Doctoral Dissertation

Probabilistic Techniques in Semantic Mapping for Mobile Robotics

José Raúl Ruiz Sarmiento
2016

Tesis doctoral
Ingeniería Mecatrónica
Dpt. de Ingeniería de Sistemas y Automática
Universidad de Málaga

UNIVERSIDAD DE MÁLAGA
DEPARTAMENTO DE
INGENIERÍA DE SISTEMAS Y AUTOMÁTICA

El Dr. D. Javier González Jiménez y el Dr. D. Cipriano Galindo Andrades, directores de la tesis titulada "Probabilistic Techniques in Semantic Mapping for Mobile Robotics" realizada por D. José Raúl Ruiz Sarmiento, certifican su idoneidad para la obtención del título de Doctor en Ingeniería Mecatrónica.

Málaga, 3 de octubre de 2016

_____

Dr. D. Javier González Jiménez

_____

Dr. D. Cipriano Galindo Andrades

Dept. of System Engineering and Automation
University of Málaga
Studies in Mechatronics

Probabilistic Techniques in Semantic Mapping for Mobile Robotics

AUTHOR: José Raúl Ruiz Sarmiento

SUPERVISORS: Javier González Jiménez
Cipriano Galindo Andrades

Thesis defended on 25th November 2016

JURY:
Antonio Jesús Bandera Rubio (Málaga University, Spain)
Luís Filipe de Seabra Lopes (Aveiro University, Portugal)
José Luis Blanco Claraco (Almería University, Spain)

To my family,
in heaven and on earth.

A mi familia,
en el cielo y en la tierra.

Table of Contents

Abstract

Semantic maps are world representations that permit a robot to understand not only the spatial aspects of its workspace, but also the meaning of the existing elements (objects, rooms, etc.) and how humans interact with them (*e.g.* functionalities, events, and relations). To achieve this, a semantic map enhances purely spatial representations, like geometric or topological maps, with meta-information concerning the types of elements and relations to be found in the working environment. This meta-information, called *semantic* or *common-sense* knowledge, is typically codified into *Knowledge Bases* (KBs).

An example of a piece of semantic knowledge stored in a KB could be: "refrigerators are big, box-shaped objects normally located in kitchens, which contain pill boxes and perishable food". Encoding and managing this semantic knowledge enables the robot to reason about the information gathered from a given workspace, as well as to infer new one in order to efficiently accomplish high-level tasks like "hey robot! take the pills to grandma, please".

This thesis contributes the usage of probabilistic techniques to build and maintain semantic maps, providing three main advantages in comparison with traditional approaches:

i) to handle uncertainty (coming from inaccurate robot sensors and models),

ii) to provide coherent environment interpretations by exploiting contextual relations among the observed elements (*e.g.* fridges are usually in kitchens) in a holistic fashion, and

iii) to yield certainty values that reflect the correctness in the robot understanding of its surroundings.

Specifically, the included contributions can be grouped into two major topics. The first set of contributions focuses on the *scene object and/or room recognition*

problems, since semantic mapping systems must reckon on reliable recognition algorithms for building proper representations. For that, we explore the utilization of *Probabilistic Graphical Models* (PGMs) for exploiting contextual relations among objects and/or rooms dealing with uncertainty, and the utilization of KBs to enhance their performance in different ways, e.g. detecting incoherent results, providing prior information, reducing the complexity of the probabilistic inference, generating synthetic training samples, enabling the learning from experience, etc.

The second group of contributions accommodates the probabilistic outcome of the developed recognition algorithms into a novel semantic map representation, coined *Multiversal Semantic Map* (*MvSmap*). This map manages multiple interpretations of the robot workspace, called *universes*, which are annotated with the probability of being the true ones according to the current knowledge of the robot. Thus, this approach gives a grounded belief about the understanding of the environment, which enables a more coherent and efficient robotic operation.

The proposed probabilistic algorithms have been thoroughly tested against other cutting-edge approaches employing state-of-the-art datasets. Additionally, this thesis also contributes: two datasets, *UMA-Offices* and *Robot@Home*, containing diverse ground truth information and sensory data from different types of devices covering office and home environments, and two software tools, the *Undirected Probabilistic Graphical Models in C++* (UPGMpp) library, and the *Object Labeling Toolkit* (OLT), for working with PGMs and processing datasets respectively.

Acknowledgments

Luckily, it is large the list of people who have been around and helped me, in one way or another, to reach the peak of this sharp mountain called *doctorate*. They were there in both good and not that good moments, and all of them deserve a warm mention. Nevertheless, the space for showing my gratitude is limited, so I will do my best!.

Foremost, I would like to express my special appreciation and thanks to my supervisors Prof. Dr. Javier González Jiménez and Dr. Cipriano Galindo Andrades. Our brainstorming meetings, source of a bunch of ideas, and their constant support, indications, and positive thoughts are responsible to a great extent of this work. I have to say that I consider them the parents of my academic career. Javier is an example of tireless effort, patience, and talent. He is an absolute passionate about his work, and successfully leads the MAPIR group, at the university of Málaga, which is in constant growing. Cipriano is the inspiration personified, always with affectionate words and acts, and fresh ideas to climb right to the top of the mountain. Thank you for believing in me.

To be part of the MAPIR group is an experience itself. All of us, as a team, celebrate the victories and regret the frustrations of others. It is difficult to imagine a more humane, and at the same time skilled group of people, within and outside the workspace. Starting with the *B team*, I have to mention Mariano Tarifa, Francisco Meléndez, Javier G. Monroy, Rubén Gómez, Manuel López, Carlos Sánchez, Jesús Briales, Andrés Góngora and Ángel Martínez, and the former members Eduardo Fernández, Ana Gago, Emil Khatib, Miguel Algaba and Gregorio Navidad. Thank you guys for being that amazing. I would also like to thank the senior researches at MAPIR, whose words helped me to stay motivated and focused. So Juan A. Fernández, Ana Cruz, Vicente Arévalo, and Francisco Moreno, thank you for that. I cannot forget Jose L. Blanco, a former member of the group, totally in love with research and with a shared passion for music, who gave me practical indications.

During my PhD years I have been in several conferences and schools. In there, I have met great people that were a plus within the study-develop-publish cycle. I also completed a stay at the University of Osnabrück, under the supervision of Prof. Dr. Joachim Hertzberg, a brilliant and close person, where I shared office with my colleague and friend Martin Günther, and I met nice people like Sebastian Stock, Jochen Sprickerhof, Sven Albrecht, Thomas Wiemann, Kai Lingemann, and Astrid Heinze. Thank you all for that enriching adventure, unfortunately my level of German is still low, I promise to improve it!. A special thanks to Bárbara Rotstein, my Spanish girl in Osnabrück, my stay there would had been quite different without her.

Friends have also played a pivotal role during the development of this thesis, specially Cristian F. Segura, Ismael Gutiérrez, José D. Pérez, José D. Sarmiento (*compadre*), Jesús Ramírez, Francisco Jiménez, Francisco A. Nieto, and Laura R., as well as their respective and lovely partners. They forgave my absence from many meetings, and illuminated me with their brilliant careers and growth as people. You fellows are awesome.

