noise with $\sigma = 5\%$ and an affine transformation with 10 degree rotation is applied. The SOIL-47 dataset already presents images captured under two different lighting condition, and one set of these images are used as lighting manipulated query images. Figure 6.7 shows the result of manipulating an image in different ways to generate query images.

Figure 6.7: Original image, Image manipulation: Gaussian, Affine, Illumination

Object class recognition

Given a query object, a sequence of views is selected for object recognition, but in this case, each query view is matched individually against the complete object dataset for object classification. Figure 6.8 shows a query sequence of three views (bottom) with corresponding best matches indicated by the arrows.

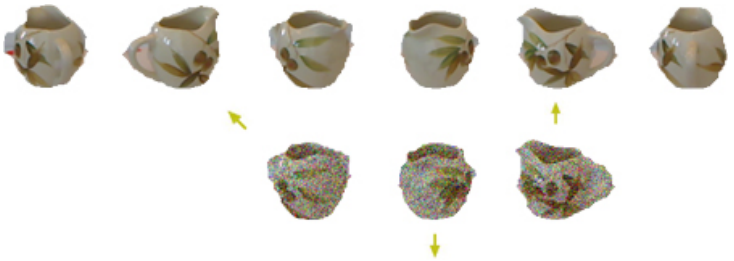


Figure 6.8: A query sequence (bottom) is matched against all object models. In this example 2 out of 3 views correctly recognize the object

The object recognition performance is measured systematically by increasing the query sequence length. The location of sequences in query objects are determined randomly. Figure 6.9 shows the results of single-view object recognition for the SOIL-47 and RGBD-dataset. The object recognition rate is determined as the individual queries which correctly recognize an object divided over all query images.

$$\text{Recognition rate} = \frac{\text{Correctly classified images}}{\text{Total number of query images}} \tag{6.3}$$

Quality and object diversity of both datasets have influence on the descriptor performance. Objects from SOIL-47 contain more details compared to the objects of the RGB-D dataset, therefore more SIFT keypoint are detected and a higher object recognition rate is achieved. Dataset SOIL-47 contains similar looking cereal boxes. This affects the object recognition performance of HOG and LBP descriptors, which are obtained directly from the pixel values in a small neighbourhood.

It can be seen from Figure 6.9 that, if considered individually, the recognition performance is not influenced much by the query sequence length. This shows the limitation of a single view based recognition system. It can also be seen that color and texture descriptors are quite immune to affine transformations. Based on this benchmark results, it is chosen to compare and evaluate novel multi-view object models by the RGB-D object dataset in combination with Gaussian data manipulation. The RGB-D dataset contains diverse household objects with 360 degree views, which is preferred over the similar looking cereal boxes of the SOIL-47 object dataset.

Object view recognition

In the development of a viewpoint correlated object model, every image will be abstracted to a single dimensional value through Vector Quantization (VQ) as explained in section 6.3. Given that two close views of an object can have the same vi-

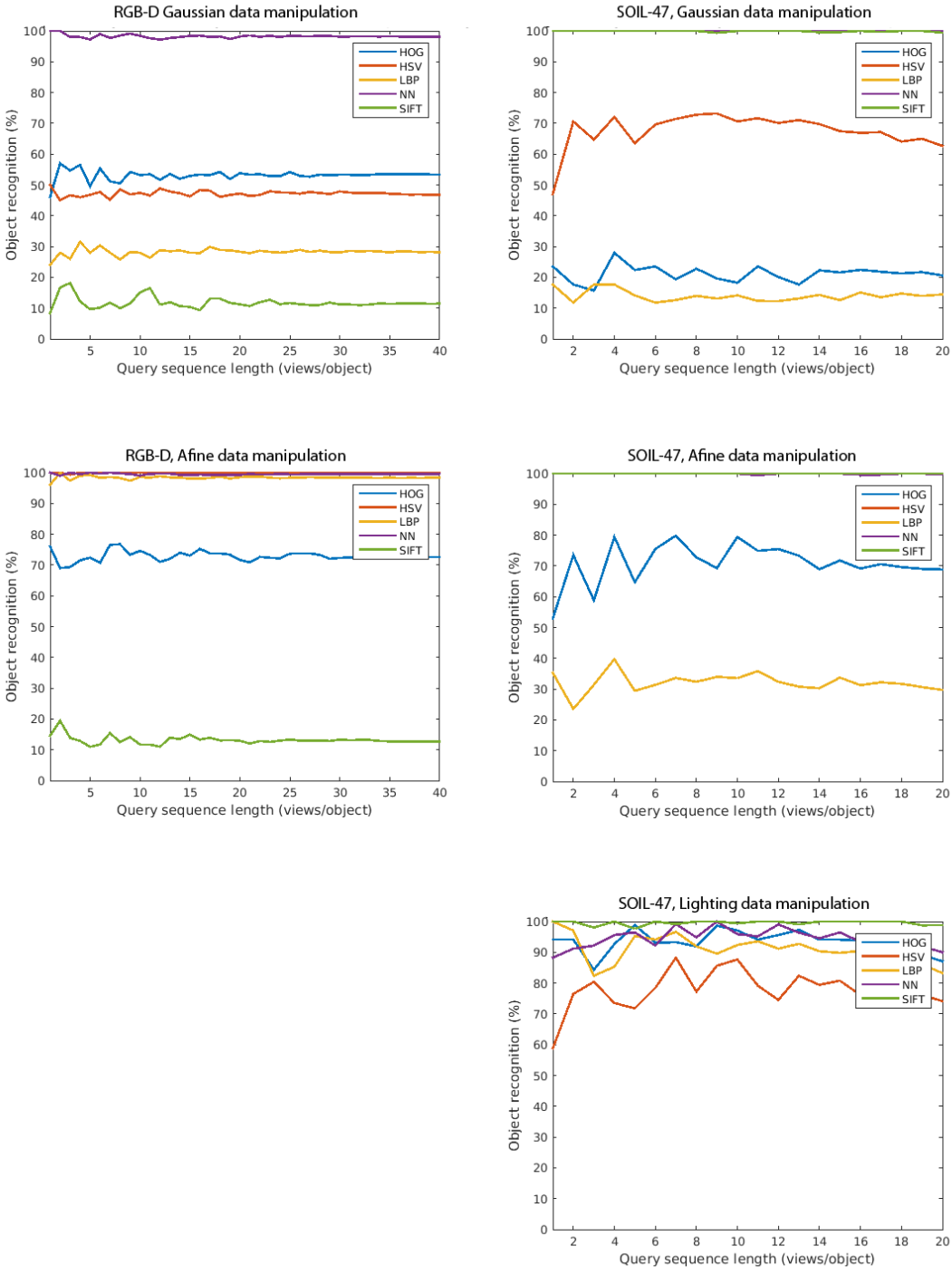


Figure 6.9: Single-view object recognition results. The left plots are obtained from the RGB-D dataset (50 objects, 40 views) and right plots from SOIL-47 dataset (17 objects, 20 views). The σ of Gaussian data manipulation is set to 5% and an affine rotation of 10 degrees is used

sual appearance, leading to redundant information and decrease recognition speed, a *View discretization* (Figure 6.10) is introduced. In order to find the optimal discretization, experiments are performed by evaluating recognition of a view while changing the discretization angle.

Figure 6.10: Data discretization on two scales with equally spaced views

Figure 6.11 shows the results of View recognition over Gaussian data manipulation. The performance is obtained as

$$\text{View performance} = \frac{\text{Number of correct views}}{\text{Number of correctly recognized objects}} \tag{6.4}$$

The ratio of correct views/objects depicts the chance a view is correctly recognized given that the corresponding object is recognized. Both plots show a ratio decrease for small discretization angles indicating that the recognition of single-views is harder for small view spacing. The optimal view spacing is determined by a high gradient of the curve in Figure 6.11 and is chosen as 20 degrees to be used in multi-view object recognition model.

The descriptor matching time is obtained from the single-view object recognition experiment. Figure 6.12 shows the matching time of one query object to all objects in a dataset. There exists a linear relationship between the descriptor matching time and the query sequence size. The matching time of SIFT descriptors is higher, because each image contains multiple descriptors.

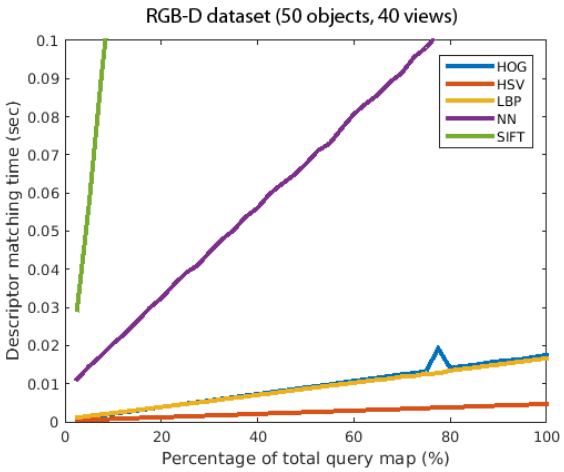
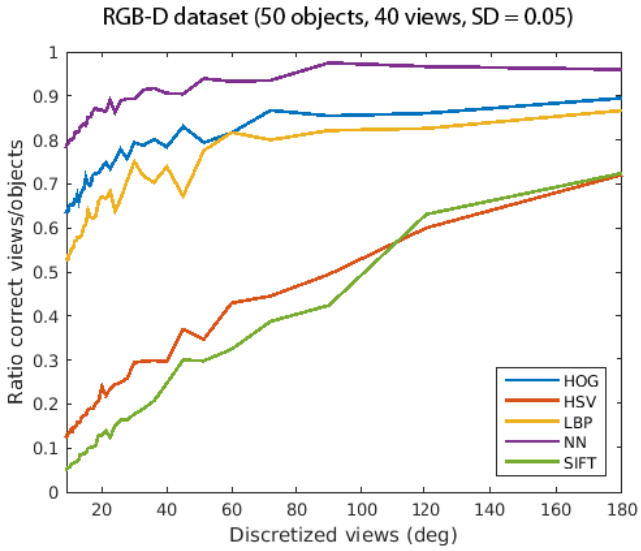
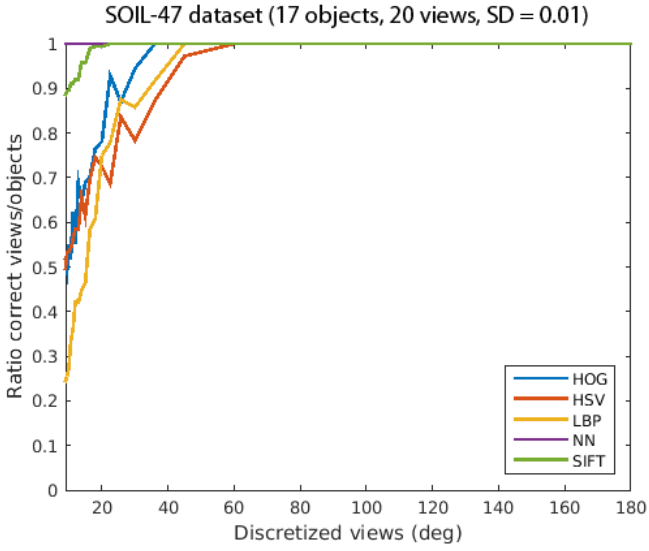


Figure 6.12: Descriptor matching time



(a) View recognition in RGB-D Dataset



(b) View recognition in SOIL-47 dataset

Figure 6.11: View recognition using Gaussian data manipulation

6.3. Multiple view recognition system

A novel multi-view object recognition system incorporating viewpoint relation between successive images is introduced and compared against the benchmark created in the previous section.

An object database (training data) is created with appearance and viewpoint data from the chosen dataset (RGB-D or SOIL-47). An object recognition model based on *Sequence Alignment* is constructed with this training data. A small sequence of data is chosen at random from the dataset, artificial manipulations are applied and then are used as input to the developed model to test the object classification performance. The length of query sequence is varied and the recognition performance is compared with results from single view object recognition.

In order to develop the model, images from datasets (RGB-D and SOIL-47) are used, as they contain systematically captured image sequences, with known rotation angle between them. While the performance of the developed multi-view object recognition system is independently evaluated in this section, the estimation of viewpoint relation from a free moving camera and its integration into the recognition model are discussed in successive sections.

The crux of the proposed method is to represent a 3D model of an object as a string (sequence) of discrete numbers which are ordered according to their relative viewpoint change. Any query is small sequence of ordered numbers and the object classification is performed based on Sequence Alignment.

6.3.1. Sequence Alignment

Sequence Alignment is a tool to discover patterns in a set of comparable sequences. This is widely used in the field of BioInformatics to find parts of matching sequences in DNA, RNA and Proteins. All these complex biological molecules comprise of long sequences of repeating elements from their corresponding *'Alphabet'*. For instance, the DNA is a sequence of letters $\{A, T, C, G\}$ each of which represents a nucleotide base. The proteins are more complex molecules which are made up of 20 types of amino acids. The main application of *Sequence Alignment* in these cases is to identify similar parts in different sequences or match a new sequence to one of the known ones. Given that similar regions in these molecules, correspond to similar functions, this is a powerful tool to identify the functionalities in a new sequence and to study evolutionary relationship between various species and also finds its application in *Human Genome* sequencing [13].

Beyond BioInformatics, sequence alignment can be used on any information that can be represented in a form of sequence of finite set of symbols. This section provides a brief introduction into *Pair-wise* sequence alignment using words of English alphabet. Equation 6.8, shows an example where parts of words are aligned. While *Global* sequence alignment attempts to align two sequences over their entire length (leading to forced gaps in between), a *Local* sequence alignment finds regions of similarity between sequences. In the case of multi-view object recognition where a small sequence of views will have to be matched with a dataset of larger number views, a *local* alignment process is suitable.

The dynamic programming approach [14] is slower compared to other methods

based on Heuristics[15] or a Probabilistic approach [16], but it is guaranteed to provide the optimal matching solution. Given the fact that the length of sequences and number of objects in our application is very small compared to the sequences faced in Bio-Informatics, a Dynamic programming approach was employed here.

A local sequence alignment algorithm based on Dynamic programming was first proposed by Smith and Waterman [14], uses a Score matrix (H) and a Trace back matrix (T) along with a scoring function to estimate the matching parts of sequences. Both these matrices are of dimension (mXn) where m and n are lengths of individual sequences to be matched. The score matrix (H) is initialized with 0 on the first row and column.

$$H(i,0) = 0, 0 \leq i \leq m, \quad H(0,j) = 0, 0 \leq j \leq n \tag{6.5}$$

This matrix is now incrementally filled by obtaining a score for each position based on the values of its neighbours towards the top and left side using 6.6

$$H(i,j) = max \begin{Bmatrix} 0 & \\ H(i-1,j-1) + s(a_i, b_j) & Match/Mismatch \\ max_{k \geq 1} \{H(i-k,j) + w\} & Gap \\ max_{l \geq 1} \{H(i,j-l) + w\} & Gap \end{Bmatrix}, 1 \leq i \leq m, 1 \leq j \leq n \tag{6.6}$$

A scoring function $s(a,b)$ is a measure of similarity based on the alphabet and can assign scores to $\{Match, Mismatch\}$ between the elements a, b of the sequence and w is a gap penalty.