My relatives have been like parts of my body, I could not conceive this period of time without them. My mother María Sarmiento, was my heart, and my father José Ruiz, was my mind. My strong brother Juan L. Ruiz, his pretty wife Mónica Gallardo, and my lovely nephew Juan A. Ruiz were my skeleton. My uncles M. Josefa Sarmiento, Inmaculada Sarmiento, Toñi Sarmiento, and Antonio Díaz, and my cousin Samuel D. Díaz were my muscles. I just put the soul, and we all together achieved this goal. I do not forget my grandparents, specially José Sarmiento. I am sure that, wherever you are, you are reading these lines. It does not matter that you did not speak English, the language of love is universal. Heartfelt thanks.

Last but not the least, I would like to thank my *neni*, Rocío, and her family, now also mine, for sharing with me the last two years of this project. You enjoyed my victories like yours, and spent weekends with me at home working in front of a computer screen, patiently waiting for having some leisure time. The effort has now its rewards, and I promise to return all your support and love back, but multiplied by two.

José Raúl Ruiz Sarmiento
Málaga
September 2016

This thesis was partially supported by the Spanish grant program FPU-MICINN 2010, and by the research projects *PROMOVE: Advances in mobile robotics for promoting independent life of elders* (DPI2014-55826-R) and *IRO: Improvement of the sensorial and autonomous capability of Robots through Olfaction* (2012-TEP-530), funded by the the *Spanish Government* and the *Andalucía Regional Government*, respectively.

Resumen

Introducción

El invierno se acerca. Un robot sirviente detecta que la temperatura está disminuyendo y decide llevarle una manta a una adorable abuela. En el mismo edificio, otro robot encargado de patrullar una planta de oficinas se alerta al detectar una luz encendida en una habitación; rápidamente se percata de que es el compañero del área de investigación, Bob, trabajando hasta tarde por tercera noche en esta semana. Mientras tanto, su hija Alice está triste por la ausencia de sus padres, y su colega robótico, apodado cariñosamente *Roboto*, busca su oso de peluche favorito. Sophie, la madre de Alice, también está contando las horas para verla, y ordena a un robot limpiar las mesas una vez que su restaurante ha cerrado al público.

Estos escenarios son ejemplos donde los robots móviles de hoy en día, en mayor o menor medida, pueden proveer una serie de servicios para mejorar el nivel de vida de la sociedad. Cada vez se vislumbra más claramente que los robots están llegando para quedarse, como se ve en su exitosa aplicación a diversas tareas como vigilancia, cuidado de la salud, compañía, entretenimiento, mantenimiento del hogar, etcétera [97], donde colaboran con humanos o los reemplazan en tediosos o peligrosos quehaceres. Algo común a todas las aplicaciones anteriores es la necesidad de construir representaciones del entorno de trabajo, comúnmente llamadas *mapas*, las cuales permiten a un robot móvil alcanzar un cierto grado de consciencia respecto a sus alrededores para poder, por ejemplo, navegar evitando obstáculos, localizarse a si mismo con respecto a un sistema de referencia dado, almacenar información relevante sobre los elementos a su alrededor, etc.

Las representaciones tradicionales del entorno de trabajo del robot, como es el caso de mapas geométricos [23, 140], topológicos [110, 109], o híbridos [139, 13], aún son intensivamente usadas gracias a las habilidades básicas con las que dotan

al robot (navegación y localización). A pesar de ello, la ejecución de tareas de alto nivel como las mencionadas en los escenarios anteriores requiere representaciones más sofisticadas, cercanas al modo en el que los humanos interpretan su entorno. Los mapas semánticos (*semantic maps* en inglés) aparecieron para cubrir esta necesidad, permitiendo a un robot no sólo *comprender* los aspectos espaciales de su entorno, sino además el significado de sus elementos (objetos y habitaciones) y cómo los humanos interactúan con ellos, por ejemplo funcionalidades, eventos, y relaciones. Para ello se considera meta-información, comúnmente conocida como *Conocimiento Semántico* (*Semantic Knowledge* o *SK* en inglés[1]), sobre los tipos de elementos que se pueden encontrar en el área de trabajo del robot, incluyendo sus relaciones. Esbozos de dicha información, típicamente codificada en una *base de conocimiento* (*Knowledge Base* o *KB* en inglés), pueden ser: las mantas se encuentran habitualmente almacenadas en armarios; las luces de la oficina deben estar apagadas tras la jornada laboral; los osos de peluche mejoran el estado de ánimo; la vajilla frágil debe lavarse en el lavavajillas.

Motivación

Típicamente, los mapas semánticos son poblados[2] con información exacta, por ejemplo un objeto es una manta o no lo es. Esto se debe a la incapacidad de las representaciones semánticas tradicionales para tratar con resultados inciertos, lo que fuerza la utilización de algoritmos de reconocimiento que provean información exacta, habitualmente mediante la aplicación de umbrales a resultados probabilísticos. Por ejemplo, un algoritmo de reconocimiento[3] indicando que un objeto puede ser una manta con una probabilidad de 0.52, y una alfombra con 0.48, podría proveer un único resultado considerando el objeto como una manta y desechando la otra hipótesis, aunque esta es también altamente probable. Este enfoque exacto claramente compromete la operación del robot: la incertidumbre, proveniente de fuentes como el propio sistema de percepción del robot o los modelos empleados para tratar el problema, se ignora al almacenar los resultados de reconocimiento en el mapa semántico. De este modo, aunque los resultados del ejemplo claramente muestran que el reconocimiento es ambiguo, nuestra querida abuela podría terminar con una áspera alfombra encima suya. Este es un escenario de entre los muchos posibles que ponen de manifiesto la necesidad de utilizar técnicas capaces de proveer mediciones de incertidumbre sobre sus resultados para poblar y mantener mapas semánticos – para lo cual la literatura recurre comúnmente a técnicas probabilísticas [141, 65] –, así como de adaptar las representaciones semánticas actuales para poder manejar información incierta. Esto resultaría en una operación más coherente y eficiente por parte del robot móvil.

[1] Cuando sea posible, a lo largo de este resumen se utilizarán los acrónimos en inglés de las herramientas utilizadas, por ser su uso más común en la comunidad científica.

[2] *Poblar* un mapa semántico se refiere al proceso de introducción de los elementos espaciales en el entorno del robot en dicho mapa, comúnmente objetos y habitaciones, percibidos mediante su sistema sensorial.

[3] Para simplificar la explicación se considera que existen sólo dos tipos de objetos, mantas y alfombras.

Tratando de evidenciar aún más la conveniencia de trabajar con información incierta, supongamos un escenario donde a un robot sirviente, recién aterrizado en su nueva casa desde el laboratorio, se le encomienda el traer las zapatillas a la abuela adorable. En ausencia de información espacial, el robot puede inferir (de acuerdo con la información cargada en su *KB*) que la localización más probable de las zapatillas es un dormitorio. Durante el mapeo inicial de la casa por parte del robot, este reconoció un dormitorio correspondiente a la habitación más lejana con respecto a la posición actual de la abuela con una probabilidad de 0.45, y 0.43 de ser una cocina[4]. Otra habitación cercana a la posición del robot ha sido reconocida como cocina con una probabilidad de 0.48, y como dormitorio con 0.47. La utilización de la interpretación más probable, el *modus operandi* usual cuando se trabaja con mapas semánticos tradicionales, daría lugar a la exploración de la habitación más lejana, con un 45% de probabilidades de ser el lugar correcto, mientras que el considerar ambas interpretaciones produciría un plan más lógico: echar primero un vistazo a la habitación más cercana.

Aunque existen numerosos algoritmos para el reconocimiento de objetos y/o habitaciones que proveen mediciones de incertidumbre sobre sus resultados, estos usualmente trabajan mediante el procesamiento individual de cada elemento espacial de acuerdo con sus características geométricas (forma, tamaño, orientación, etc.) o de apariencia (color, textura, brillo, etc.). En otras palabras, si el tipo más probable para un objeto es *manta*, este es considerado una manta sin tener en cuenta que otros objetos hay a su alrededor ni su localización. Este enfoque ignora la rica información contextual presente en los entornos humanos: la distribución de las habitaciones sigue un cierto orden, y los objetos no están colocados aleatoriamente, sino siguiendo una cierta configuración acorde a su funcionalidad (por ejemplo, un mando a distancia suele estar en el entorno de una televisión, un pasillo conecta habitaciones, o una bañera suele encontrarse en el cuarto de baño) [113, 73, 117]. El modelado y aprovechamiento de esta información contextual puede ser útil, por ejemplo, para clarificar resultados inciertos: siguiendo con el ejemplo anterior, si el objeto se encuentra en un armario, este pertenecerá más probablemente al tipo *manta* que al tipo *alfombra*, el cual se encuentra usualmente sobre el suelo. Este tipo de información puede codificarse de manera natural en las bases de conocimiento, no obstante, su explotación para el reconocimiento contextual de objetos/habitaciones manejando incertidumbre no es simple.

Los *Modelos Gráficos Probabilísticos* (*Probabilistic Graphical Models* o *PGMs* en inglés) [65] son una herramienta ampliamente utilizada para el modelado y la explotación de relaciones de contexto tratando con incertidumbre. Estos modelos trabajan con una representación en forma de grafo, donde los nodos representan variables aleatorias y los arcos conectan variables que tienen algún tipo de relación. Por ejemplo, en el caso del reconocimiento de objetos, cada objeto en la escena es representado como una variable aleatoria que toma valores de entre los tipos de objetos posibles

[4]Nótese que la suma de ambas probabilidades es de 0.88. El resto, hasta sumar 1, se corresponde con las probabilidades de pertenecer a otro tipo de habitación, *e.g.* pasillo, cuarto de baño, salón, etc.

(mesa, sofá, libro, etc.), mientras que los arcos conectan variables cuyos objetos asociados están situados cerca en la escena. Esta representación soporta la ejecución de algoritmos de inferencia probabilística, los cuales son capaces de proveer los resultados de reconocimiento deseados, junto con mediciones de incertidumbre sobre dichos resultados. Los *PGMs* han sido aplicados con éxito a tareas como eliminación de ruido en imágenes, procesamiento de lenguaje natural, reconocimiento de la actividad en una escena, predicción meteorológica, etc. A pesar de ello, estos modelos muestran una serie de limitaciones que deben ser tratadas antes de ser utilizados para poblar mapas semánticos, a saber: son computacionalmente intratables cuando la complejidad del problema a modelar incrementa (en este caso, cuando el número de objetos/habitaciones en el entorno y sus posibles tipos crece), necesitan una considerable cantidad de datos de entrenamiento para ajustar modelos exitosos, y son incapaces de detectar resultados incoherentes así como de aprender de experiencias pasadas.

Contribuciones

Las contribuciones de la presente tesis tratan de solucionar las limitaciones de los mapas semánticos tradicionales anteriormente comentadas mediante el uso de técnicas probabilísticas. Concretamente, los objetivos de la tesis, que tuvieron como fruto el desarrollo de dichas técnicas, fueron definidos como:

- **Desarrollo de un sistema de reconocimiento completo:** Proveer algoritmos probabilísticos para el reconocimiento de objetos y/o habitaciones manejando información tanto de contexto como incierta, en los cuales también se considere conocimiento semántico, con el objetivo de presentar una serie de características deseables como escalabilidad, detección de resultados erróneos, aprendizaje de experiencias pasadas, etc.
- **Mejora de los mapas semánticos para el manejo incertidumbre:** Acomodar los resultados probabilísticos de dichos algoritmos en una novedosa representación de mapas semánticos, de tal modo que un robot pueda explotarlos para conseguir una noción de la certeza del mismo sobre su comprensión del entorno de trabajo, permitiéndole operar de un modo más coherente.

De este modo, las contribuciones de esta tesis pueden agruparse en dos temas principales: comprensión contextual de la escena, y mapeo semántico de la misma.

Contribuciones a la comprensión contextual de la escena

El primer grupo de contribuciones, presentadas en los artículos [114, 116, 119, 117, 115, 122, 121], se centra en el problema del reconocimiento de objetos y/o habitaciones empleando información contextual. Los PGMs en general, y los Campos Aleatorios Condicionales (*Conditional Random Fields* o *CRFs* en inglés) en particular, son usados para modelar este problema desde un punto de vista holístico, considerando las

relaciones de contexto entre objetos y/o habitaciones, y tratando de manera formal la incertidumbre inherente al proceso de reconocimiento. Su aplicabilidad al problema tratado ha sido verificada tras una exhaustiva evaluación de los algoritmos más populares tanto de entrenamiento como de inferencia probabilística sobre dichos modelos.

Estos *CRFs* trabajan en conjunción con *KBs*, lo que permite mantener sus ventajas cuando trabajan por separado a la vez que se mitigan sus limitaciones:

- Las *KBs* dotan a los *CRFs* con capacidades para: reducir su complejidad, explotar información a priori sobre el dominio del problema, verbalizar sus resultados, generar un numero aleatorio de ejemplos de entrenamiento sintéticos para su ajuste, detectar resultados incoherentes, y aprender de la experiencia del robot.

- Los *CRFs* permiten a las *KBs* manejar información incierta y explotar relaciones de contexto de acuerdo con una base teórica fundamentada.

Los resultados devueltos durante la evaluación de los métodos desarrollados han sido comparados con los de otras soluciones punteras empleando conjuntos de datos del estado del arte. Además, se ha reunido y hecho público un nuevo repositorio de datos, llamado *UMA-Offices*, consistente en observaciones tridimensionales de 25 habitaciones de nuestro entorno de oficinas. También se ha implementado la librería software de código abierto *Undirected Probabilistic Graphical Models in C++*[5] (UPGMpp) con el fin de manejar eficientemente los PGMs.

Contribuciones al mapeo semántico

El objetivo del segundo grupo de contribuciones, presentadas en los artículos [120, 118, 123], es el de acomodar los resultados probabilísticos provenientes de las técnicas anteriores en una representación semántica del entorno. Para ello se ha desarrollado la representación *Multiversal Semantic Map* (*MvSmap*), la cual permite considerar diferentes interpretaciones del entorno de trabajo del robot en forma de *universos*, también almacenando información sobre la probabilidad de que sean las interpretaciones correctas. Esto permite al robot tener en cuenta no sólo el universo más probable, sino otros que también muestran una alta probabilidad de ser válidos. Este novedoso mapa se acompaña de técnicas para mantener tratable el número de universos considerados, de tal manera que sea aplicable a entornos complejos con numerosos objetos y habitaciones.