Consider an example case of matching two strings "signal" and "align". Equation 6.7 shows the score matrix H for these sequences where matches are rewarded with +2, mismatches -1 and gaps -0.5. The position from which the maximum condition of equation 6.6 for each position in $H(i,j)$ is obtained, is indicated by an arrow in the trace back matrix T.

$$H = \begin{pmatrix} & - & s & i & g & n & a & l \\ - & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ a & 0 & 0 & 0 & 0 & 0 & 2 & 1.5 \\ l & 0 & 0 & 0 & 0 & 0 & 1.5 & 4 \\ i & 0 & 0 & 2 & 1.5 & 1 & 1 & 3.5 \\ g & 0 & 0 & 1.5 & 4 & 3.5 & 3 & 3 \\ n & 0 & 0 & 1 & 3.5 & 6 & 5.5 & 5 \end{pmatrix} \quad T = \begin{pmatrix} & - & s & i & g & n & a & l \\ - & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ a & 0 & 0 & 0 & 0 & 0 & \nwarrow & \leftarrow \\ l & 0 & 0 & 0 & 0 & 0 & \uparrow & \nwarrow \\ i & 0 & 0 & \nwarrow & \leftarrow & \leftarrow & \uparrow & \uparrow \\ g & 0 & 0 & \uparrow & \nwarrow & \leftarrow & \leftarrow & \uparrow \\ n & 0 & 0 & \uparrow & \uparrow & \nwarrow & \leftarrow & \leftarrow \end{pmatrix} \tag{6.7}$$

To obtain the best aligning position, the $Max(H)$ element is selected and its predecessor elements are traced back until a score of 0 is reached. For the example sequences 'signal' and 'align' the best alignment is obtained as following with a total score of 12 (6+4+2).

$$\begin{matrix} \_ & s & i & g & n & a & l \\ a & l & i & g & n & \_ & \_ \end{matrix} \tag{6.8}$$

When there are multiple positions having the highest score are possible, alignments from each of these positions are obtained and the total scores are compared to identify the best match.

A pair-wise local alignment of a new sequence with all the sequences in the database is performed to obtain the best matching sequence. In case of multi-view object recognition, each object is represented as a string of numbers, with each number encoding the visual appearance of an object from a different viewpoint.

6.3.2. Appearance Quantization

As described earlier, a Feature vector (FV) is an abstract representation of visual appearance of an object. Hence a 3D object can be represented as a sequence of feature vectors with adjacent viewpoints. Given that each FV is a high dimensional vector, finding an alignment based on these high dimensional data is computationally intractable. Hence, an additional abstraction which converts these high dimensional FV to a single dimensional value is used. This abstraction is similar to the system used in Bag of Words (BoW) [17], where every local keypoint descriptor is converted into a single dimension using dictionary learning. The method used in BoW performs clustering of similar feature vectors and assigns the cluster number to each FV. The cluster assignment by this algorithm being highly sensitive to the *number of clusters* parameter, can have a high intra-cluster variability in the FVs. Since the views with very close visual appearance are already filtered with the viewpoint discretization (subsection 6.2.4), we require a method to attribute a different single dimensional value to the visual appearance from every different viewpoint. Hence a fast and parameter free *Vector Quantization (VQ) coding* method based on *Kd-tree* is used. This utilizes the same *Kd-tree* trained to decrease the computational demand during Euclidean distance estimation between the various FV's in a single view object recognition model of the benchmark 6.2.3.

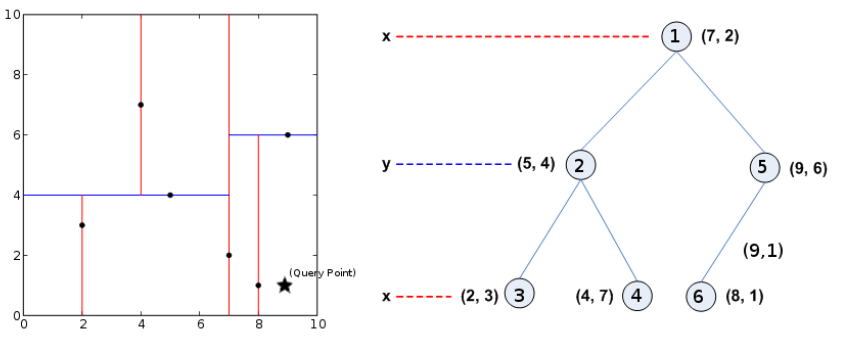


Figure 6.13: Illustration of vector quantization using Kd-tree nearest neighbour

Provided a Kd-tree is constructed with training (reference) data, Vector Quantization (VQ) encoding involves assigning a number to a newly encountered query FV. This is done by performing a *Nearest Neighbour* search of a query point and assigning it the index of its closest neighbour in the tree. In the example in Figure 6.13, a query data point of (9,1) will be assigned *6* which is the index of its closest neighbour (8,1) in the tree.

In this manner, every viewpoint is represented as a single dimensional value and a 3D object at every elevation is represented as a sequence of these numbers. Now having obtained a complete sequence for every object, any new sequence of images can be recognized by aligning them with sequences in the database.

6.3.3. Object model for multi-view object recognition

The object recognition performance is measured using the feature vectors, Kd-tree and single value representation of objects. A query view sequence is represented by single values and compared to each database object for optimal alignment. For each object alignment the total Euclidean distance is calculated between feature vectors of the aligned view sequence and query view sequence. The object that contains the alignment with minimal Euclidean distance is selected for object recognition. Algorithm 1 shows the pseudo code for sequence alignment based object recognition.

```
Data: Query views, output Algorithm 1
Result: Object recognition
begin
    for each query view do
        Extract features using a descriptor;
        Represent query view by a single value based on the feature vector
        location in Kd-tree;
    end
    for each object in database do
        if if the object and query object share equal single values then
            Find optimal alignment position;
            Calculate the total Euclidean distance between the aligned (query)
            object feature vectors;
        else
            Evaluate next object;
        end
    end
    The object with smallest total Euclidean distance is selected for object
    recognition
end
```

Algorithm 1: Object recognition sequence alignment algorithm

Algorithm performance

A flow diagram of the proposed algorithm is illustrated in figure 6.14. The object recognition performance is evaluated using the benchmark described in section 6.2.4. The proposed algorithm uses global feature vectors, which are represented by single values. For this reason the local feature descriptor SIFT cannot be evaluated as there are multiple features per view.

Sequence alignment compares two strings for optimal alignment. Objects cap-

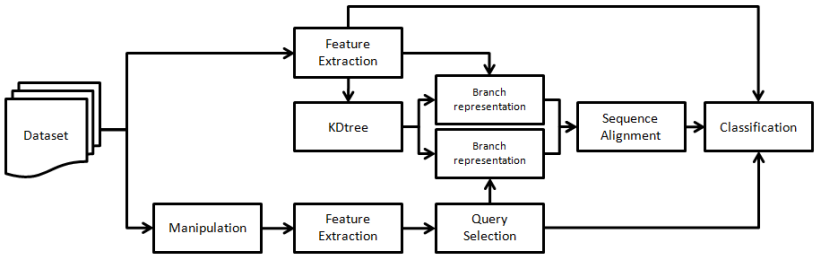


Figure 6.14: Flow diagram sequence alignment based object recognition

tured at multiple elevation angles are for this reason split and evaluated at each elevation angle. When an optimal alignment is found between two view sequences the total Euclidean distance is obtained for the complete query object and database object as described in algorithm 1.

Figure 6.15 shows the comparison of single-view and multi-view sequence alignment object recognition. Both plots are obtained from the same (manipulated) object data from the RGB-D dataset. Feature descriptors NN, HOG and LBP show an increased recognition performance for increasing query sequence length. In case of the color descriptor HSV the recognition performance is slightly decreased with respect to single-view object recognition. This difference can be declared by evaluating the view recognition rate shown in table 6.2. In case of HSV only 4.5% of all views are correctly recognized. Sequence alignment is only beneficial compared to single-view object recognition when views are correctly recognized.

Table 6.2: View recognition rate (40 views) per descriptor

Descriptor	HOG	HSV	LBP	NN
View recognition rate	32.4%	4.5%	15.2%	72.6%

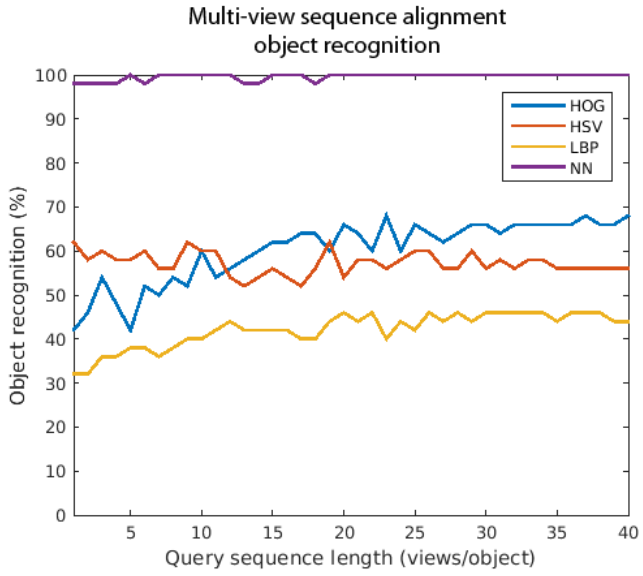
The descriptor matching time for single-view and multi-view sequence alignment object recognition is shown in figure 6.16. In multi-view sequence alignment object recognition finding the optimal alignment is responsible for the additional execution time.

6.4. Online relative viewpoint estimation

An object recognition model incorporating the viewpoint relationship between different visual appearances of an object was introduced in the previous section. The algorithm has been developed and evaluated using standard object databases which also contain the viewpoint information for every image. In order to use this algorithm in practice, an online relative viewpoint estimation is required, which is further discussed in this section.

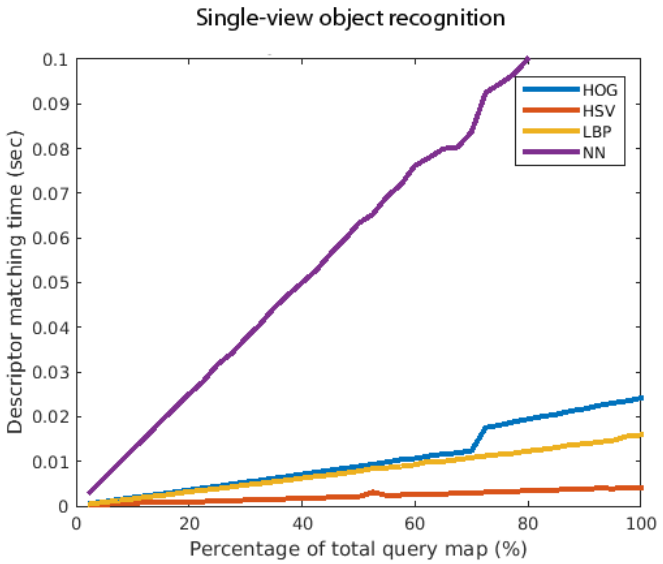
When an object is being explored by a camera system attached to a robot (either in the head of the robot or as an eye in hand camera system), it is possible

(a) Single view performance

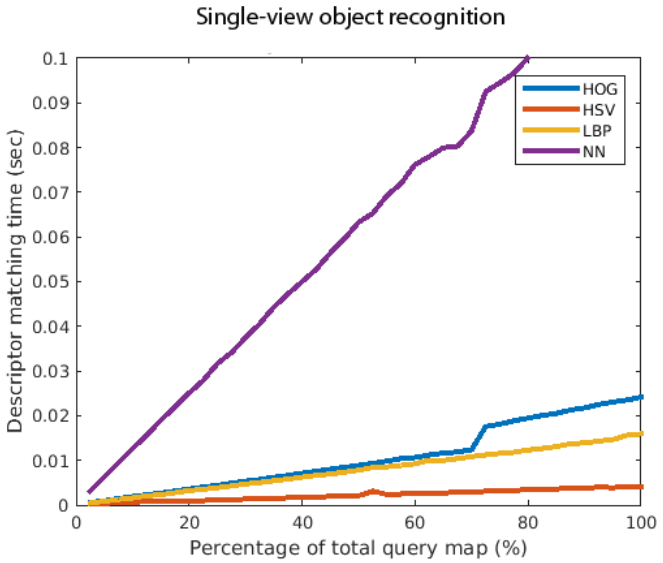


(b) Multiview performance

Figure 6.15: Object recognition RGB-D dataset and Gaussian data manipulation (50 object, 40 views, $\sigma = 0.01$)



(a) Single view recognition time



(b) Multiview recognition time

Figure 6.16: Descriptor matching time RGB-D dataset(50 object, 40 views). The increased slope is a consequence of sequence alignment time.

to estimate the relative motion by propagating the individual joint motion recorded by the encoders through the forward kinematic chain till the camera link. But there are inaccuracies accumulated over every joint due to limitations arising from inexpensive hardware. This leads to drift in the estimated values which are significant, especially in case small and continuous rotations as encountered in this scenario. Hence a method to obtain this relative viewpoint change, based on visual data is preferred and used. Apart from being applicable to cases when the user is showing different views of an object to the robot, a purely visual information based relative viewpoint estimation can make a stand-alone multi-view recognition system.

6.4.1. View Registration methods

The alignment of an image pair with shared visual and depth information is called view registration. One of the assumptions used in this study has been that the geometry of an object does not change over time. This implies that view registration can be obtained as a rigid transformation with rotation matrix R and translation vector T (figure 6.17).

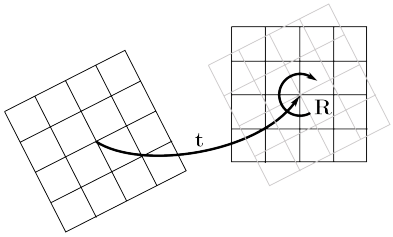


Figure 6.17: Rigid view transformation

The methods to obtain view registration can be classified into

1. Global Registration
2. Local Registration (Visual odometry)

Global registration methods can estimate the transformation between any two arbitrary views as long as there is a shared visual appearance between them. Global registration of point clouds via iterative closest point (ICP) was first described by Besl and McKay [18] and this works by minimizing the mean squared error of common points of two arbitrary rotated and translated point clouds. Because of their ability to handle large transformations, this class of algorithms come with a limitation in low speed and high computational requirements.

Local registration methods also known as *Visual odometry/ ego-motion*, estimate small relative transformations between two frames, generally successive images in a video sequence. The work of Fang and Zhang [19] gives an evaluation of RGB-D visual odometry methods which are divided in three categories according to what kind of data mainly is used. Image-based, depth-based and hybrid-based methods.

Huang *et al.* [20] uses sparse image features (i.e SIFT, SURF) to find correspondences between consecutive frames. Depth information of the feature locations is used to obtain a transformation matrix by minimizing the re-projection error (figure 6.18). Kerl *et al.* [21] proposed an alternative dense feature approach, which uses all pixels within a frame to estimate the transformation matrix. This method assumes a world point observed by two frames has the same brightness. The goal is obtaining a transformation that best satisfies the photo-consistency constraint over all pixels.

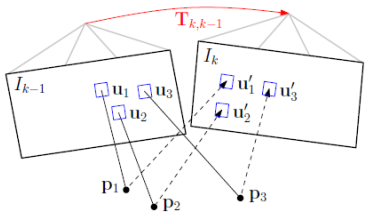


Figure 6.18: Optimization by minimization of re-projection error

An extension of the sparse feature based method detects and extracts planes, lines or points. Tang *et al.* [22] use these depth features to estimate a transformation matrix by minimizing the re-projection error between features of two consecutive frames. Registration of a dense point cloud is possible as described earlier by the ICP algorithm.

In hybrid-based methods an initial registration guess is obtained from sparse image features. Dense point registration techniques are used to refine the registration ([23], [24]).

6.4.2. Fast ego motion estimation

In order to obtain fast estimation of relative transformation, visual odometry performs linearization of the motion model. This introduces minuscule motion errors, which become significant only after prolonged motion. Given that our application deals with short range motions, we chose to adapt a lightweight image-based sparse visual odometry, partly based on the work done by Lui *et al.* [25].