La idoneidad de los *MvSmaps*, así como su capacidad para manejar datos inciertos de una manera eficiente, se ha comprobado empleando el novedoso conjunto de datos *Robot@Home*, acumulado por un robot móvil al explorar una serie de entornos domésticos. Además, el conjunto de herramientas *Object Labeling Toolkit*[6] (OLT), disponible públicamente para la comunidad investigadora, ha sido desarrollado para

[5] `http://mapir.isa.uma.es/work/upgmpp-library`
[6] `http://mapir.isa.uma.es/work/object-labeling-toolkit`

procesar de manera fácil y rápida conjuntos de datos formados por secuencias de información sensorial, como es el caso de *Robot@Home*.

Publicaciones

La presente tesis ha dado lugar a las siguientes publicaciones:

Revistas

- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, y Javier Gonzalez-Jimenez*. **Building Multiversal Semantic Maps for Mobile Robot Operation**. Enviado a Knowledge-Based Systems (2016).
- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, y Javier Gonzalez-Jimenez*. **A Survey on Learning Approaches for Undirected Graphical Models. Application to Scene Object Recognition**. En International Journal of Approximate Resoning, (aceptado, por aparecer) (2016).
- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, y Javier Gonzalez-Jimenez*. **Robot@Home, a Robotic Dataset for Semantic Mapping of Home Environments**. Enviado a International Journal of Robotics Research (2016).
- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, y Javier Gonzalez-Jimenez*. **Scene Object Recognition for Mobile Robots Through Semantic Knowledge and Probabilistic Graphical Models**. En Expert Systems with Applications, vol. 42, no. 22, pp. 8805–8816, (2015).
- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, y Javier Gonzalez-Jimenez*. **Exploiting Semantic Knowledge for Robot Object Recognition**. En Knowledge-Based Systems, vol. 86, pp. 131–142, (2015).

Conferencias

- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, y Javier Gonzalez-Jimenez*. **Probability and Common-Sense: Tandem Towards Robust Robotic Object Recognition in Ambient Assisted Living**. En 10th International Conference on Ubiquitous Computing & Ambient Intelligence, Las Palmas de Gran Canaria, Spain, (2016).
- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, y Javier Gonzalez-Jimenez*. **Joint Categorization of Objects and Rooms for Mobile Robots**. En IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, (2015).
- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, y Javier Gonzalez-Jimenez*. **OLT: A Toolkit for Object Labeling Applied to Robotic RGB-D Datasets**. En European Conference on Mobile Robots (ECMR), Lincoln, UK, (2015).

- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, y Javier Gonzalez-Jimenez*. **UPGMpp: a Software Library for Contextual Object Recognition**. En 3rd. Workshop on Recognition and Action for Scene Understanding (REACTS), Valletta, Malta, (2015).

- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, y Javier Gonzalez-Jimenez*. **Mobile Robot Object Recognition through the Synergy of Probabilistic Graphical Models and Semantic Knowledge**. En European Conference on Artificial Intelligence, Workshop on Cognitive Robotics (CogRob), Prague, Czech Republic, (2014).

Marco de la tesis

Esta tesis es el resultado de 5 años de trabajo del autor como miembro del grupo *Machine Perception and Intelligent Robotics*[7] (MAPIR), el cual se encuentra dentro del departamento de *Ingeniería de Sistemas y Automática* de la *Universidad de Málaga*. La investigación realizada ha sido principalmente financiada por el programa de ayudas *Formación de Profesorado Universitario* (FPU), promovido por el *Ministerio de Educación*.

Durante este periodo, el autor completó con éxito el programa doctoral en *Ingeniería Mecatrónica*, coordinado por el mismo departamento del que es miembro, donde obtuvo un conocimiento sólido sobre los pilares fundamentales de la robótica: sistemas de control, sistemas electrónicos, sistemas mecánicos, y ordenadores. Esta educación académica fue completada con distintos cursos, como es el caso de *Writing in the sciences*, impartido por la *Universidad de Stanford*, y la participación en la *Primera Örebro Winter School en Artificial Intelligence and Robotics*, la cual pretende acercar dos campos estrechamente relacionados como son los de la Inteligencia Artificial y la Robótica. Esta escuela también hizo posible el conocer otros investigadores en el mismo campo de estudio, relaciones que se mantienen a día de hoy.

El autor también completó una estancia de tres meses en el *Knowledge-Based Systems Research Group*[8], en la *Universidad de Osnabrück* en Alemania, durante el año 2014, bajo la supervisión de *Prof. Dr. Joachim Hertzberg*. Durante este tiempo la investigación realizada se centró en el análisis y la implementación de diferentes algoritmos para el manejo eficiente de *PGMs*, así como de su aplicación para el reconocimiento *online* de objetos en robots móviles. En esta gran experiencia también se establecieron colaboraciones con distintos miembros del grupo receptor.

Además, también cabe destacar que el autor ha estado activo en el proceso de revisión de artículos de conferencias y revistas prestigiosas, como es el caso de las conferencias *International Conference on Robotics and Automation* (ICRA, 2014, 2015, 2016), e *International Conference on Intelligent Robots and Systems* (IROS, 2015), o las revistas *Association for the Advancement of Artificial Intelligence* e *Intelligent Service Robotics*.

[7] http://mapir.isa.uma.es/
[8] www.inf.uos.de/kbs/

La beca FPU también ofreció al autor la oportunidad de colaborar como profesor asistente con el departamento del que es miembro. Concretamente, impartió docencia en la asignatura de *Robótica* en la *Escuela Técnica Superior de Ingeniería Informática*, en la *Universidad de Málaga*. También supervisó el trabajo fin de grado de un estudiante, David Zúñiga, titulado *Visual SLAM with RGB-D Cameras Based on Pose Graph Optimization*.

Además de la investigación presentada en esta tesis, el autor también ha participado en otros proyectos dentro del grupo MAPIR, algunos de ellos de temática relacionada:

- **TCS: Tunnel Continuous Setout** (Nov'08 – Jul'11): Este proyecto se centró en el desarrollo de un sistema para el replanteo automático de secciones de túneles a ser perforadas. El prototipo del sistema, que toma el mismo nombre que el proyecto, combina una unidad de escaneo que realiza mediciones sobre el frente de excavación y un láser proyector que continuamente muestra la sección del túnel a perforar. La parte más desafiante del proyecto fue la implementación de las técnicas de calibración para localizar con exactitud todos los componentes del sistema dentro de un marco de referencia global.

- **ExCITE: Enabling SoCial Interaction Through Embodiment** (Jul'10 – Jun'13): El rol del autor en este proyecto estuvo relacionado con el desarrollo de mejoras técnicas para la plataforma robótica de telepresencia *Giraff*: un manejo más simple y seguro, detección de obstáculos, y visualización de la posición del robot en un mapa esquemático del lugar visitado. Una arquitectura de control, llamada *Navigation Assistant* (NAS), fue desarrollada para cumplir con estas necesidades especiales.

- **Taroth: New developments toward a Robot at Home** (Ene'12 – Dic'15): Este proyecto persiguió tres objetivos principales: i) aumentar la independencia del robot en cuanto a su movimiento, ii) integrar y explotar información semántica para mejorar la autonomía del robot y permitirle interactuar con humanos, y iii) desarrollar una arquitectura de control robótica para el manejo de servicios de la llamada *Ambient Assisted Living*, como son el entretenimiento, la domótica, las relaciones sociales, la seguridad, etc.

- **IRO: Improvement of the sensorial and autonomous capability of Robots through Olfaction** (Ene'14 – Feb'19): La investigación en este proyecto se orienta al estudio de mecanismos para usar información olfativa en problemas como el reconocimiento de objetos y la interpretación de la actividad en una escena. Dicho estudio presta especial atención al rol de la información semántica en los procesos de percepción por parte del robot y toma de decisiones, persiguiéndose una mejora en términos de eficiencia, autonomía y utilidad.

Del trabajo del autor en estos proyectos se desprendieron una serie de publicaciones adicionales:

Revistas

- *Javier Gonzalez-Jimenez, Vicente Arévalo, Cipriano Galindo, y Jose-Raul Ruiz-Sarmiento*. **An Automated Surveying and Marking System for Continuous Setting-out of Tunnels**. En Computer-Aided Civil and Infrastructure Engineering, vol. 31, no. 3, pp. 219–228, (2016).

Conferencias

- *David Zuñiga-Noël, Jose-Raul Ruiz-Sarmiento, y Javier Gonzalez-Jimenez*. **Detección de Lugares con Cámaras RGB-D. Aplicación a Cierre de Bucles en SLAM**. En XXXVII Jornadas de Automática, Madrid, Spain, (2016).
- *Javier Gonzalez-Jimenez, Jose-Raul Ruiz-Sarmiento, y Cipriano Galindo*. **Improving 2D Reactive Navigators with Kinect**. En 10th International Conference on Informatics in Control, Automation and Robotics (ICINCO), Reykjavic, (Iceland, 2013).
- *Javier Gonzalez-Jimenez, Cipriano Galindo, Francisco Melendez-Fernandez, y Jose-Raul Ruiz-Sarmiento*. **Building and Exploiting Maps in a Telepresence Robotic Application**. En 10th International Conference on Informatics in Control, Automation and Robotics (ICINCO), Reykjavic, Iceland, (2013).
- *Javier Gonzalez-Jimenez, Cipriano Galindo, y Jose-Raul Ruiz-Sarmiento*. **Technical Improvements of the Giraff Telepresence Robot Based on Users' Evaluation**. En The 21st IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), Paris, France, (2012).
- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, y Javier Gonzalez-Jimenez*. **Cámaras basadas en tiempo de vuelo. Uso en la mejora de métodos de detección de caras.** En XXXII Jornadas de Automática, Sevilla, Spain, (2011).

Informes técnicos

- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, y Javier Gonzalez-Jimenez*. **Experimental Study of the Performance of the Kinect Range Camera for Mobile Robotics**. Universidad de Malaga, Andalucia Tech, Departamento de Ingenieria de Sistemas y Automatica, (2013).

Estructura de la tesis

Mas allá del capítulo introductorio (**Chapter 1: Introduction**) el resto de capítulos en la primera parte de esta tesis (**Part I: Thesis description**) están organizados como sigue:

Chapter 2: Theoretical background provee nociones básicas sobre la teoría detrás de dos herramientas intensivamente empleadas en esta tesis: *PGMs* y *KBs*, de tal modo que el lector no experto en estas materias pueda obtener un conocimiento básico para una mejor comprensión de los siguientes capítulos. El autor ha tratado que sea una lectura lo más amena posible.

Chapter 3: Contextual scene understanding describe los enfoques tradicionalmente seguidos para el reconocimiento de objetos y habitaciones por parte de un robot móvil, y de que modo están relacionados con las contribuciones presentadas. También se dan detalles sobre la sinergia entre *PGMs* y *SK* codificado en *KBs* persiguiendo el entendimiento de escenas. Este capítulo también discute los repositorios de datos empleados para evaluar las técnicas desarrolladas, incluyendo *UMA-Offices*, así como el software implementado para manejar *PGMs*.

Chapter 4: Semantic Mapping esboza las representaciones de mapas semánticos comúnmente empleadas en robótica móvil, y describe las contribuciones de esta tesis en relación a una representación capaz de manejar información incierta: el *Multiversal Semantic Map*. Las virtudes de dicho mapa han sido comprobadas empleando un novedoso dataset, *Robot@Home*, cuyas características son descritas en este capítulo, junto con las del software usado para su procesamiento: *Object Labeling Toolkit*.

Chapter 5: Summary of included papers lista los artículos que conforman la segunda parte de esta tesis, **Part II: Included papers**, describiendo brevemente su contenido y el papel del autor en los mismos.

Chapter 6: Conclusions and future work discute las conclusiones que se pueden extraer del trabajo realizado, así como las líneas de investigación que quedan abiertas e interesantes extensiones a dicho trabajo.

Publicaciones incluidas en la Tesis

Esta sección realiza un esbozo de los artículos incluidos en la segunda parte de la tesis, así como las contribuciones del autor a cada uno de ellos.

Artículo A: Aprendiendo *Conditional Random Fields* con datos provenientes de *Semantic Knowledge*

Descripción: Este trabajo estudia la aplicabilidad de *CRFs* entrenados con datos sintéticos, generados a partir de *SK*, al problema del reconocimiento de objetos explotando su contexto. El objetivo de este enfoque para el entrenamiento es el de evitar la recopilación de datos reales para ajustar sistemas de reconocimiento. Dicha recopilación es una tarea pesada que requiere de una alta dedicación temporal, además de

no ser realizable en ciertos entornos, ya que los datos recogidos deben ser suficientemente representativos del dominio del problema. Para solucionar esta cuestión se codifica *SK* en una Ontología, la cual define las clases (o tipos) de objetos del dominio de discurso (por ejemplo, en el dominio del hogar, ejemplos de estos tipos serían *horno*, *microondas*, *salón*, o *cocina*), sus propiedades y sus relaciones, y es usado para generar ejemplos de entrenamiento sintéticos. La conveniencia del método de aprendizaje propuesto debe ser comprobada empleando conjuntos de datos reales, por lo que UMA-Offices y NYUv2 [131] formaron el banco de pruebas necesario para responder a preguntas como: *¿Cuánto contribuyen las relaciones de contexto al éxito del método?*, *¿Cómo afecta el tamaño del conjunto de datos de entrenamiento al rendimiento?*, o *¿Capturan los datos sintéticos generados características y relaciones reales?*.

Contribución del autor: Estudió el estado del arte sobre *PGMs* y *KBs* abordando el problema del reconocimiento de los objetos de una escena. Diseñó el modo de codificar información relevante en la Ontología para su posterior aprovechamiento. Implementó el algoritmo para la generación automática de un número arbitrario de ejemplos de entrenamiento. Procesó el conjunto de datos UMA-Offices, y realizó los experimentos necesarios para demostrar la validez de la propuesta.