The first step is selection of points in the frames to be aligned. 2D Keypoints are extracted by approximation of the image Laplacian as used in corner detectors [26]. Points can either be matched or tracked across multiple frames to form point pairs. Depth value of the detected keypoints is used to obtain their real world locations in $[x, y, z]$. The algorithm of Lui *et al.* [25] obtains egomotion by considering small rotations and translations, such that $sin(\theta) = \theta$ and $cos(\theta) = 1$. The relationship between matched features in consecutive frames is expressed as

$$p_{t+1} = [\delta R|\delta T]^{t+1} p_t \tag{6.9}$$

Where $M = [\delta R|\delta T]$ is the 6DoF small rotation and translation approximation model

$$M = \begin{bmatrix} 1 & -\theta_z & \theta_y & t_x \\ \theta_z & 1 & -\theta_x & t_y \\ -\theta_y & \theta_x & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.10}$$

The motion model in equation 6.9 is rewritten using linear velocities. The matrix Jacobian is obtained from partial derivatives of M with respect to the unknown motion parameters $\Theta = (t_x, t_y, t_z, \theta_x, \theta_y, \theta_z)$.

$$p_{t+1} = J\hat{\Theta} + p_t \tag{6.11}$$

With point pair Jacobian

$$J_p = \begin{bmatrix} 1 & 0 & 0 & 0 & z & -y \\ 0 & 1 & 0 & -z & 0 & x \\ 0 & 0 & 1 & y & -x & 0 \end{bmatrix} \tag{6.12}$$

The total Jacobian is constructed by stacking each point pair on top of each other such that $J = [J_{p,1}, J_{p,2}, ..., J_{p,n}]$. Rewriting equation 6.11 gives an estimation of the motion parameters.

$$\hat{\Theta} = (J^T J)^{-1} J^T \begin{bmatrix} p_{1,t+1} - p_{1,t} \\ p_{2,t+1} - p_{2,t} \\ \dots \\ p_{n,t+1} - p_{n,t} \end{bmatrix} \tag{6.13}$$

Where $(J^T J)^{-1} J^T$ is the pseudo inverse of J. From the estimated motion parameters equation 6.9 is updated and iterated until the projection error falls below a threshold

$$p_t - [\delta R|\delta T]^{t+1} p_t \leq threshold \tag{6.14}$$

The least squares solution presented above is not robust in the presence of outliers. A M-estimator is used to reduce the influence of outliers on motion estimation by assigning weights to each point pair.

The absolute distance of a point pair is calculated via

$$\Delta r = \sqrt{\sum \left(\begin{bmatrix} x_{t+1} & y_{t+1} & z_{t+1} \end{bmatrix}^T - \begin{bmatrix} x_t & y_t & z_t \end{bmatrix}^T \right)^2} \tag{6.15}$$

The standard deviation σ of all point pair distances is calculated and used to assign weights to each individual point pair

$$W = 1 - \frac{\Delta r^2}{\sigma^2 + \Delta r^2} \tag{6.16}$$

Adding the weights to equation 6.13 gives

$$\hat{\Theta} = (J^T W J)^{-1} J^T W \begin{bmatrix} p_{1,t+1} - p_{1,t} \\ p_{2,t+1} - p_{2,t} \\ \dots \\ p_{n,t+1} - p_{n,t} \end{bmatrix} \tag{6.17}$$

The camera's pose at T^{t+1} relative to its starting pose is updated from its previous pose, T^t by some small incremental motion described by $M(\hat{\Theta})^{t+1}$ with $T^0 = I$ a 4x4 identity matrix.

$$T^{t+1} = M(\hat{\Theta})^{t+1} T^t \tag{6.18}$$

This equation expresses frame T^{t+1} in the coordinates of T^t. The camera location in world coordinates is obtained by the matrix inverse $(T^{t+1})^{-1}$.

6.4.3. Performance

The accuracy of the egomotion algorithm is measured by comparing the estimated camera motion to the real camera motion. A rope is attached to the RGBD (Kinect) camera and a fixed point in the scene, which introduces a constraint on the camera motion. Figure 6.19 shows the estimated motion versus the real camera motion for repeated movement over $\pi/4$. Violation of the linearized motion model by rapid or complex camera movements introduces small motion errors. These errors are accumulated in each frame and causes deviation of the estimated motion compared to the real camera motion.

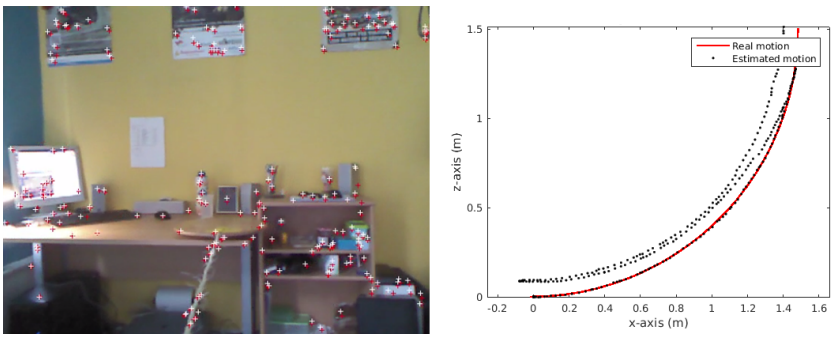


Figure 6.19: Estimated egomotion with a circular constraint

Viewpoint correlated object models will be created from the estimated egomotion. A hand held Kinect camera is moved around an object on three different elevation angles. The estimated egomotion from this experiment is visualized in figure 6.20. This figure clearly shows three circles on separate heights, despite the accumulated motion error. The results indicate that it is possible to construct a viewpoint correlated object model similar to the models from the RGB-D object dataset.

The integration of the estimated ego-motion into a stand-alone multi-view object recognition system is explained in the following section.

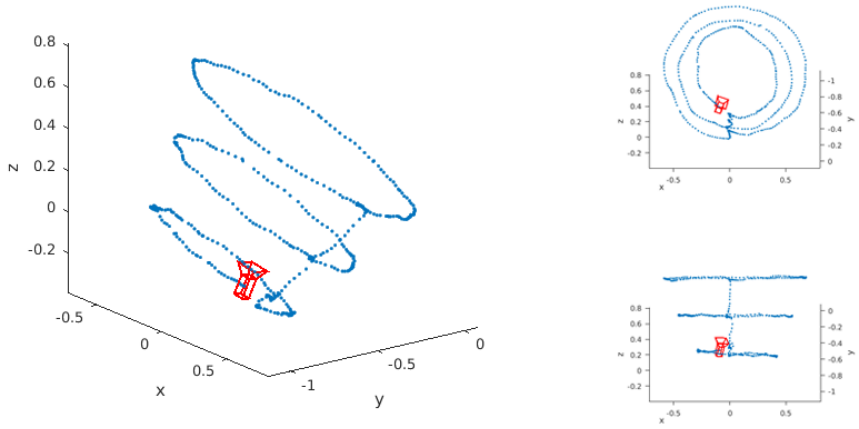


Figure 6.20: Egomotion estimation of three rotations on different heights

6.5. Integration

In order to use the sequence alignment based recognition system in real world and to be evaluated on a custom dataset, few more steps such as object segmentation and initial camera orientation have to be integrated. This section details these algorithms and integration with viewpoint estimation and multi-view recognition. Based on these a custom dataset is created and recognition performance is evaluated over arbitrary positioning and changing illumination.

6.5.1. Object segmentation

Object masks were available along with the datasets in RGB-D and SOIL-47. But they have to automatically determined during runtime, if this system has to be used in real life scenarios. Since most household objects are placed on planar support surfaces, the most dominant plane is identified using depth data and objects on top of this plane are segmented to be recognized and learned.

RANSAC [27] is the most commonly used algorithm to estimate planes in depth data. This has a drawback of being slow and computationally demanding. In accordance to the goal of having entire system working in run-time, we use a faster approach [28] which directly operates on a plane equation by segmentation in normal space followed by refinement in distance space. A plane is a collection of points, which satisfy the equation

$$n_x x + n_y y + n_z z + h = 0 \tag{6.19}$$

This contains two parameters, orientation (n_x, n_y, n_z) and distance (h) with respect to camera origin. The orientation is estimated by clustering surface normals using 3D voxel grid and merging closely related clusters. The most dominant cluster is selected as the required plane.The left plot in figure 6.21 shows segmentation in normal space.

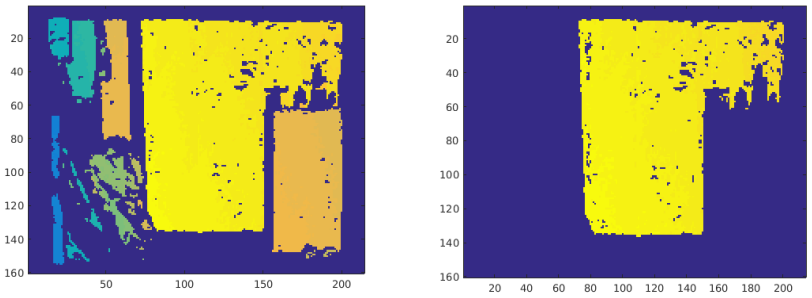


Figure 6.21: Left: plane segmentation in normal space, Right: refinement in distance space

The remaining unsolved variable in equation 6.19 is the plane height, which is solved by refinement in the distance space. The points after initial segmentation represent multiple planes with similar normal orientation. For each point the shortest distance is calculated to a plane through the origin with a dominant surface normal found in the previous step.

$$h = p_x n_x + p_y n_y + p_z n_z \tag{6.20}$$

A histogram with edges of 0.025m is created from the initial segmented data, which ranges from 0 to 2 meter. The highest edge count of the histogram is selected as the dominant plane height (figure 6.21). Segmented object data is obtained by evaluating all points in front of the found dominant plane.

$$p_x n_x + p_y n_y + p_z n_z - h < -0.025 \tag{6.21}$$

The resulting object mask can contain noise, which is reduced by the morphological filers opening and border fill. Figure 6.22 shows segmented object data results.

Figure 6.22: Segmented object data representing bananas, milk carton and a shoe

6.5.2. Camera alignment with ground plane

The segmented object data is used in combination with the estimated egomotion to construct viewpoint dependent objects. The egomotion estimation as described in section 6.4 is initialized with a 4x4 identity matrix. This implies no rotations or translations from camera body perspective. In other words the camera is located at the center of the world frame. Alignment of the dominant plane normal to the world

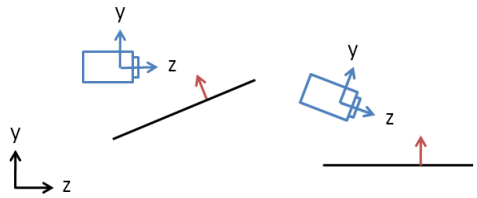


Figure 6.23: Camera orientation before and after alignment of the ground plane with the world y-axis

y-axis, gives the relative camera orientation with respect to the plane. A schematic representation of this process is shown in figure 6.23.
The angle between the world y-axis and plane normal is obtained via

$$\phi = \arccos\left(\begin{bmatrix} n_x & n_y & n_z \end{bmatrix}^T \cdot \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}^T\right) \tag{6.22}$$

Construction of a rotation axis by taking the cross product of two vectors

$$v = \begin{bmatrix} n_x & n_y & n_z \end{bmatrix}^T \times \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}^T \tag{6.23}$$

The Rodriguez rotation formula is used to compute a rotation matrix of the rotation ϕ over axis v

$$r = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} + \sin(\phi) \begin{bmatrix} 0 & -v_z & v_y \\ v_z & 0 & -v_x \\ -v_y & v_x & 0 \end{bmatrix} + (1 - cos(\phi)) \begin{bmatrix} 0 & -v_z & v_y \\ v_z & 0 & -v_x \\ -v_y & v_x & 0 \end{bmatrix}^2 \tag{6.24}$$

The initial camera position is set as the height between plane and camera as used in distance refinement in plane height estimation. The alignment is completed by replacement of the initial motion matrix with the aligned motion matrix.

$$T^0 = \begin{bmatrix} r & r & r & 0 \\ r & r & r & h \\ r & r & r & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.25}$$

The viewpoint variables azimuth and elevation are easily obtained from the estimated egomotion matrix. Elevation is the angle between camera y-axis and the world y-axis, azimuth is the camera rotation angle around the world y-axis. The elevation and azimuth angles are discretized for the construction of object models at respectively 10 and 20 degrees. Figure 6.24 shows the result of a captured object model.

6.5.3. Integrated multi-view object recognition

Finally, object recognition performance in real world scenario is evaluated by integrating all the developed components. RGB and depth data of objects are captured by moving a hand held Kinect camera around it. Viewpoint dependency is achieved

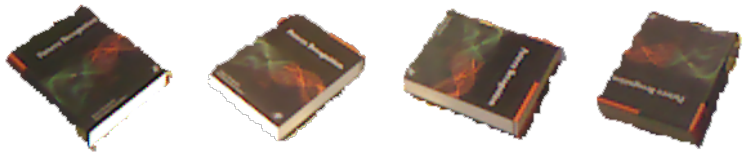


Figure 6.24: Hand held Kinect object model data
Elevation angle 40-50, Azimuth angles 0, 90, 180, 270

by finding the relative viewpoint with respect to the object via egomotion estimation. Segmented object data is stored at discretized viewpoint angles of $20\,\mathrm{deg}$. Figure 6.25 shows the schematic overview for the construction of novel viewpoint dependent object model.

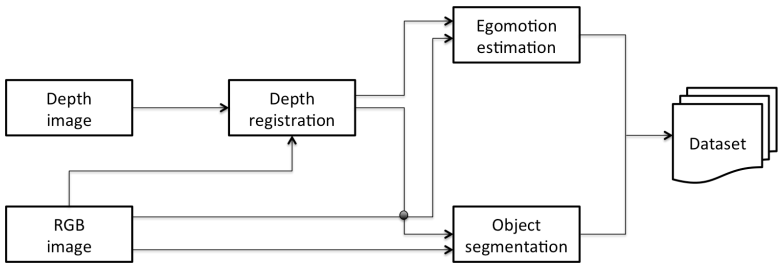


Figure 6.25: Flow diagram object dataset construction

Two object datasets are constructed to determine the object recognition performance. Each object is captured after which the orientation of that object is placed in a random pose in order to construct the query object. The first object dataset differs from the second query object dataset by change in lighting and orientation (figure 6.27). The elevation angle is kept equal across all objects for consistency. Keeping an equal elevation angle while walking around an object with a hand held camera appeared to be difficult, therefore a fixed elevation angle is chosen instead of multiple. Specifications of the dataset are stated below.

- Dataset contains household objects (figure 6.26)
- 36 objects
- 18 views per object, azimuth discretization of 20 degrees
- Fixed elevation angle between 40 and 50 degrees
- Captures 360 degrees
- Dataset captured twice by manual change of object orientation

Figure 6.26: Selection of household objects from the object dataset

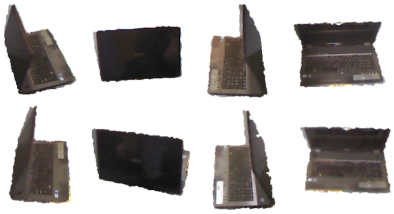
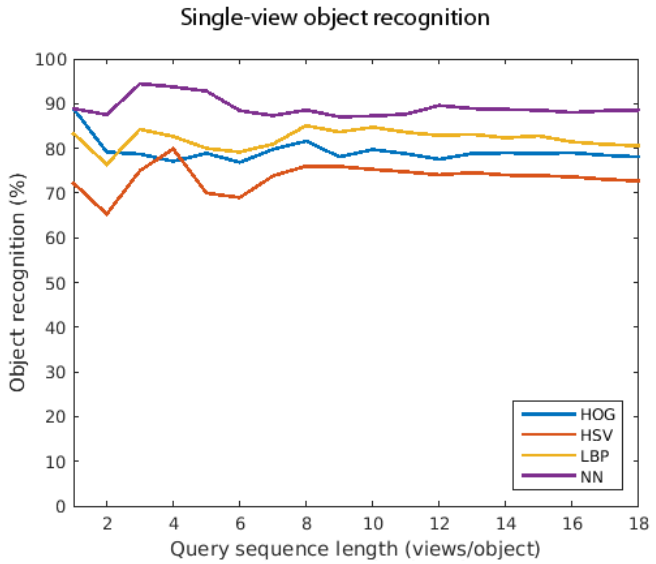


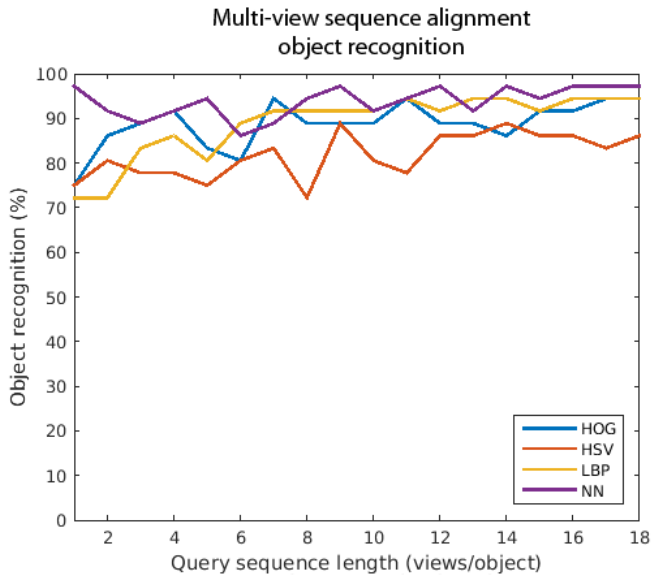
Figure 6.27: Unequal view distribution and lighting noise in a query object model after manual rotation

The object recognition performance is evaluated using the benchmark described in section 6.2.4. Figure 6.28 shows the result of single-view versus multi-view sequence alignment object recognition. The obtained results are similar to the object recognition performance found obtained on RGB-D dataset. Unequal view distributions in the (query) objects are caused by small errors in the estimated egomotion and view discretization. Changing the orientation of an object for the construction of a query object introduces lighting noise. Despite these noise factors an increase in object recognition is found for the sequence alignment method. The influence of view recognition on the results can't be investigated, because the orientation of objects are arbitrary changed for the construction of query object models.