Artículo B: Categorización conjunta de objetos y habitaciones

Descripción: En este artículo se extienden los métodos desarrollados en el anterior trabajo para también considerar las habitaciones del entorno. Motivado por estudios recientes que destacan la conveniencia de modelar conjuntamente los problemas de reconocimiento de objetos y habitaciones (dada la influencia mutua que tienen los tipos de los objetos reconocidos y los tipos de las habitaciones), la Ontología definida en el Artículo A es aumentada para también incluir tipos de habitaciones, sus atributos, y relaciones entre ellas así como entre objetos y habitaciones. Un ejemplo de esta información sería que los dormitorios están usualmente conectados con pasillos y suelen contener camas. Los *CRFs* también son convenientemente adaptados para trabajar con diferentes tipos de variables aleatorias (representando categorías de objetos o habitaciones) y relaciones de contexto. Para validar el método se emplean escenas ilustrando entornos domésticos dentro del conjunto de datos NYUv2.

Contribución del autor: Estudió las técnicas en el estado del arte para modelar conjuntamente los problemas de reconocimiento de objetos y habitaciones. Diseñó la expansión de la Ontología en el artículo anterior, así como de los *CRFs* y el algoritmo implementado para la generación de ejemplos sintéticos. Realizó los experimentos que soportan las afirmaciones del trabajo.

Artículo C: Empleando *Semantic Knowledge* para un reconocimiento eficiente y coherente

Descripción: La complejidad de los *CRFs* aumenta considerablemente cuando se aplican a escenarios repletos de objetos. Esto implica la utilización de técnicas de inferencia aproximada para obtener los resultados de reconocimiento, lo que en algunos casos compromete el éxito del método en comparación con el uso de soluciones de inferencia exacta. Este artículo propone la utilización de *SK* para reducir la complejidad del proceso de inferencia. Dicho conocimiento, codificado de nuevo en una Ontología, se aprovecha para generar hipótesis sobre los tipos más probables a los que pueden pertenecer los objetos en la escena, empleando para ello sus características. Estas hipótesis son consideradas por el *CRF* como las únicas categorías candidatas posibles, reduciendo de este modo la complejidad del proceso de inferencia, incluso habilitando en ciertos casos la inferencia exacta. Adicionalmente, también se codifica en la Ontología información a priori sobre la frecuencia de aparición de los distintos tipos de objetos. Esta información muestra que, por ejemplo, en un entorno de oficinas es más probable encontrar un ordenador a un sofá, mientras que es bastante improbable encontrar una tabla de planchar. El artículo también propone una modificación a la formulación usual de los *CRFs* para el aprovechamiento de dicha información. La ganancia en cuanto a la eficiencia y coherencia proporcionada por esta solución es medida con los conjuntos de datos UMA-Offices y NYUv2.

Contribución del autor: Diseñó el marco para, empleando las hipótesis generadas mediante inferencia lógica sobre la Ontología, reducir la complejidad del modelo probabilístico. Adaptó la formulación de los *CRFs* para también considerar información previa sobre la frecuencia de aparición de los diferentes tipos de objetos desde la Ontología. Evaluó la reducción de complejidad conseguida y la mejora en cuanto a la coherencia de los resultados devueltos empleando dos repositorios de datos distintos.

Artículo D: Libería UPGMpp para manejar *Conditional Random Fields*

Descripción: Este trabajo presenta la librería *Undirected Probabilistic Graphical Models in C++* (UPGMpp), un paquete software para trabajar con este tipo de modelos probabilísticos. La librería está especialmente diseñada e implementada para ser eficiente a la hora de tratar el problema del reconocimiento de objetos y/o habitaciones. El artículo describe cómo usar el software para modelar este problema, y presenta sus tres partes fundamentales: *base* (implementa la funcionalidad para construir y manipular modelos gráficos), *training* (permite la definición de conjuntos de datos para entrenar los modelos), e inferencia (implementa algoritmos de inferencia probabilística). Para mostrar la flexibilidad y usabilidad de la librería, este trabajo ilustra los procesos necesarios para entrenar y testear – realizar inferencia sobre – *PGMs*, incluyendo ejemplos de código. También se reportan los resultados de reconocimiento devueltos por distintos métodos de inferencia al tratar con escenas del conjunto de

datos NYUv2, así como el tiempo de ejecución requerido por dichos métodos.

Contribución del autor: Estudió la teoría detrás de los *PGMs* no dirigidos, así como otras librerías relacionadas para tratar con los mismos. Diseñó e implementó las partes de la librería, con el objetivo de que fueran eficientes, versátiles, extensibles, y fáciles de usar. Hizo la librería pública, ejemplificó su uso, y realizó las mediciones sobre tiempos de ejecución y éxito del reconocimiento.

Artículo E: Conjunto de herramientas para el tratamiento de repositorios de datos con información RGB-D

Descripción: En este trabajo se presenta el conjunto de herramientas software *Object Labeling Toolkit* (OLT), desarrollado para el procesamiento eficiente de repositorios de datos compuestos de secuencias de observaciones RGB-D (intensidad, RGB, más profundidad, D), capturadas por un número arbitrario de sensores de este tipo. Para ello, OLT construye una reconstrucción 3D de cada secuencia de observaciones y permite al usuario, mediante una interfaz gráfica, anotar los objetos y habitaciones en dicha reconstrucción con el tipo al que pertenecen (cama, mesa, lámpara, cocina, etc.). El artículo describe sus componentes principales, a saber: pre-procesamiento del conjunto de datos, construcción de mapa 2D, localización de las poses de las observaciones, visualización secuencial, etiquetado de la escena, y propagación automática de etiquetas a cada observación individual, de los cuales sólo el etiquetado de la escena requiere la intervención de un operador humano. También se ejemplifica el uso de OLT para el etiquetado fácil y rápido de dos secuencias de observaciones RGB-D, analizando sus virtudes con respecto a una técnica de etiquetado tradicional.

Contribución del autor: Diseñó el conjunto de herramientas. Estudió e implementó/adaptó las técnicas necesarias para los procedimientos de: procesado de imágenes tanto RGB como de profundidad, construcción de mapas geométricos 2D, reconstrucción de escenas 3D, visualización e interacción con las reconstrucciones, y propagación automática de las anotaciones a través de las secuencias de observaciones. Comparó el tiempo ahorrado empleando OLT con respecto al uso de una técnica de etiquetado típica.

Artículo F: Mapa semántico capaz de manejar incertidumbre

Descripción: En este artículo se propone un mapa semántico novedoso que permite la manipulación de incertidumbre, también aprovechando las relaciones contextuales de los elementos espaciales en el entorno del robot (objetos y habitaciones). Esta representación adopta el nombre de *Multiversal Semantic Map* (*MvSmap*). El artículo proporciona un estudio completo sobre otros enfoques para realizar un mapeo semántico del entorno, así como de técnicas para poblar dichos mapas. Los *MvSmaps* son descritos en detalle y definidos formalmente, incluyendo los algoritmos necesarios para su construcción, donde las técnicas de reconocimiento desarrolladas en trabajos

previos tienen un rol principal. Además, este trabajo estudia algoritmos para tratar eficientemente la incertidumbre modelada en estos mapas. Finalmente, el conjunto de datos Robot@Home [123] es el elegido para evaluar el rendimiento de los distintos sistemas envueltos en la construcción de *MvSmaps*.

Contribución del autor: Diseñó la representación *Multiversal Semantic Map* para el almacenamiento y tratamiento de información incierta. Integró las técnicas de reconocimiento de objetos y habitaciones anteriormente desarrolladas en un sistema para poblar dichas representaciones. Diseñó e implementó el proceso para la construcción de *MvSmaps* de acuerdo a la información percibida por un robot móvil. Procesó el conjunto de datos Robot@Home para que fuera útil durante el testeo de los sistemas en este trabajo.

Conclusiones y líneas futuras

Esta tesis ha explorado y hecho contribuciones al fascinante mundo del mapeo semántico del entorno por medio de un robot móvil. Este tipo de mapas dotan al robot de herramientas para comprender cuales son los elementos y espacios que tiene a su alrededor, así como sus propiedades, lo cual sienta las bases para una operación inteligente, autónoma y eficiente. En la investigación llevada a cabo se ha prestado especial atención a la población de mapas semánticos con información sobre los elementos espaciales en el entorno de trabajo del robot, es decir objetos y habitaciones, a través de la combinación de técnicas de los campos del *Aprendizaje Automático* y la *Inteligencia Artificial*. Estos campos se encuentran actualmente en un momento dulce, donde los estudios y aplicaciones en las que son utilizados sigue creciendo, tal y como apuntó en una reciente entrevista uno de los directivos de Amazon, Ralf Herbrich, afirmando que *"Estamos en una edad dorada para el aprendizaje automático y la inteligencia artificial. Nos encontramos aún lejos de hacer cosas del mismo modo en el que los humanos las hacen, pero estamos solventando problemas increíblemente complejos cada día y consiguiendo un progreso increíblemente rápido"*. En opinión del autor, la investigación de sistemas que aprovechen la sinergia de sendos campos, potenciando sus ventajas y mitigando sus limitaciones, puede llevar a avances notables en la comunidad robótica. Este es el caso de las técnicas desarrolladas en la presente tesis.

Para que un robot móvil alcance un cierto grado de consciencia del entorno en el que se desenvuelve, este debe ser capaz de reconocer los elementos espaciales observados a través de su sistema sensorial. El primer grupo de contribuciones de esta tesis trata este tema, centrándose en la combinación de *Conditional Random Fields* (*CRFs*), una variante discriminativa no dirigida de los *Probabilic Graphical Models* (*PGMs*), y *Semantic Knowledge* (*SK*) del dominio de discurso codificado en una Ontología. Ambos enfoques han alcanzado un éxito notable en distintos problemas de clasificación.

Por un lado, los *CRFs* permiten el modelado de relaciones de contexto entre elementos espaciales, al mismo tiempo que maneja la incertidumbre proveniente del

sistema sensorial del robot y de los modelos empleados para definir el problema. Estos modelos también permiten la ejecución de métodos de inferencia probabilística. Precisamente, una de las primeras contribuciones de esta tesis fue la librería *Undirected Probabilistic Graphical Models in C++* (UPGMpp), desarrollada como consecuencia de la ausencia de herramientas software para manejar *PGMs* no dirigidos en general, y *CRFs* en particular, proveyendo las características que demanda un sistema de reconocimiento ejecutándose en un robot móvil (*e.g.* eficiencia, flexibilidad, o facilidad de integración). Esta librería, disponible públicamente, implementa algoritmos populares para la construcción, aprendizaje e inferencia sobre modelos gráficos. Las posibles combinaciones de métodos para entrenar e inferir información sobre *CRFs* motivó el estudio de diferentes estrategias de aprendizaje, el cual reportó valiosas conclusiones no sólo para la correcta utilización de estos modelos en el resto de contribuciones, sino para su empleo por parte de cualquier miembro de la comunidad robótica que desee configurar rápidamente un sistema de reconocimiento tan exitoso como sea posible.

A pesar de su notoria utilización en distintos campos, los *CRFs* muestran una serie de limitaciones a la hora de ser aplicados al problema de reconocimiento. En primer lugar, para ser correctamente entrenados requieren una considerable cantidad de ejemplos (datos) que, además, cubran por completo los elementos dentro del dominio de trabajo. La recogida de dichos conjuntos de datos es una tarea tediosa y que requiere una alta dedicación temporal, además de ser irrealizable en algunos dominios, tal y como experimentó el autor al procesar el repositorio *UMA-Offices*. Dicho conjunto de datos contiene 25 escenas capturadas por un robot móvil en entornos de oficinas de la Universidad de Málaga, y se recogió con el fin de evaluar las técnicas de reconocimiento desarrolladas – de manera conjunta con otros repositorios del estado del arte. Para evitar la dependencia de conjuntos de datos conteniendo información real, se mostró como *SK*, convenientemente codificado en una Ontología, puede usarse para generar sin esfuerzo una cantidad arbitraria de datos de entrenamiento representativos del dominio de discurso. Las Ontologías suponen una manera natural de codificar *SK*, además de ser compactas, leíbles por un humano, y directamente utilizables en tareas de razonamiento de alto nivel. No obstante, son incapaces de manejar incertidumbre, y es complejo dar el salto de información sensorial de bajo nivel a información codificada sin emplear procesos *ad-hoc*. Su combinación con *CRFs* elimina estas limitaciones, sentando las bases de una relación de beneficio mutuo.

En esta tesis se ha mostrado como las Ontologías que codifican *SK* tienen mucho más que ofrecer en su matrimonio con *CRFs*. Por ejemplo, se han empleado para generar hipótesis sobre los posibles tipos de objetos/habitaciones en una escena, reduciendo drásticamente la complejidad de los *CRFs* cuando modelan dicha escena. Esto incrementa la eficiencia de los métodos de inferencia aproximada sobre *CRFs*, así como amplía el abanico de escenarios donde es posible realizar una inferencia exacta. Nótese que la eficiencia del método de reconocimiento es fundamental para el apropiado funcionamiento del robot, ya que este debe compartir los (usualmente limitados) recursos del robot con otros algoritmos en ejecución, como puedan ser los de navegación o localización. Además, las Ontologías pueden codificar distintos tipos de

información sobre los elementos del dominio, lo cual se ha aprovechado para definir la frecuencia de aparición de los distintos tipos de objetos. La usual formulación de los *CRFs* ha sido consecuentemente adaptada para explotar esta fuente de información, permitiendo a estos modelos alcanzar unos resultados de reconocimiento más coherentes. El *SK* también se ha empleado para la detección de incoherencias en los resultados, y para aprender de las mismas en colaboración con un humano. Este enfoque soluciona la incapacidad de los *CRFs* para aprender de experiencias pasadas, y les permite mejorar su rendimiento y robustez a largo plazo en su aplicación a entornos humanos.