The developed algorithm also has its limitations. In the sequence alignment algorithm each view is represented by a different branch in a Kd-tree. This might be a problem for objects with similar views from multiple viewpoints (i.e. balls, plates and bowls). Multiple branches may represent similar views, which has influence on the sequence alignment performance. Each view is represented by only one value based on the closest feature vector in a Kd-tree. The sequence alignment algorithm does not take the second or third closest feature vectors into account. Future work is proposed in the direction of using more nearest neighbours in the sequence alignment model and also pruning the Kd-tree resulting in clustering similar descriptors. A probabilistic classifier can be used to obtain certainty of recognition to stop further exploration of an object.



(a) Single view performance



(b) Multiview performance using sequence alignment

Figure 6.28: Object recognition in custom dataset

6.6. Conclusion

We have developed a novel multi-view object recognition model that incorporates the spatial relations between different viewpoints. This system is made generic to use any different feature descriptors based on different application scenarios. We have created a benchmark to evaluate performance of different features with and without this developed system. A Sequence Alignment algorithm has been used with vector quantized features from each view to achieve view point correlation in object recognition. The model is developed based on standard object datasets (RGB-D, SOIL-47) which provide viewpoint data for every object. Experiments show a 5 - 20% increase in recognition performance using the developed model. A fast Visual odometry estimation has been used to obtain viewpoint relations in an unsupervised manner and this has been incorporated with runtime object segmentation to provide a standalone system that can be used in real world scenarios. An evaluation of this system with a custom developed dataset also shows a similar increase in performance. The entire system is developed without using hyper parameters and hence is directly applicable with different feature vectors in any operating environment. The complete algorithm can perform recognition in $\sim$ 4 Frames Per Second on MATLAB code running on the robot's computer. We believe this is the first system to explicitly incorporate spatial viewpoint relations into object recognition and consider this an appropriate direction to improve the reliability of robot's object recognition in unconstrained environments.

References

[1] H.-P. Chiu, *Models for multi-view object class detection*, Ph.D. thesis, Massachusetts Institute of Technology (2009).

[2] S. Savarese and L. Fei-Fei, *3d generic object categorization, localization and pose estimation,* in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on* (IEEE, 2007) pp. 1–8.

[3] D. Zhou, J. Huang, and B. Schölkopf, *Learning with hypergraphs: Clustering, classification, and embedding,* in *Advances in neural information processing systems* (2006) pp. 1601–1608.

[4] J. Burianek, A. Ahmadyfard, and J. Kittler, *Soil-47, the surrey object image library,* Centre for Vision, Speech and Signal processing, Univerisity of Surrey.[Online]. Available: http://www. ee. surrey. ac. uk/Research/VSSP/demos/colour/soil47 (2000).

[5] K. Lai, L. Bo, X. Ren, and D. Fox, *A large-scale hierarchical multi-view rgb-d object dataset,* in *Robotics and Automation (ICRA), 2011 IEEE International Conference on* (IEEE, 2011) pp. 1817–1824.

[6] A. Vedaldi and B. Fulkerson, *VLFeat: An open and portable library of computer vision algorithms,* `http://www.vlfeat.org/` (2008).

[7] D. G. Lowe, *Object recognition from local scale-invariant features,* in *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, Vol. 2 (Ieee, 1999) pp. 1150–1157.

[8] N. Dalal and B. Triggs, *Histograms of oriented gradients for human detection,* in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, Vol. 1 (IEEE, 2005) pp. 886–893.

[9] T. Ojala, M. Pietikäinen, and T. Mäenpää, *Multiresolution gray-scale and rotation invariant texture classification with local binary patterns,* Pattern Analysis and Machine Intelligence, IEEE Transactions on **24**, 971 (2002).

[10] M. Rudinac, *Exploration and Learning for Cognitive Robots*, Ph.D. thesis, Delft University of Technology (2013).

[11] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, *CNN features off-the-shelf: an astounding baseline for recognition,* CoRR **abs/1403.6382** (2014).

[12] K. Simonyan and A. Zisserman, *Very deep convolutional networks for large-scale image recognition,* arXiv preprint arXiv:1409.1556 (2014).

[13] L. Stein, *Genome annotation: from sequence to biology,* Nature reviews genetics **2**, 493 (2001).

[14] T. F. Smith and M. S. Waterman, *Identification of common molecular subsequences,* Journal of molecular biology **147**, 195 (1981).

[15] G. S. Slater and E. Birney, *Automated generation of heuristics for biological sequence comparison,* BMC bioinformatics **6**, 31 (2005).

[16] C. B. Do, M. S. Mahabhashyam, M. Brudno, and S. Batzoglou, *Probcons: Probabilistic consistency-based multiple sequence alignment,* Genome research **15**, 330 (2005).

[17] J. Zhang, M. Marszałek, S. Lazebnik, and C. Schmid, *Local features and kernels for classification of texture and object categories: A comprehensive study,* International journal of computer vision **73**, 213 (2007).

[18] P. J. Besl and N. D. McKay, *Method for registration of 3-d shapes,* in *Robotics-DL tentative* (International Society for Optics and Photonics, 1992) pp. 586–606.

[19] Z. Fang and Y. Zhang, *Experimental evaluation of rgb-d visual odometry methods,* International Journal of Advanced Robotic Systems **12** (2015).

[20] A. S. Huang, A. Bachrach, P. Henry, M. Krainin, D. Maturana, D. Fox, and N. Roy, *Visual odometry and mapping for autonomous flight using an rgb-d camera,* in *International Symposium on Robotics Research (ISRR)*, Vol. 2 (2011).

[21] C. Kerl, J. Sturm, and D. Cremers, *Robust odometry estimation for rgb-d cameras,* in *Robotics and Automation (ICRA), 2013 IEEE International Conference on* (IEEE, 2013) pp. 3748–3754.

[22] T. J. J. Tang, W. L. D. Lui, and W. H. Li, *A lightweight approach to 6-dof plane-based egomotion estimation using inverse depth,* in *Australasian Conference on Robotics and Automation* (2011).

[23] H. Andreasson and T. Stoyanov, *Real time registration of rgb-d data using local visual features and 3d-ndt registration,* in *SPME Workshop at Int. Conf. on Robotics and Automation (ICRA)* (2012).

[24] I. Dryanovski, R. G. Valenti, and J. Xiao, *Fast visual odometry and mapping from rgb-d data,* in *Robotics and Automation (ICRA), 2013 IEEE International Conference on* (IEEE, 2013) pp. 2305–2310.

[25] W. L. D. Lui, T. J. J. Tang, T. Drummond, and W. H. Li, *Robust egomotion estimation using icp in inverse depth coordinates,* in *Robotics and Automation (ICRA), 2012 IEEE International Conference on* (IEEE, 2012) pp. 1671–1678.

[26] H. Bay, T. Tuytelaars, and L. Van Gool, *Surf: Speeded up robust features,* in *Computer vision–ECCV 2006* (Springer, 2006) pp. 404–417.

[27] M. A. Fischler and R. C. Bolles, *Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography,* Communications of the ACM **24**, 381 (1981).

[28] D. Holz, S. Holzer, R. B. Rusu, and S. Behnke, *Real-time plane segmentation using rgb-d cameras,* in *RoboCup 2011: robot soccer world cup XV* (Springer, 2011) pp. 306–317.

7

Novelty Detection for Online Action Recognition and Learning

This chapter focuses on online learning for action recognition in personal robots. Online learning is essential for personal robots to adapt in open environments and enable long-term human-robot interaction. This work presents methodology to detect unknown (novel) human action sequences, which is a crucial step towards incremental learning. We construct a new compact feature vector using the Kinect camera's skeleton sequence. This feature is used in a Hidden Markov Model (HMM)-based generative classifier, which has shown a good performance in standard action recognition tasks. Subsequently, novelty detection is approached from both a posterior likelihood and hypothesis testing view, which is unified as background models. Since novelty detection for action recognition has not been reported before, we investigated a diverse set of background models: sum over competing models, filler models, flat models, anti-models, and some weighted combinations. Our standard recognition system has an inter-subject recognition accuracy of 96% on the Microsoft Research Action 3D dataset. Moreover, the novelty detection module combining anti-models with flat models has 78% accuracy in novelty detection, while maintaining 78% standard recognition accuracy as well. Our methodology can increase robustness of any current HMM-based action recognition system against open environments.

Chapter modified from article:
Thomas Moerland, Aswin Chandarr, Maja Rudinac and Pieter Jonker: Knowing What You Don't Know - Novelty Detection for Action Recognition in Personal Robots, The 11th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, VISAPP 2016, Rome, Italy 2016

7.1. Introduction

Recognizing human actions is a very important aspect of robot perception. This becomes even more relevant for personal robots working together with humans in the near future. It is very difficult for the robot to learn all different human actions together and have robust recognition performance. Additionally, the subset of actions each robot will need to recognize differs based on the operating environment. Hence an online learning system is necessary, where the robot continuously extends its knowledge about various actions. This process is essential for long term autonomy of personal robots.

In many ways, an action recognition system can be paralleled with speech recognition, with key poses and its sequence similar to alphabets and words. Hence, we borrow motivation from the development of linguistic knowledge in children and translate certain concepts from speech processing into action recognition. It has been studied in psycholinguistics that bootstrapping allows for expansion of cognitive development starting around three years into child growth [1]. Indeed, one of the main components of human intelligence is our adaptivity: we can not only detect what we know, but also identify what we do not know. Moreover, humans use this new input to extend their knowledge, by closing the learning loop (Figure 7.1). In this context, bootstrapping involves equipping the robot with a basis structure and some starting knowledge, from which the robot can detect novel classes and subsequently learn them. Following [2], this entire learning process can be modularized into three steps (Figure 7.1):

(i) Anomaly detection (separation): separating videos belonging to known classes from those belonging to unknown classes.

(ii) Cohesion detection: identifying overlapping patterns among buffered anomalous videos identified in (i).

(iii) Retraining: efficiently retraining the ordinary classifier with the new action class, using the detected example videos from (ii).

In this chapter, we focus on the first step and investigate new methods for detecting unknown (anomalous) sequences for action recognition systems.

The recent advent of stable 3D imaging technology has strongly increased data quality in action recognition. A landmark paper for 3D action recognition from [3], introducing the Microsoft Research Action 3D dataset (MSRA 3D). The authors sample a frame-wise feature from the depth map and use a Hidden Markov Model (HMM) as back-end classifier. Their method has good average recognition accuracy on an inter-subject recognition task (92.9%), but is strongly view-dependent and computationally heavy.

More recently, Shotton et al. introduced the stable extraction of human skeletons from single depth images [4]. While the labeling of body parts had been an active research field for many years [5], the direct availability of skeletons raised much interest in the research community. Several papers studied view-invariant skeleton features [6]. Some examples are pairwise joint distances and joint motions [7] or joint angles and joint angle velocities [8]. An interesting approach

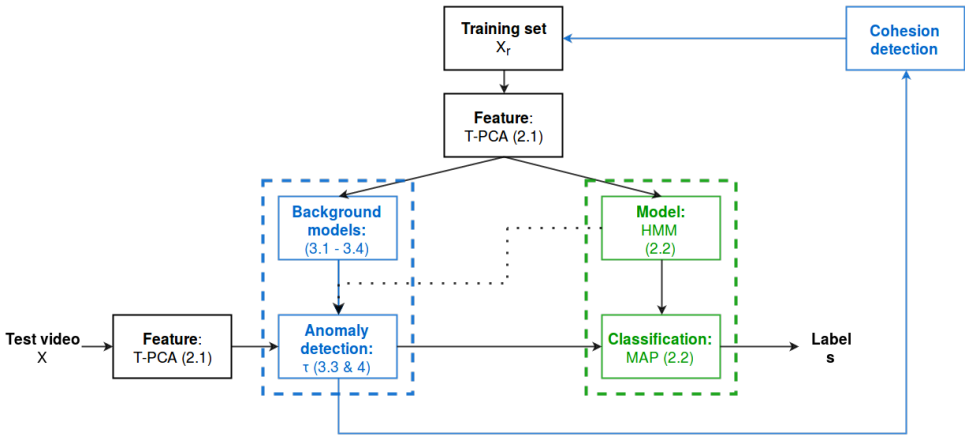


Figure 7.1: Overall system structure. Proposed novelty detection is shown in dotted blue, all standard action recognition components are shown in black and green.

combining skeleton and depth map information is called Space-Time Occupancy Patterns (STOP) [9]. The authors use the wireframe skeleton to reorientate the depth map to make the subject camera-facing. The feature is constructed from the depth-map occupation over a regular space-time grid, while the back-end classifier is based on a HMM again. Their method still holds the state-of-the-art result on the MSRA 3D inter-subject recognition task (97.5%).

Although the mentioned approaches have made important advancements, they exclusively study their methodology on a closed-set recognition task. Thereby, the system's performance is evaluated on action classes which were also available in the training set. None of the methods consider the occurrence of *unknown* action classes. This specific problem is studied in the machine learning field of *novelty detection*, which is for example reviewed in [10]. While a standard classifier assigns each new instance to the best-fitting class (which is by definition wrong if the video truly belongs to an unknown class), a novelty detection module first tries to identify such novel instances. In machine learning this problem is also known as anomaly detection.

To our knowledge, we are the first to report on the topic of anomaly detection for human action sequences. We present a unified approach called *background models* applicable to any generative classifier like a Hidden Markov Model (HMM)-based recognizer.

Our overall system structure is shown in Figure 7.1. Starting from the training set, we first construct a new compact frame-wise feature based on a Torso-PCA (T-PCA) framework. Then, we train a set of Hidden Markov Models (HMM) with shared keypostures. Each incoming test video is decoded under all class HMM's and assigned according to the maximum-a-posteriori (MAP) rule (green box in Figure 7.1). We extend this system with a novelty detection module (blue in Figure 7.1. First, we learn background models from the training set. These background models are combined with the normal HMM's to obtain a test statistic (*the back-*

ground corrected likelihood). In the anomaly detection part we determine a single optimal threshold τ on this test statistic. When the threshold is exceeded we proceed with standard classification (see Figure 7.3). Else, we identify the video as 'novel/unknown' and buffer it. A cohesion detection module can identify overlap among the buffer videos. When a human supervisor labels the unknown class, we can extend the training set with the new action and close the adaptive learning loop. This work focusses on the first step of novelty detection: anomaly detection through background models (blue dotted box in Figure 7.1).

The rest of this chapter is organized as follows. In the next section we introduce a compact and view-invariant representation of the human pose from the Kinect's skeletal joint information. The back-end classification through HMM's is formally described in section 7.2.2.

Then, we introduce two dominant views on anomaly detection from speech recognition: posterior probability and hypothesis-testing in sections 7.3.1 and 7.3.2. These approaches are subsequently unified as background models in section 7.3.3. Since this topic has not been studied before for action recognition, we investigate several background models: sum over competing models, filler models, flat models, anti-models and some reweighted combinations of them as described in section 7.3.4. The remaining sections of this work present the experimental setup and dataset (section 7.4), our results including both standard recognition accuracies and various novelty detection results (section 7.5) and a discussion of our results (section 7.6).

7.2. Action Recognition System

In order to investigate novelty detection, we first need a functioning standard recognition system. This consists of two modules; a feature vector which encodes the pose of a given frame into a compact representation, and a generative classifier which uses the encoded features to obtain the probabilities over the trained classes. Our proposed model uses a novel and compact feature vector based on the skeleton information (7.2.1). The back-end classifier is based on a set of Hidden Markov Models with shared keypostures (7.2.2).

7.2.1. Representation: Torso-PCA Framework

A good feature is ideally both compact and information-rich. Compact features are especially important for HMM-based back-end classifiers, since it is difficult for these generative probabilistic models to separate signal from possible feature noise. Earlier work on human perception of biological motion has shown that humans can recognize actions by looking only at movements of lights attached to the major joints [11], implying that tracking of human skeletal poses can provide sufficient information for action recognition.

The availability of real-time skeletal tracking from depth images introduced by [4] has advanced research in action recognition based on this information. The raw skeleton sequence contains the 3D locations of 20 joints at each frame. Many approaches in literature [12], [13] obtain a feature vector using all pairwise joint

distances, velocities and angles. For example, all pairwise joint distances result in a large feature vector (P=190), which not only contains redundant information, but also make the training process difficult due to the dimensionality. Many approaches in literature [7], [9] employ some dimension reduction technique (usually PCA) to reduce the feature vector length. However, PCA techniques might harm novelty detection, so we construct a novel and compact frame-wise feature, Torso-PCA (T-PCA) based on earlier work by [14].

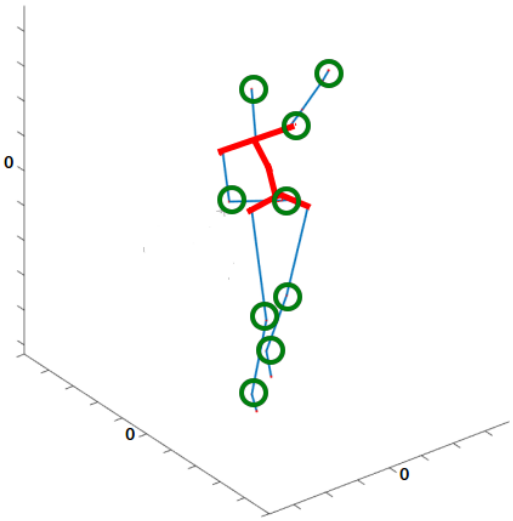


Figure 7.2: Skeleton-based features

Figure 7.2 illustrates the process of obtaining T-PCA features from every frame of video sequence. The raw skeleton sequence contains the 3D locations of 20 joints (in blue) for each frame (P=60) which are obtained following work of Shotton et al. [4]. We now construct a more compact frame-wise feature vector (P=30) from it. First, we translate the full skeleton to have its origin at the mean of the seven torso joints (marked by red lines). Subsequently, we apply PCA on the seven torso joint locations (which form a 7×3 matrix) to estimate a local coordinate frame with respect to the subject. The three principal axis correspond to the vertical, horizontal and frontal body axis, respectively.

The final feature vector is constructed from the 3D locations of the head, elbows, wrists, knees and ankles (marked by green circles) augmented with the three rotation angles (yaw, pitch, roll) related to the torso coordinate system. This is based on the assumption that body pose information relevant to actions is majorly encoded in the extremities (ignoring the noisy hand extraction), while the almost rigid body torso can be fully represented by its orientation in 3D space.

Reorientating the full skeleton to make the subject camera-facing has also been implemented. However, this reorientated feature slightly decreased our model performance. This can be explained from the large proportion of camera-facing sub-

jects in our dataset. We therefore choose to use the unrotated feature vectors. Finally, we will report results on the full length feature (P=30) and a PCA-reduced variant (P=10), comparing both when applicable.

7.2.2. Classification: HMM with Shared Key Postures

We use the sequences of compact feature vectors obtained from the skeletal data to perform action recognition using a generative model. Hidden Markov Models have shown their major success in speech recognition applications [15], and are now also frequently used as classifiers in action recognition [5]. These state-space models naturally handle variation in the speed of performed action. Furthermore, their probabilistic nature allows for novelty detection in low density area's, which will be further pursued in the next section.

We adopt a Hidden Markov Model system with shared key postures between action classes, earlier introduced as an Action Graph [16]. The pooled estimation of the emission model (associated with each key posture) increases model robustness, and furthermore ties the class models together.

The formal definition is as follows: We observe a set of videos x_r, $r = 1, 2, ..N$, with associated class label $s_r \in Q = \{q_1, q_2, q_3, .., q_m\}$, for m different action classes. Each x_r, of length t_r, has at timepoint t an observation vector x_{rt} of length P. Let $W = \{w_1, w_2, .., w_K\}$ denote a set of key postures. In the HMM we assume each feature vector x_{rt} has an associated hidden state variable $z_{rt} \in W$, and the transitions between subsequent hidden states follows a first-order Markov property. Thereby, the transitions between states can be represented as a $K \times K$ transition matrix, where each entry denotes the transition probability between states at subsequent timesteps, i.e. $A_{ij} = P(x_t = w_j | x_{t-1} = w_i)$. We assume K=50 for this work, which is close to the number reported for this dataset elsewhere [9]. Furthermore, the relation between the hidden nodes and observation vectors ($P(X|Z = w_k)$), i.e. the emission model, is modeled as a Gaussian with mean vector μ_k and diagonal covariance matrix Σ_k.

For each action class s we estimate a separate HMM. However, it is reasonable to assume the key postures and associated emission models are similar between actions. We therefore jointly estimate these parts of the HMM's over the different classes, effectively pooling their contributions. The action classes are discriminated by the class-specific transition matrix A_s. The full set of HMM's is thereby defined by the tuple $\Lambda = \{\mu, \Sigma, A\}$, where $A = \{A_1, A_2 .. A_m\}$.

Under these model assumptions we can write the full data log-likelihood as:

$$\mathcal{L}(X, Z, S|\Lambda) = \sum_{r=1}^{N} \left(\sum_{t=1}^{t_r} \ell(x_{rt}|z_{rt}, \mu, \Sigma) + \sum_{t=2}^{t_r} \ell(z_{rt}|z_{r(t-1)}, S, A) \right) \quad (7.1)$$

where ℓ denotes a log-probability, and X, S and Z denote the video, class and hidden state random variables, respectively. Since the hidden states Z are unob-

served, the model is estimated through the well-known Expectation-Maximization (EM) algorithm.

For comparison, we also include a clustered estimation approach. Here, we pool all frame-wise observations vectors in the training set and subsequently cluster these through k-means. Then, we consider each cluster as a key posture, estimating the observation model and transition matrices from their assigned feature vectors. Effectively, we now employ a 'hard' hidden node assignment, compared to the 'soft' hidden node assignment estimated in the EM algorithm.

To perform inference on an incoming video of the test set, we use the maximum-a-posterior (MAP) decision rule:

$$\begin{aligned}\hat{s} &= \underset{S \in Q}{\operatorname{argmax}} P(S|X) \\ &= \underset{S \in Q}{\operatorname{argmax}} \frac{P(X|S)P(S)}{P(X)}\end{aligned}$$

Ordinary speech recognition systems usually assume $P(S)$ is uniform (i.e. no prior on the action class) and ignore $P(X)$, since it does not depend on S. Thereby, classification effectively boils down to selecting the class with the highest raw probabilities, $P(X|S)$. These raw likelihoods are obtained through Viterbi decoding.

7.3. Novelty Detection

The introduced HMM system can only estimate the probability of the input video over the trained classes (Q). In this section we introduce novelty detection methodology that has been used in speech recognition and explain our proposed method for novel action detection.

Novelty detection for HMM's has been previously studied in speech recognition under the name of *Confidence Measures* (CM $\in [0,1]$) [17]. Confidence measures were introduced to post-evaluate the reliability of a recognition decision (as in Equation 7.2). Since a misrecognition might well be due to a currently unknown class, the goals of CM research and novelty detection are highly overlapping.

We will first introduce the two dominant streams in CM research: posterior probabilities (7.3.1) and hypothesis testing (7.3.2). Then we unify both approaches as *background models* (7.3.3). Since we are the first to investigate novelty detection for action recognition, we will investigate a diverse set of background model types (7.3.4).

7.3.1. Posterior Probability

A simple and direct way to detect novel classes is to threshold the raw probability $P(X|S)$, as used for assignment in the MAP rule (Equation 7.2). But this method does not provide a good measure of novelty as $P(X|S)$ is only a relative measure of fit. We do know which class is most likely, but we do not know how good the match really is. In contrary, an absolute and very intuitive measure of fit is the posterior probability of the class given the video: $P(S|X)$. In accordance to Equation 7.2, we

need the marginal probability of the video ($P(X)$) as a normalizing constant. This marginal video probability can be expressed as:

$$P(X) = \sum_{G} P(X|G)P(G) \tag{7.2}$$

where G denotes the full model space, including the models for all unknown classes. For example, the marginal probability could separate a novel class (high $P(X)$) from a noisy extraction (low $P(X)$). However, the distribution in the unseen model space is not known, and we will need methodology to approximate it. This will be elaborated shortly.

7.3.2. Hypothesis Testing

Another approach to confidence measures for HMM's was developed independently at AT&T Bell Labs [18] [19] [20]. Their work on *utterance verification* casts the problem as a statistical hypothesis test:

H_0: X_r is known and correctly recognized
H_1: X_r is novel and/or incorrectly recognized

A well-known choice, based on the Neyman-Pearson lemma, is to use the likelihood ratio test (LRT) statistic for testing:

$$LRT = \frac{P(X|H_0)}{P(X|H_1)} \geq \tau \tag{7.3}$$

As noted by [17], the major difficulty lies in modeling H_1, which is a very composite event with unknown data distribution.

7.3.3. Background Models

We propose both posterior probability and hypothesis testing approaches can be cast in the same framework as *background models*. Both $P(X)$ and $P(X|H_1)$ can be understood as the likelihood of the video under the (partially unobserved) background of the model space. On the log-scale, both methods technically reduce to subtracting the raw likelihood, $P(X|S)$, by a correction factor:

$$\begin{aligned} \ell^{corrected}(X|S) = & \quad \ell(X|S) - \ell(X) \\ = & \quad \ell(X|H_0) - \ell(X|H_1) \end{aligned} \tag{7.4}$$

As a confirmation of this similarity, both posterior probability and hypothesis testing approaches have independently developed 'filler' models, by [21] and [19] respectively.

The corrected posterior log-likelihood will be used as the test statistic for anomaly detection. We will identiy the video as 'novel' when the statistic is below a critical threshold τ, i.e. when:

$$\ell(X|S) - \ell(X|M) \leq \tau \tag{7.5}$$

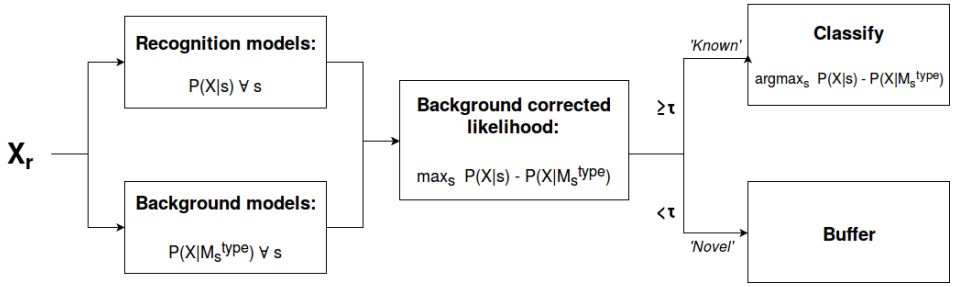


Figure 7.3: Flow diagram of background model correction and anomaly detection (i.e. expansion of the blue box in Figure 7.1). All probabilities denote their log-scale equivalents.

where M denotes the background model type. If this statistic is higher than τ, we continue with standard class assignment through the MAP decision rule (equation 7.2). The full test flow is depicted in Figure 7.3. A test video is decoded under standard class models and all background models. Then, the latter is subtracted from the former to give the background corrected posterior likelihood. The class with the highest posterior likelihood is considered for assignment. When the background corrected posterior likelihood exceeds a threshold τ, we proceed to standard classification through the MAP assignment rule (Equation 7.2). Else, we refrain from classifying and store the video in a buffer for future processing. Optimization of τ is illustrated in Figure 7.4. In the next section we introduce different types of background models. Estimation of τ is discussed in section 7.4.

7.3.4. Background Model Types

Since we are the first to study novelty detection for action recognition, we will investigate a diverse set of background models: sum of competing classes, filler models, flat models and anti-models. Filler and flat models are very generic, modelling the distant background of the model space. On the other hand, anti-models approach the closer surroundings of each class. Therefore we also investigate a reweighted combination of them, to combine their advantages.

Background models are themselves Hidden Markov Models, estimated on the same training set as the standard models. However, key postures and emission models obtained in the standard model estimation remain fixed now. All background models except the 'sum over competing hypothesis model' estimate a (class-specific) transition matrix: $\lambda_{type}^{(s)}$. All background models are estimated through EM.

We will denote the video's probability under the background model as $P(X|M_{type}^{(s)})$. The proposed background models are:

(i) Sum over competing hypothesis: This approach is related to the N-best list approaches in speech recognition, like for example in [22]. However, the number of action classes in action recognition is usually smaller, so we can sum over *all* known competing models:

$$\ell(X|M_{sum}) = \log\Big(\sum_{s} P(X|s)\Big) \tag{7.6}$$

(ii) Filler models: Filler models estimate one general transition matrix on all data, which is intended to approximate all humanly possible movements and possible background noise. Speech recognition variants can be found in [21] and [19]:

$$\ell(X|M_{filler}) = \log\left(P(X|\lambda_{filler})\right) \tag{7.7}$$

(iii) Flat models: Filler models are very generic models for the background, but they are still data dependent. We also include a uniformly initialized transition matrix with each entry equal to $\frac{1}{K}$:

$$\ell(X|M_{flat}) = \log\left(P(X|\lambda_{flat})\right) \tag{7.8}$$

(iv) Anti-models: As opposed to the previous background models, anti-models are class-specific. They are estimated on all videos *not* belonging to the specific class s [19]. Thereby, they are intended to approximate the surroundings of the classes true density area:

$$\ell(X|M^s_{anti}) = \log\left(P(X|\lambda^s_{anti})\right) \tag{7.9}$$

(v) Reweighted combinations: To combine the different strengths of the previous approaches, we also include the geometric mean of filler/flat models with anti-models:

$$\ell(X|M^s_{C1}) = \log\Big(0.5 \cdot \exp\left(P(X|\lambda_{filler})\right) + 0.5 \cdot \exp\left(P(X|\lambda^s_{anti})\right)\Big) \tag{7.10}$$

$$\ell(X|M^s_{C2}) = \log\Big(0.5 \cdot \exp\left(P(X|\lambda_{flat})\right) + 0.5 \cdot \exp\left(P(X|\lambda^s_{anti})\right)\Big) \tag{7.11}$$

We use the estimated background likelihoods $\ell(X|M)$ with the novelty detection statistic in Equation 7.5 to detect previously unknown action sequences as shown in the pipeline of Figure 7.3.

7.4. Dataset and Experimental Setup

We evaluate our proposed method over the publicly available 'Microsoft Research Action (MSRA) 3D' dataset. It contains segmented videos of 20 dynamic actions performed by 10 subjects for ideally 3 repetitions (N=557). Most literature follows the dataset's original paper [3], where tests are performed in subsets of 8 actions.

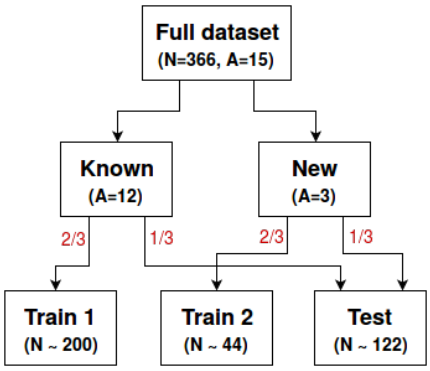


Figure 7.4: Novelty detection setup. (N = number of videos in a set, A = number of action classes)

Table 7.1: Standard recognition accuracy (no novelty detection) for clustered and EM estimated models.

Estimation method	**Recognition Accuracy**
Clustered	94%
EM	96%

Since we want to investigate novelty detection, we decide to pool 15 action classes together. These are: horizontal arm wave, hammer, high throw, draw circle, hand clap, two hand wave, side-boxing, bend, forward kick, side kick, jogging, tennis swing, tennis serve, golf swing, pickup & throw. We only retained videos with three repetitions per subject and action, and also removed some very noisy videos (N=366). For evaluation purposes we use 2/3 of the dataset for training (i.e two of the three videos per subject per action), which corresponds to 'Test 2' of the original paper. Standard recognition results are obtained over three epochs of a 3-fold cross-validation.

To evaluate novelty performance we will need a double dataset split, as depicted in Figure 7.4. For each run, 3 videos are randomly split off as 'novel'. Then, in a nested 3-fold cross-validation, HMM's and background models are trained on known videos (Train 1) and optimal threshold τ is determined on Train 1 and Train 2. Finally, novelty and recognition accuracy are evaluated on Test. Novelty results are obtained over two full epochs, which each consist of a 5-fold novelty split with nested 3-fold cross-validation (Figure 7.4).

With the introduction of novelty detection, we can also make errors at two levels. Apart from mistakes in the binary novelty module, we can also correctly identify a video as known, but still assign it to the wrong class. The latter is called a *putative* error. However, we are primarily interested in 1) recognition accuracy (percentage of known videos identified to the correct class, i.e. sensitivity) and 2) novelty accuracy (percentage of novel videos correctly identified as novel, i.e. specificity). Therefore, optimal τ will be determined from the largest sum of both accuracies, thereby ignoring the underlying error types.

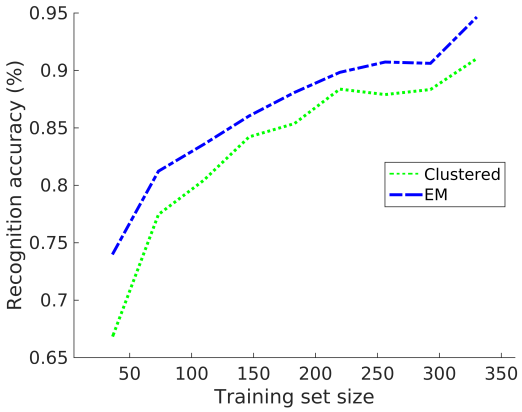


Figure 7.5: Learning rate: recognition accuracy as a function of the training set size for two estimation methods (EM and clustered)

7.5. Results

Recognition accuracy for the standard classification task in shown in table 7.1. Our novel compact feature (P=10) has accuracy close to the state-of-the-art results on this dataset, although we did use a slightly different test set-up (see section 7.4). The ability of our estimation methods to learn from smaller amounts of data is shown in Figure 7.5. The graph indicates EM estimation is able to learn from the data more quickly, although both methods eventually approach the same recognition accuracy. It is to be noted that the test results at 2/3 training set size are slightly lower compared to table 7.1. The setup for this plot is however not solely inter-subject, but makes a random split over the data. Although performance slightly decreases, these results also indicate our method generalizes well for larger training set sizes.

Table 7.2: Overview of recognition accuracy (sensitivity) and novelty accuracy (specificity).

	Model			
	Clustered		*EM*	
Background model	Sensitivity	Specificity	Sensitivity	Specificity
Raw (none)	0.72 (0.73)	0.60 (0.61)	0.71 (0.70)	0.57 (0.63)
Sum	0.64 (0.54)	0.77 (0.77)	0.66 (0.59)	0.74 (0.73)
Filler	0.73 (0.73)	0.66 (0.67)	0.73 (0.75)	0.58 (0.52)
Flat	0.68 (0.66)	0.77 (0.78)	0.71 (0.73)	0.74 (0.69)
Anti-model	0.77 (0.73)	0.77 (0.70)	0.73 (0.75)	0.69 (0.62)
Combination 1 (filler + anti)	0.76 (0.72)	0.75 (0.71)	0.73 (0.77)	0.68 (0.62)
Combination 2 (flat + anti)	**0.78** (0.75)	**0.78** (0.75)	0.73 (0.77)	0.73 (0.65)

Table 7.3: Optimal novelty detection results for the clustered standard model with flat&anti-model background (bold result in table 7.2). The results illustrate that we hardly make any putative errors (i.e. known videos assigned to the wrong class)

	True label	
Assigned label	**Known**	**Novel**
Known (correct)	78%	22%
Known (wrong)	1%	-
Novel	21%	78%

Novelty detection results are reported in table 7.2. It shows recognition accuracy (sensitivity) and novelty accuracy (specificity) for two estimation methods (clustered versus EM) and various background models. Each cell reports accuracy for the PCA-reduced feature vector (P=10) and between brackets the same result for the full-length feature vector (P=30). Optimal performance is obtained for the combined background model (flat + anti-model), with both accuracies at 78%. Results do not differ between the PCA-reduced (P=10) and full-length (P=30) feature vectors. The raw posterior likelihood is able to identify around 72% of the known videos in the correct class, while also detecting 60% of the novel videos. Obviously, recognition accuracy has decreased compared to table 7.1, since we augmented the problem by adding a set of videos from classes unavailable during training.

The different background models all improve performance, but in different ways. Optimal performance is achieved for the combined background of flat and anti-model, which reaches novelty and recognition accuracy of both 78%. Interestingly, the clustered estimation seems to outperform EM for novelty detection in general.

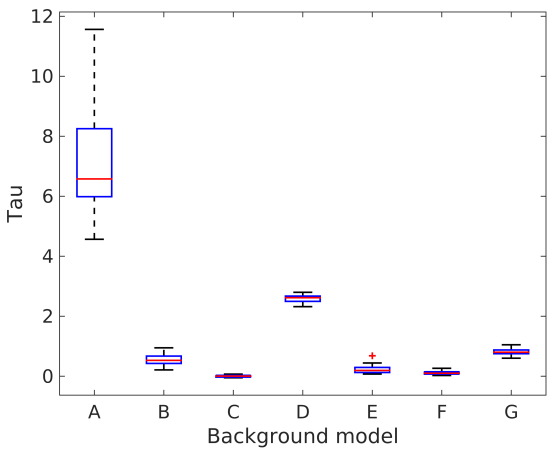


Figure 7.6: Consistency of τ.

Table 7.3 shows the underlying errors of our optimal novelty detection result. Although we optimized τ to maximize the sum of recognition accuracy and novelty accuracy, we can observe a clear difference in the type of errors our system makes.

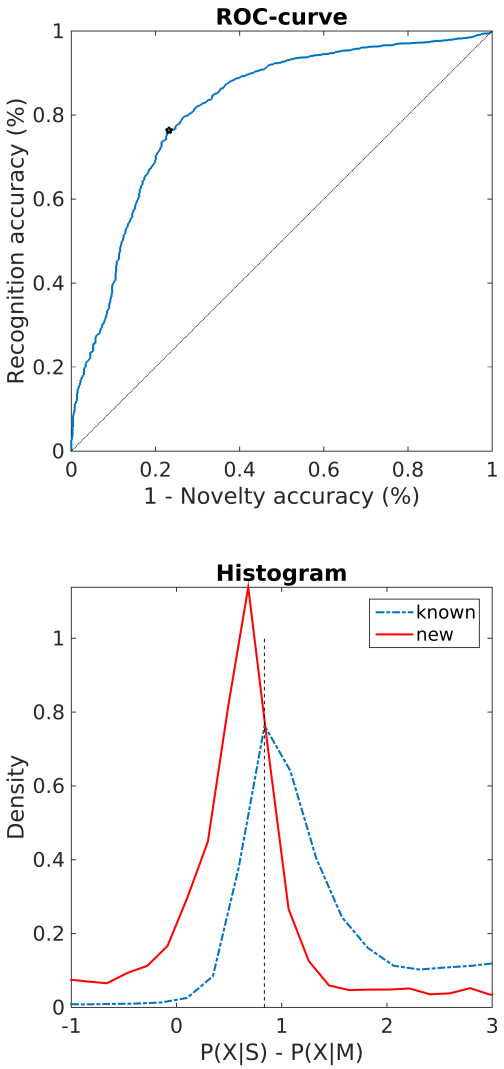


Figure 7.7: Scaling of τ. Top: ROC-curve showing recognition and novelty accuracy for various levels of τ. Optimal joint performance is marked with an asterix (the associated value of τ is visible in the bottom plot). Bottom: Distribution of background corrected likelihood for known (blue dashed line) and novel (red solid line) videos. Optimal τ is indicated by the vertical dashed line.

Putative errors, i.e. known videos assigned to the wrong class, occur only for 1% of the known videos. Thereby, the recognition accuracy considering only the known classes (i.e. a closed-set recognition problem) has actually increased compared to table 7.1. The decrease in recognition accuracy from about 95% (table 7.1 to 78% can be almost fully attributed to misrecognized novel videos. We could have expected this result, since the confidence measure methodology was developed to identify misrecognitions.

A closer inspection of the scaling of τ is provided in Figure 7.7. The model has been cluster estimated with combined (flat + anti-model) background correction. The ROC-curve (top) shows both accuracies for different values of τ. The bottom plot shows the distribution of the background corrected likelihood for known and new classes. As expected, the distribution under the known class has a higher mean posterior likelihood. We see an overlapping area, corresponding to the 22% errors on both sides.

Finally, we investigate the ability of τ to generalize over different novelty settings. Figure 7.6 shows the distribution of the optimal τ for all background models over various dataset splits (according to Figure 7.4). Boxplots show the distribution of τ over multiple splits for the seven background models (A-G) as reported in the rows of table 7.2, respectively. Results are obtained over 2 full epochs, 30 runs, on the clustered model with P=10. On each run, we determine a single value of τ. We see the raw likelihood without background correction (model A) has difficulty generalizing over different splits, i.e. the variance of the optimal τ is large. On the other hand, all background methods clearly improve the consistency of τ, indicating the background models do systematically correct the different scaling of the raw likelihood.

7.6. Discussion and Conclusion

To our knowledge, our work is the first to report on novelty detection for action recognition. Our methodology can assign 78% of the known videos to the correct class, while also identifying 78% of the unknown videos as novel. Furthermore, our background model methodology shows consistent results over various dataset splits, indicating the method should generalize well to different settings. Background models can be easily implemented on any HMM-based action recognition system, providing it with robustness against open-set environments.

An interesting aspect of our results is the relatively good performance of the flat background model. This model was the only data-independent background approach. Most literature on novelty detection tries to re-use the dataset in some smart way. The good results of the flat model touch upon a fundamental challenge in novelty detection: 'you can not model the unknown (new classes) from the known (data)'.

We think the results in table 7.3 could give a motivation to use our methodology even for closed-set environments. As we mentioned before, the background model system can also be used as a confidence measure to identify misrecognitions. Considering only the closed-set problem (i.e. first column of table 7.3), we would refuse to classify 21% of the videos, but for the assigned videos we can be very certain

that the class is correct.

The decrease in recognition accuracy from 96% for closed-set recognition to 78% for open-set recognition nicely illustrates the inevitable trade-off in novelty detection. As table 7.3 clearly shows, the bottleneck of this decrease is in the novelty detection module. By including a set of unknown videos, we strongly increased the difficulty of the classification task. However, real-life is by definition an open-set, and any system (like a personal robot) ignoring this problem will see their good closed-set recognition performance strongly decrease in practical application.

In conclusion, we identify three purposes for our anomaly detection methodology based on background models: 1) increased accuracy in *closed-set* recognition tasks by acting as a confidence measure, 2) increased robustness against *open-set* problems by filtering of unknown videos and 3) as a first step towards *adaptive* learning by closing the learning loop of Figure 7.1. Due to the large resemblance of human intelligence, novelty detection can significantly extend both robotic functionality and human-robot interaction. We intend to implement the complete online learning system on our personal robot [23] and tackle the challenges posed by unconstrained motions and environments.

References

[1] S. Pinker, *Language Learnability and Language Development* (Cambridge, MA: Harvard University Press, 1984).

[2] M. M. Masud, Q. Chen, L. Khan, C. C. Aggarwal, J. Gao, J. Han, A. N. Srivastava, and N. C. Oza, *Classification and adaptive novel class detection of feature-evolving data streams.* IEEE Trans. Knowl. Data Eng. **25**, 1484 (2013).

[3] W. Li, Z. Zhang, and Z. Liu, *Action recognition based on a bag of 3D points,* in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on* (2010) pp. 9–14.

[4] J. Shotton, T. Sharp, A. Kipman, A. Fitzgibbon, M. Finocchio, A. Blake, M. Cook, and R. Moore, *Real-time human pose recognition in parts from single depth images,* Communications of the ACM **56**, 116 (2013).

[5] D. Weinland, R. Ronfard, and E. Boyer, *A survey of vision-based methods for action representation, segmentation and recognition,* Computer Vision and Image Understanding **115**, 224 (2011).

[6] J. Aggarwal and L. Xia, *Human activity recognition from 3D data: A review,* Pattern Recognition Letters **48**, 70 (2014).

[7] X. Yang and Y. Tian, *Effective 3D action recognition using eigenjoints,* Journal of Visual Communication and Image Representation **25**, 2 (2014).

[8] S. Nowozin and J. Shotton, *Action points: A representation for low-latency online human action recognition,* Microsoft Research Cambridge, Tech. Rep. MSR-TR-2012-68 (2012).

[9] A. W. Vieira, E. R. Nascimento, G. L. Oliveira, Z. Liu, and M. F. Campos, *STOP: Space-time occupancy patterns for 3D action recognition from depth map sequences,* in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications* (Springer, 2012) pp. 252–259.

[10] M. Markou and S. Singh, *Novelty detection: a review–part 1: statistical approaches,* Signal Processing **83**, 2481 (2003).

[11] G. Johansson, *Visual perception of biological motion and a model for its analysis,* Perception & psychophysics **14**, 201 (1973).

[12] G. Yu, Z. Liu, and J. Yuan, *Discriminative orderlet mining for real-time recognition of human-object interaction,* in *Asian Conference on Computer Vision* (2014).

[13] J. Wang, Z. Liu, Y. Wu, and J. Yuan, *Learning actionlet ensemble for 3D human action recognition,* Pattern Analysis and Machine Intelligence, IEEE Transactions on **36**, 914 (2014).

[14] M. Raptis, D. Kirovski, and H. Hoppe, *Real-time classification of dance gestures from skeleton animation,* in *Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '11 (ACM, New York, NY, USA, 2011) pp. 147–156.

[15] M. Gales and S. Young, *The application of hidden Markov models in speech recognition,* Foundations and trends in signal processing **1**, 195 (2008).

[16] W. Li, Z. Zhang, and Z. Liu, *Expandable data-driven graphical modeling of human actions based on salient postures,* Circuits and Systems for Video Technology, IEEE Transactions on **18**, 1499 (2008).

[17] H. Jiang, *Confidence measures for speech recognition: A survey,* Speech communication **45**, 455 (2005).

[18] R. Sukkar, C.-H. Lee, *et al.*, *Vocabulary independent discriminative utterance verification for nonkeyword rejection in subword based speech recognition,* Speech and Audio Processing, IEEE Transactions on **4**, 420 (1996).

[19] M. G. Rahim, C.-H. Lee, and B.-H. Juang, *Discriminative utterance verification for connected digits recognition,* Speech and Audio Processing, IEEE Transactions on **5**, 266 (1997).

[20] R. C. Rose, B.-H. Juang, and C.-H. Lee, *A training procedure for verifying string hypotheses in continuous speech recognition.* in *ICASSP* (IEEE Computer Society, 1995) pp. 281–284.

[21] S. O. Kamppari and T. J. Hazen, *Word and phone level acoustic confidence scoring.* in *ICASSP* (IEEE, 2000) pp. 1799–1802.

[22] T. Kemp and T. Schaaf, *Estimating confidence using word lattices.* in *EUROSPEECH*, edited by G. Kokkinakis, N. Fakotakis, and E. Dermatas (ISCA, 1997).

[23] A. Chandarr, M. Bruinink, F. Gaisser, M. Rudinac, and P. Jonker, *Towards bringing service robots to households: Robby ,Lea smart affordable interactive robots,* in *IEEE/RSJ International Conference on Advanced Robotics (ICAR 2013)* (2013).

8

Conclusion

8.1. Research Goal

The research interest and applicability of robotics have diversified and seen a tremendous growth in recent years. There has been a shift from industrial robots operating in constrained settings to consumer robots working in dynamic environments associated closely with everyday human activities. This has been fuelled by advancements in various engineering domains, especially computing power density and battery capacity which have made commercial robots with higher autonomy very feasible. The development of efficient probabilistic algorithms to handle uncertainties in sensing and control are enhancing the autonomous capabilities, enabling them to assist humans in tackling various challenges in future. Among many other applications, personal service robots to assist elderly, compliant robots with advanced perception skills for flexible manufacturing and autonomous driving vehicles for safe transportation are quite relevant. In all these cases, robots have to work in close cooperation with human users and an intuitive higher level interaction between robots and layman users is essential for its widespread acceptability. Hence development of cognitive and perceptual skills in humans is studied where a tight link between perception, action and interaction is observed. This thesis focuses on developing robots perceptual skills, especially based on visual information from a user interaction point of view. A domestic service robot is taken as a use case to develop such algorithms. A physical robot (LEA) is developed from scratch considering the affordability and user acceptability. A generic human centric architecture for highly autonomous and interactive robots is proposed to integrate various capabilities of a robot that are triggered by user interaction. A specific case of object recognition is investigated as many tasks faced by such robots involve perception and manipulation of different household objects. Various algorithms are developed to enhance the reliability of object perception overcoming challenges posed by dynamic environments and affordable hardware by incorporating various modalities of information available to a robot. The development of algorithms in

this direction is significant as the concepts can be readily extended to incorporate user and environment recognition to complete the perceptual capabilities of robots.

8.2. Bottom up development of a service robot: LEA

In Chapter 2, we have developed robot LEA which can perform various tasks of a domestic service robot. Keeping in mind affordability and appearance for user acceptability, the entire robot has been developed from ground up in our laboratory. The mechanical construction of the robot has 4 parts, a Non-holonomic mobile base, torso, arm and head. The mobility of the robot is achieved using a differentially driven base which also houses the batteries which also provide mechanical balance. The Torso is constructed over the base which contains the arm and the head. A 4DoF arm is built using lightweight materials, which in turn decrease the torque required from the motors. This is achieved by a smart mechanical design, which places the heavier motors for shoulder, elbow and wrist tilt inside the torso and coupled to joints through a series of belt drives. An underactuated 3 fingered gripper is attached to the end of arm link to grasp rigid and non rigid objects of various shape and sizes without requiring explicit vision based grasp planning. A compact pan-tilt neck has been developed to provide required torque and speed for the head motion, while still maintaining a compact form factor. An aesthetically pleasing robot head which also houses various sensors has been designed and 3D printed with a glossy appearance.

LEA is also equipped with a minimalist sensor suit that enables it to perceive the world around it. Visual perception is handled by an RGB-D camera (Kinect) in combination with a high definition RGB camera which are disguised as mouth and nose of the robot. The robot is equipped with a highly sensitive directional microphone and an amplifier driven set of speakers to enable voice interaction with human users even in noisy environments. The speakers are disguised as eyes of the robot and can also be manually adjusted to represent different emotional states. The robot has a planar laser scanner and an array of ultrasonic sensors to provide safe navigation in cluttered environments. Geared DC motors are used for base, arm and neck joints. Every motor is equipped with an incremental encoder which is used for feedback control of the joints done by custom developed two axis motor controllers, 3Mxl. There are a total of 5 3Mxls which are daisy chained together and controlled by a single high speed USB connection. The robot has RGB leds to indicate its internal states and also a touch display for the user to easily change the robots behaviour. All the sensors and actuators are integrated and controlled through a central consumer grade computer running Linux operating system. The entire mechanics and electronics are concealed within a specially designed black *velvet dress* and the robot head. Care has been taken not to fall in the *"Uncanny valley"* of social acceptance of human like robots. The successfulness of this design has been seen by public reactions at various demonstrations and interest from print and visual media.

The entire robot control software has been developed based on the Robot Oper-

ating System (ROS) as a set of individual nodes connected over an internal network. The nodes are organized into a modular 4 layer software architecture which comprises of core, subcore, functionality modules and low level drivers. Core is the central state machine which has a sequence of steps to be performed for different kind of tasks that user requests. This commands the subcores, which are set of small tasks that can be reused in different applications. The subcore initiates various functionality modules to complete the task it has been requested. LEA is equipped with various functionality modules which include online face recognition and learning, object recognition and grasping, autonomous navigation, user tracking and following, voice based natural user interaction and learning actions from user demonstration. The sensor data is aggregated and actuators are controlled by the functionality modules through low level drivers.

Various modules for object perception and learning are developed and evaluated in real world scenario with this service robot.

8.3. Joint visual attention

An efficient and intuitive non-verbal interaction between user and robot for conveying objects of interest is the first step in human centric object perception. When an user and a robot are looking at a common visual scene, understanding which part of the scene the user is interested constitutes *Spatial grounding*. Though this looks quite straightforward in a context of two humans interacting, there exist challenges due to varying positioning between user and robot, object properties, illumination and cluttered backgrounds. These have been tackled in Chapter 3 by combining bottom up segmentation using RGB and Depth information with a top down segmentation involving different different modalities based on pointing and eye-gaze of the user

Bottom up segmentation involves focussing attention on specific regions in a visual scene by separating background information from regions of interest in the foreground. This is performed only based on structural and chromatic data from different parts of a visual scene. A biologically inspired saliency model of Itti which estimates the saliency by considering the distinctive local features based on colour, intensity and orientation is used to obtain most significant regions in the RGB image of a visual scene. In a geometric representation of a visual scene, largely planar regions generally constitute background (walls) and support surfaces (tables and floor) in general man-made environments. Dominant planes are estimated using Depth data to identify and remove these regions as backgrounds. An *organized multiplane estimation* algorithm is used to detect these planar regions by clustering of surface normals in point clouds. A background mask is created in the 2D image plane by projecting these 3d points using intrinsic camera parameters.

Top down segmentation involves segmenting a visual scene with respect to a specific requirement, in this case the intentions of the user. A novel pointing based saliency map is generated based on detecting the users arm and finger direction. The angle between the vector from a 3D point to the user's pointing finger and the vector from the user's hand to the finger is estimated for every valid point in the depth based saliency mask. A normal distribution is fitted over this angle to

obtain a pointing based segmentation mask. Though this mask in combination with an RGB based saliency mask provides an accurate localization of the user's region of interest, it can work only when the user is in front of the robot. In order to obtain this localization when the robot and user are standing next to each other, a novel gaze based segmentation is developed. The robot uses an eye tracker worn by the user to detect regions of interest the user is looking at. A graph cut based algorithm is used to segment the region around the gaze point in user's visual frame and this is translated to the robot's visual frame using a combination of back-propagation and feature matching. When the user is looking at an object with high texture, SURF features are detected and matched between the two visual frames to estimate the gaze map. But when the object of interest has a plain appearance, a back-projection based on HSV histogram is used. These two modes are switched automatically based on the number of detected keypoints.

The interest maps obtained from all these different modalities are combined together using a Hadamard product based on spatial configuration between the user and the robot which influences if a gaze map and/or a pointing map can be estimated. Extensive experiments are performed in cases where the robot and user are next to each other, opposite to each other and in between, with textured and plain objects. We obtained a good spatial grounding between robot and user even in challenging conditions with poor illumination and cluttered background.

Having obtained the user's region of interest in the visual scene, a semantic object recognition system has been developed in Chapter 4

8.4. Multimodal semantic object recognition

In line with the proposed human centric robot architecture, the spatial grounding from Chapter 3 leads into semantic grounding and perceptual inference modules. A three layered semantic recognition framework that can incorporate multiple modalities of information has been developed taking inspiration from cognitive psychological studies. Modalities of color, shape and location have been considered in this research. The first layer consists of semantic grounding modules that abstract raw sensory information into a probability distribution over meaningful semantic concepts familiar to humans. A second layer operates on these semantic features to obtain an object hypothesis based on every individual semantic modality. The last layer performs knowledge association to estimate combined probability over known objects and final inference is obtained.

Color semantic features are obtained by converting an image into a histogram over 11 linguistic color names that are shared among various languages. A SLIC super-pixeling method is performed over the raw RGB image to decrease noise and reduce number of pixels to be color labelled. After this, each super-pixel is assigned a semantic color name based on a mapping learned from a large number of online images using a PLSA model . A normalized histogram over these 11 colors provides a scale invariant semantic feature vector.

Shape information is quantified from object point clouds using Ensemble of Shape Features (ESF). These are global descriptors which describe spatial relations between points in a point cloud for a given view. Shape semantic features

that abstract this ESF features into a distribution of basic geometric shapes (cubic, spherical, cylindrical) are obtained by training ESF features of representative objects from datasets such as 3DNet. Novel shape categories are incrementally learned by thresholding the obtained probability distribution over known shapes and retraining the shape semantic classifier.

The location of the robot inside a household gives some powerful cues regarding the object classes that it can encounter in that particular location. The location of the robot is determined based on its position the metric map it uses for its navigation. The different rooms are demarcated in the map and the robots current metric location from its navigation system is looked up in the demarcations to obtain the semantic location. During the training phase, the information about the most commonly present locations for each object is acquired. Such information is also available in the benchmark for personal robots such as RoboCup@Home and RocKiN competitions.

Apart from using this to perform recognition, characteristics of novel objects can be communicated to human users based on these semantic features. Object hypotheses are obtained based on these modalities separately by a supervised training of random forest based specific classifiers. With this a probability of an object given the modality ($P(obj|modality)$) is obtained for every mode of information available. All these individual hypotheses are combined together using a weighted fusion to obtain the overall object probability. These weights are based on the certainty of a particular hypothesis quantified using entropy of the distribution. The object inference is obtained as a Maximum Likelihood Estimate of the final probability. The inherent advantage of this method is its ability to handle a missing modality where the weights automatically become zero. This also leads to selection of dominant features of an object.

Extensive experimentation over a custom created dataset with objects present in different locations with changing illumination and orientations shows a competitive performance of this multi modal semantic recognition system. Similarly colored or shaped objects give rise to ambiguity in final recognition and hence a tracking system to explore the object is developed in Chapter 5 and used for multi-view recognition system developed in Chapter 6.

8.5. Tracking system for exploration and interaction

A method to track contours of objects and humans when explored from various relative motions is developed in Chapter 5. Apart from enabling visual servoing based grasping of objects, it allows the learning of shape and color distributions of an entire object.

The cause of target drifting in many current trackers stems from the presence of background regions in the initial target selection and these regions expand with time. Hence a target initialization is performed where points inside the initial window are clustered and the dominant cluster is considered as the target to be tracked. A *track by detection* algorithm is used to continuously track the initial target and it

starts with a search space refinement. The metric consistency in physical size of objects is exploited to decrease the search window to twice the current object size in 3D around its current location. This is projected to a 2D bounding box using intrinsic camera parameters. Additionally the targets encountered by the robot are present on planar support surfaces in many conditions. Hence all planes perpendicular to the objects principal planar component are identified and removed from the search space. This refinement provides an increase in speed as well as immunity to drifting into similar regions in the background.

A target appearance model is created using color and depth modalities. With this, dense maps which represent the probability of every pixel belonging to object or background are obtained for different modalities. A color based map is obtained by using semantic color features described in Chapter 4. Samples from positive and negative target regions are obtained and a discriminative model separating these regions on an 11 dimensional manifold is learned using a fast second order online learning classifier, AROW (Adaptive Regularization Of Weights). The classifier outputs a confidence score at every part of the image to be foreground and these are normalized to obtain the color target probability. A depth model is constructed based on the spatial compactness of a target object. Given the initial distance of the object from the camera, an object is modelled as a single dimensional Gaussian with initial mean and covariance of target depth. While the covariance is kept constant throughout the tracking process, the mean of this distribution is constantly updated. This provides intrinsic robustness to occlusion by assigning higher costs to any regions that can come in between target and camera.

The obtained foreground probabilities from color and depth are combined to detect the target at consecutive frames using a graph cut formulation where the search space is considered as a Markov random field. The smoothness cost for this graph cut is assigned based on physical (Euclidean) distances between 4 connected neighbours and the data cost is obtained by an optimal fusion of color and depth probabilities which use a target-background dissimilarity measure. Color similarity is obtained using the Bhattacharyya distance between histograms of an 168 dimensional HSV feature from target and background. The depth similarity is obtained by the same distance over histograms of depth quantized into 10cm bins. A min-cut algorithm performs final segmentation of target from the background using the obtained costs.

The tracking performance is quantified with a target boundary overlap criterion and is evaluated over motions involving scale, orientation and viewpoint changes with rigid and non-rigid objects showing good performance at a speed sufficient to perform run-time on a robot.

8.6. Viewpoint correlated multi-view object recognition

With the ability to detect users' objects of interest (Chapter 3) and track it with different relative motions between object and robot (Chapter 5), the reliability of recognition of these objects in challenging domestic environments is enhanced us-

ing visual appearances from multiple views while incorporating the spatial relations between these viewpoints as well. Concluding from an extensive literature search that methods that explicitly use viewpoint correlation between visual appearances do not exist, a novel algorithm using *Sequence Alignment* techniques from Bio-Informatics is developed. This method is designed to be generic to incorporate any different feature descriptors, avoiding hyper-parameter tuning and operating at run-time conditions on a robot.

Firstly, a benchmark is developed to compare feature descriptors describing different properties of a visual appearance such as color, shape, texture, edges without considering viewpoint correlation inorder to evaluate the performance improvements of the developed algorithm. To decrease influence of external factors, two standard datasets, RGB-D and SOIL-47 are used in this benchmark as they have sequentially captured object images with respective viewpoints. A simple nearest neighbour classifier based on Euclidean distance is used to assign class labels to image views. The matching speed is improved by use of a KdTree approach. The classification system has been designed to be parameter free to avoid specific tuning for different conditions. Also experiments to identify the influence of the viewing angle between consecutive images for view recognition are performed to obtain the view-discretization angle to be used in the Sequence alignment system.

A sequence of views obtained while moving a camera around an object is modelled as a string of characters in a specific order. This is achieved by a Vector Quantization algorithm using the KdTree trained in the benchmark for single view recognition. This process abstracts a multi-dimensional feature vector into a single dimensional element. A view-discretization is performed which generates a new character in a string with every 20° of motion around an object. The generated string during training is used to create an object database to which smaller substrings obtained in the testing phase are matched. The matching is performed with a Sequence Alignment principle, where a dynamic programming based algorithm identifies the best matching sequences. The sum of Euclidean distances between features in the testing subsequence and matched sequence are calculated and the object with the lowest score is assigned as class of the test sequence. Experiments on datasets with varying sequence length show the improvement achieved by this algorithm over single view based systems.

An unsupervised estimation of spatial relationship between different viewpoints is obtained using a Visual Odometry (VO) system. A fast VO system based on feature tracking and a linearized motion model is implemented which estimates local motions as a transformation matrix with an iterative algorithm that minimizes the re-projection error between matched point-pairs from successive frames. Experiments show the ability to estimate relative motions with an accuracy sufficient to match the viewpoint data from the RGB-D dataset, without requiring a turn table. This is integrated with a object segmentation system based on fast dominant plane detection and clustering points above this plane to obtain target objects. A camera alignment with ground plane initializes angles for orientation estimation from VO. This integration provides a standalone system that can perform view-correlated multiview object recognition. A custom dataset with challenging illumination and

orientation conditions is created to evaluate the performance of the developed algorithms in real world scenarios. Experiments on this data reflect the improvements shown in experiments based on reference datasets. The integrated algorithm operates at run-time on the robot and enhances the reliability of its object recognition.

8.7. Action recognition and learning affordances

Chapters 4, 5, 6 have focussed on improving object recognition in detail. Their emphasis lies mainly on the use of visual information with color and depth images and their optimal fusion for improving the reliability of recognition. In Chapter 4 a generic framework is introduced which can integrate different modalities of information to recognize objects. An example of using location information from robots navigation system for this purpose was implemented. The use of other modalities such as affordances of objects to augment recognition has been discussed in Chapter 1. One mode of affordances is to identify the utility of a certain object like "Apple is for eating, Soap is used for cleaning, A pen for writing" etc. These affordances are elicited by the kind of actions a human user performs with objects. Hence, recognizing and learning of new actions of the user is essential to use these affordances for object recognition and interaction.

In Chapter 7, a Hidden Markov Model (HMM) based action recognition system using skeleton tracking from 3D data is used to recognize user actions. An important aspect of action recognition is the ability to detect previously unknown actions which can then be clustered and learned from user interaction. Novelty detection techniques from the audio processing domain are applied to action sequences to detect previously unknown action sequences.

A compact feature vector representing essential information from skeletal structure is developed to assist the HMM based generative classifier to separate a signal from possible feature noise. A Torso centred coordinate system with respect to the user is obtained by performing a PCA over the seven torso joints of the skeleton with three principal directions of (x, y, z). The center of this coordinate system is obtained as the mean of all the torso joints. The entire skeleton is transformed to this coordinate system and roll, pitch, yaw of the extremities are concatenated to obtain a representation invariant to scale and orientation change between robot and user. This is based on the assumption that the body pose information relevant to actions is majorly encoded in the extremities

Sequences of these compact feature vectors obtained from skeletal data are used to perform action recognition using a HMM based generative model whose probabilistic nature allows for novelty detection. An action graph which is a HMM system with shared key postures between action classes is adopted. With the pooled estimation of the emission model, robustness to speed changes in action sequences is increased apart from tying the classes together in model space. Different actions are discriminated based on the transition matrix between their hidden states and they are obtained using an EM algorithm on training data. The inference on incoming action streams is obtained by a Maximum-a-Posterior (MAP) rule which results in a class with highest value in its raw likelihood which is obtained through Viterbi decoding. Evaluating this algorithm on Microsoft Research Action

(MSRA) dataset, obtains a recognition performance of 96 % which is comparable to the state of the art.

Confidence measures which quantify the uncertainty in predictions are used to perform novelty detection. A combination of *posterior probability* and *hypothesis testing* algorithms unified as background models is used for this purpose. The absolute posterior probability of the HMM inference is obtained from its raw likelihood using the marginal probability of the video as a normalizing constant. Hypothesis testing is performed by thresholding the Likelihood ratio (LRT) statistic. Both of these methods involve subtraction of the raw likelihood of a video with the likelihood of the video under the (partially unobserved) background of the model space. Different methods to estimate this background model which include *sum over competing hypothesis*, *filler models*, *anti-models*, *flat models* and their reweighted combinations are compared based on sensitivity and specifity metrics, which provide a 78% accuracy in estimating novel classes, while maintaining the same accuracy in assigning a known video to the correct class. This shows the advantages obtained by this algorithm.

This thesis shows a good progression of perceptual capabilities of service robots in relation to their user intractability. In the same time, it also opens a wide area of research for the future. There is the challenging issue to mechanical design of the robot to increase the object manipulation workspace to reach till the floor in order to complete its structural capability of a domestic service robot. This has to be achieved without affecting the visual appearance of the robot. Force sensing can be employed in various joints to enhance bi-directional physical interaction with users. Spatial grounding in Chapter 3 can be improved from working in static scenes to more realistic dynamic operating environments. This can be achieved by adding more cues such as hand motion and face tracking to robustly deal with distracted users, background motions etc. Also the integration of visual cues with natural language understanding can effect common grounding of objects even without the user being in physical vicinity of the robot. Semantic recognition of objects in Chapter 4 can be improved by adding more cues such as affordances learned from actions, linguistic interaction with the user etc. Additionally, closed loop online object learning with user feedback can allow customization of a robots perceptual abilities to different user requirements. Making use of advancements in deep learning technologies with the integration of language based Recurrent Neural Networks (RNN) with image based Convolutional Neural Networks (CNN) trained to extract semantic features is a promising direction. Expanding the semantic recognition model to human user recognition is essential for robots dealing with multiple users. The tracking of objects in Chapter 5 can be improved by integrating a 6DoF motion model to allow handling of rapid linear and rotational motions. The algorithm can be extended to simultaneously track multiple objects, which in combination with human skeletal tracking can be used to obtain affordances for object recognition in addition to directly learning actions based on observing human behaviour. Learning objects by correlating visual appearances with spatial relations in Chapter 6 can be expanded to include changes in scale and simultaneous motions in *yaw, pitch and roll* angles of the camera. Integration of proprioception data of joint motions, with

the visual estimation of relative motion in a Kalman framework can enhance the reliability of spatial relation estimation in challenging visual conditions with untextured regions and drastic illumination changes with shadows, etc. Detection of novel actions in Chapter 7 can be improved by user feedback and can be integrated into a complete online action learning system by development of clustering, interaction and retraining modules. An abstraction from action to object affordances can lead to a complete semantic object recognition model in Chapter 4.

This thesis has focussed on human centric object recognition capabilities of a robot. A tighter integration of the developed algorithms with cognitive and action capabilities of a robot as proposed in the human centric robot architecture is required to complete a fully functional service robot, providing reliable usability to its end user.

Acknowledgements

A PhD is not just an academic process, its a rugged journey that helps you evolve and mature in life. Focussing on a very small domain over 4 years goes through its ups and downs and its not possible without scientific, financial, social, moral support that comes from various people. A PhD is incomplete without expressing gratitude and acknowledgements to all the people that made it possible for this to happen.

Firstly, i would like to thank all the esteemed members of my graduation committee who spent their valuable time and efforts reviewing and refining my thesis.

I would like to express my heartfelt gratitude to my Promotor, Prof. Pieter Jonker. Thank you for trusting me and giving me a lot of freedom and autonomy during the course of my research. It gave me courage to explore various directions without hesitation and helped grow as an independent researcher. Teaching students and public speaking did not come to me naturally. I matured in the art of lecturing with your constant encouragement and feedback, that now i would cherish teaching anytime. I look up to you for your detailed knowledge on a wide variety of fields and been engrossed in your explanations of different aspects of human system through an engineering perspective. Your openness and welcoming nature makes many students comfortable and brings the best out of them and i hope to emulate you in due course.

I could not have finished my PhD without the never ending support of my co-promotor Maja Rudinac. Thank you Maja for giving me confidence that i can be a successful researcher and making me trust in myself. I cannot be here without your patient guidance to develop my academic writing and structuring my thought process. I learnt the characteristics of a good supervisor from you and this has proved to be quite valuable to me. You have been a very good friend, family and an inspiration both within and outside academics. I admire your resilience during tough phases of life, your never exhausting energy and your innate nature to care for your people more than yourself. I hope to imbibe these and looking forward to working with you again in RCS, tackling bigger challenges.

I would like to thank Prof. Jenny Dankelman and Dr. John van den Dobbelsteen for initiating my PhD with DORA project. Though i could not integrate the results of the project into my thesis, the core ideas of need of collaborative robots and bridging perception of autonomous agents with humans were developed from DORA, while interacting with surgeons and designing image processing solutions to enhance the surgical workflow. Also, thank you Arjan van Dijke for the support in DORA.

I express my sincere gratitude to Prof. Martijn Wisse for initiating the Robocup @ Home project under which robots Robby and Lea were developed, which laid the foundation for my Masters and Ph.D thesis. The lessons learnt from participating in

Robocups in Mexico and Eindhoven have helped immensely to identify and work on challenges of developing robust robots that can operate outside industrial settings.

Thank you Machiel and Floris for being amazing friends and colleagues over the last 5 years. Your motivation and commitment made the long days and nights spent assembling and programming robots, an enjoyable process. I really enjoy the ease at which we can work together and I believe it was an important factor for all the progress we have achieved in the past few years and for the future. It has also been a wonderful experience travelling with you guys and Maja to Mexico, Spain, Iceland, Argentina, Japan etc. Those are among my most cherished memories.

I would like to thank all the members of the Delft Biorobotics laboratory who created a wonderful and productive working atmosphere. There is no real robot without its hardware. Especially, thank you Guus and Jan for investing your valuable time and experience to help us make the robot into reality. Guus, thank you very much for your untiring efforts for rapidly making all the electronic components and helping us whenever we called you even at odd times. I would like to sincerely thank Wouter Caarls who helped me in improving my software and linux skills and for being patient all the stupid doubts I had initially. Thank you my close colleagues Boris, Mukunda, Susana, Tim, Xin for all the interesting discussions we have had and making the last 4 years fly away in no time.

I would like to thank deeply all the Master students i have worked with. Jeff, Koen, Joris, Thomas, Leon, Wouter, Aashish most of you who later became my colleagues, I appreciate dedication and effort you had put in your thesis. I enjoyed many discussions we have had, the new ideas you always had. I also thank all the students who i have had a chance to work with, Akshay, Mirian, Lu Xia, Bart, members of Minor robotica.

I would like to express my heartfelt gratitude for all the members of the BME secretariat. Dineke and Mirjam thank you for your time and efforts in administration. Thank you Nancy, Sabrina, Hanneke, Anouk, Diones for your smiling welcome everytime and your quick assistance, whenever i come to you.

I would like to thank Martin Roos and Lerovis for creating RCS, which gives a platform to develop and bring robots outside the laboratory to assist many people in real life. It gives inspiration and meaning, especially in times of critical self introspection. I would also like to thank all my friends at RCS, Wouter, Toby, Mitsos, Charlotte, Aashish, Fabian, Loes, Luiz, Gabriel, Eelko and others for the good times working together. All of you have made my transition from academia to industry very smooth. Looking forward to the future with all you guys.

I would like to thank my good friend Karthik for the latent support he has been throughout this journey. Though far apart, we shared many common ideas, ideals and the adventures we planned. I would also thank Sergio and Nishant who were together with me from beginning of my days in Delft and shared a house and all the good and not so good things in life. I really miss the nights we spent around Nishant's table in Bagijnhof looking at the stars (when we could) and discussing about the world and life as self made philosophers. I would also like to thank my friends and fellow delftlings Daniel, Martina, Renata, Minos, Vasilis, Arvind, Joeri, Lindsay, Valia, Chris for the happy faces and the lovely dinners and parties we

have had. I would also like to thank my pals who made adjusting to life in Delft a pleasure, Johann, Raghu, Malli, Advait, Praveen, Sriram, Ananth, Prasanth. I would also like to thank Kamal who was an inspiration from high school times and when life brought us closer again in Europe and the good times in our travels. I also thank all the people at Phoenix Delft where i could get much needed refreshment from focussed table tennis, especially thanks Quan for teaching me new tricks.

I express my sincere regards and respects to my Professors at NIT-Trichy, India, especially Prof. Sundareswaran and Dr. Sankaranarayanan for creating a strong engineering fundamentals and inspiring towards pursuing higher studies.

Its beyond my vocabulary to express my gratitude for Revathi, my partner for life. She has been around me since my first day in Delft and i cannot imagine life in Netherlands without her. Rev, you are a bundle of unbridled enthusiasm and excitement and i cherish every moment spent with you. You have been a big factor in my transformation from a naive little boy into mature and responsible person and made me question, wonder and smile about life. You have always motivated me to giving more than the best of my abilities and kept pushing me for delivering a high quality thesis. I feel complete with you and I am confident that, together we can sail through the sea of this life with so much thrill and satisfaction. I would also like to thank my mother and father-in-laws and Rashmi for their concern, constant support and encouragement. I am very grateful to have earned your good wishes and a new family.

Last but nevertheless least, I would like to thank my parents in my mother tongue.

என்னை பெற்று வளர்த்த தாய்க்கும் தந்தைக்கும் என்னுடைய இதயம் கனிந்த நன்றி. நீங்கள் இல்லாமல் நான் இன்று இங்கு நிர்க்க முடியாது. சிறு வயதில் இருந்து நான் எல்லாவற்றிலும் சிறந்து விளங்கவேண்டும் என்று தன்னலமின்றி உங்கள் வாழ்வின் நோக்கமாக கொண்டீர்கள். அதற்காக நான் என்றும் உங்களுக்கு நன்றிக்கடன் பட்டு-ள்ளேன். பல நாடுகள் கடந்து இருந்தாலும் உங்கள் நினைவு இல்லாமல் ஒரு நாளும் சென்றது இல்லை. ஏன்மேல் எல்லையற்ற பாசம் கொண்ட அமீயா, உங்களை என்றும் நான் மிகவும் நேசிக்கிறேன். உங்களை போல் ஒருவர் கிடைக்க நான் பெரிய அதிர்ஷ்டசாலி. சில வருடங்களாக உங்களை பிரிந்து இருக்க நான் மிகவும் வருந்துகிறேன். மதுரையில் உங்களுடனும் அய்யாவுடனும் வளர்ந்த நாட்கள் என்னால் மறக்க முடியாது. இந்த புத்தகம் உங்கள் அனைவருக்கும் என்னுடைய அர்ப்பணிப்பு. என் வாழ்வின் எல்லா முயற்சிகளிலும் உங்கள் ஆசீர்வாதம் வேண்டும்.

என் வாழ்வில் தெளிவு குடுத்து, என்னுடைய முழு திறனையும் வெளி-ப்படுத்த ஆற்றல் தரும் சத்குருவுக்கு என்னுடைய நன்றி.

I am eternally grateful to the invisible universal energy that makes this physical existence a reality.

Aswin Chandarr A.B
Den Haag, September 2016

Curriculum Vitæ

Aswin Chandarr was born in Karur, India on 24-08-1988. He finished his high school in 2006 at Jawahar Higher Secondary School, Neyveli and the same year started his studies at National Institute of Technology - Trichy, India, specializing in Electrical and Electronics engineering. During his studies he was attracted to robotics and gained exposure with his internships in National University of Singapore and DFKI, Bremen. He was also active in participating in many competitions and organizing workshops to teach robotics.

He graduated with Bachelor of Technology in 2010 and consequently started Masters Degree at Delft University of Technology. He performed an internship at Renesas electronics, Paris in 2011 where he worked on Realtime time optical flow estimation using SIMD processors. He worked on Delft personal robotics project during his graduation where he developed object recognition and manipulation algorithms for robot Robby and was a part of Robocup@Home competition in Mexico in 2012.

He started his PhD under Pieter Jonker at TU-Delft after graduating with Masters in Mechanical Engineering, specializing in Biomechanical design in 2012. He initially worked on improving workflow in surgical procedures in hospitals using vision algorithms as a part of DORA project. He also worked on second generation of Delft personal robot LEA and took part in the Robocup@Home competition in 2013 in Eindhoven. Learning from the challenges of robot perception in real world scenarios, he directed his research towards developing robust algorithms for learning and recognizing objects in a human centric manner.

In 2014, he cofounded Robot Care Systems(RCS) along with some other members of the robocup team, which focuses on developing affordable and interactive robots that assist elderly people. Since September 2016 he is working as a Senior Robotics Developer at RCS.