Una vez desarrolladas las técnicas para el reconocimiento, estas fueron integradas en un sistema de mapeo semántico. Para ello se diseñó una novedosa representación del entorno llamada *Multiversal Semantic Map* (*MvSmap*), la cual es capaz de acomodar y aprovechar los resultados probabilísticos de los métodos de reconocimiento. Dicho mapa considera diferentes interpretaciones de los elementos espaciales, o *universos*, como instancias de Ontologías, creándose un *multiverso*. Estas Ontologías son además automáticamente anotadas con las probabilidades devueltas por el sistema de reconocimiento, así como con su probabilidad de ser las interpretaciones correctas. De este modo, el desempeño del robot no se limita a la utilización del universo más probable, *modus operandi* de los mapas semánticos tradicionales, sino que también puede considerar otras posibles explicaciones con diferentes interpretaciones semánticas. Además se discutió una estrategia para mantener tratable el número de universos considerados, clave para la eficiencia de esta representación semántica.

También se han hecho públicos dos recursos relacionados con las técnicas de mapeo semántico. El primero se corresponde con el conjunto de datos *Robot@Home*, el cual contiene, entre otros: 87,000+ observaciones recogidas en distintas casas por un robot móvil dotado de un aparejo con 4 cámaras RGB-D y un escáner láser 2D, reconstrucciones tanto en 2D como en 3D de las escenas exploradas, información topológica sobre la conectividad de las habitaciones, y anotaciones sobre los tipos de los objetos y habitaciones percibidos. El repositorio de datos es rico en información contextual de los elementos espaciales antes mencionados, una característica que no se encuentra en la mayoría de los repositorios actuales, lo cual puede ser aprovechado por sistemas de mapeo semántico. La segunda contribución a este respecto es el conjunto de herramientas denominado *Object Labeling Toolkit* (OLT), diseñado para procesar eficientemente repositorios de datos compuestos de secuencias de observaciones RGB-D. Estas herramientas son altamente customizables y expansibles, facilitando la integración de algoritmos ya desarrollados, y han mostrado su utilidad para reducir drásticamente el tiempo y esfuerzo necesarios para procesar repositorios conteniendo ese tipo de información. Por ejemplo, OLT fue usado para el procesamiento de Robot@Home.

Como observación final, cabe destacar que aunque las técnicas descritas en esta tesis han sido evaluadas con conjuntos de datos provenientes de entornos domésticos y de oficinas, su utilización no se limita a esos dominios, sino que pueden ser empleadas en cualquier escenario que exhiba información semántica como pueda ser el caso de hospitales o centros comerciales. También es interesante añadir que su uso no

está restringido al campo de la robótica móvil, sino que podrían ser exportadas a otros campos que se pudieran beneficiar de la explotación de mapas semánticos tales como asistencia a invidentes o personas mayores, realidad aumentada, y otras aplicaciones por venir en la era de los dispositivos portátiles con gran capacidad de cómputo. Hoy en día, de hecho, nuestros teléfonos móviles son casi tan potentes como los ordenadores de sobremesa. Los esfuerzos en la investigación en mapeo semántico, junto con los avances tecnológicos, nos aseguran la aparición de apasionantes y rompedoras aplicaciones. ¡Manténgase atento!.

Trabajos futuros

El trabajo realizado en la presente tesis deja abiertas una serie de líneas de investigación y expansiones. Algunas de las más relevantes se describen a continuación.

Generación de hipótesis. La generación de hipótesis empleando la información codificada en la Ontología podría ser demasiado restrictiva en algunas situaciones, principalmente con objetos que muestran unas características particulares. Supóngase una escena con un libro en el suelo. En esta situación el razonador lógico no devolvería la clase *libro* como hipótesis, dado que su altura desde el suelo difiere en gran medida de la esperada. Una opción podría ser considerar el resultado del proceso de inferencia lógica como una puntuación a ser considerada en la formulación de los *CRFs*, a expensas de comprometer la opción de inferencia exacta.

Aprovechamiento de los MvSmaps. El potencial real de los *Multiversal Semantic Maps* (en opinión del autor) está aún por verse. Se han diseñado y realizado diversas pruebas de concepto en tareas típicamente robóticas, pero debe estudiarse en mayor detalle el beneficio de estos mapas en problemas reales como navegación eficiente y búsqueda de objetos, localización del robot, planificación de tareas con información incierta/incompleta, etc.

Aprendiendo de experiencias. El sistema propuesto para el aprendizaje en base a la experiencia acumulada puede ser ampliado en diferentes aspectos. Primero, debe realizarse una evaluación rigurosa del sistema empleando complejos *CRFs* y Ontologías, incluyendo información de objetos y habitaciones, a lo largo de extensos periodos de tiempo. También podría estudiarse, dado que un humano forma parte del bucle de aprendizaje, cómo afectan al rendimiento del sistema posibles instrucciones incorrectas por parte del usuario. Además el sistema también se podría beneficiar de un estudio acerca de cuándo sería más apropiado preguntar a dicho humano sobre un resultado incoherente, de tal manera que se le moleste lo mínimo posible.

Posibles desarrollos dentro de UPGMpp. Sería interesante explorar algunas características adicionales relacionadas con el rendimiento de UPGMpp. Por ejemplo, aunque las partes que requieren más tiempo de ejecución han sido paralelizadas empleando *OpenMP*, algunas operaciones repetitivas que utilicen datos de forma ma-

siva podrían beneficiarse de su ejecución en núcleos GPU empleando, por ejemplo, *CUDA* u *OpenCL*. También sería útil el contar con herramientas gráficas para visualizar y modificar los grafos de los *PGMs*, así como para comprender cómo evolucionan en tiempo de ejecución. También se contempla la incorporación de técnicas para la generación de muestras de la distribución de probabilidad definida por un *PGM* (como *Markov Chain Monte Carlo*). Por supuesto, es bienvenida cualquier contribución a esta librería por parte de la comunidad robótica o de visión por computador.

Mejoras a OLT. La incorporación de algoritmos para un registro globalmente consistente de las observaciones RGB-D en una secuencia podría dar lugar a reconstrucciones incluso más precisas. La experiencia de usuario también se podría mejorar considerando otras primitivas geométricas para segmentar y etiquetar escenas, además de las cajas empleadas actualmente, como puedan ser esferas o cilindros. Por último, el tiempo necesario para el etiquetado también podría reducirse si se ofreciera al usuario una segmentación inicial de la escena, así como etiquetas tentativas para los objetos/habitaciones apareciendo en la misma.

Punto y aparte

Esta sección concluye el resumen de la presente tesis, *Probabilistic Techniques in Semantic Mapping for Mobile Robotics*. El lector puede continuar con los siguientes capítulos, en el idioma inglés, donde se describen en mayor detalle las contribuciones de la misma.

Part I

Thesis description

1 Introduction

Winter is coming. A servant robot senses that the temperature is decreasing and takes a blanket to a lovely grandma. In the same building, another robot patrolling an offices' floor is alerted by a light turned on in a room; rapidly it notices that the research fellow, Bob, is working late in the night, the third time that week. Meanwhile, baby Alice, Bob's daughter, is sad because of the absence of her daddy, and her robotic colleague warmly nicknamed as *Roboto* looks for her favorite teddy. Sophie, Alice's mom, is also counting the hours to see her, and commands a robot to clean the tables once the restaurant she runs is closed to the public.

These scenarios are some examples where mobile robots, to a greater or lesser extent, can provide a number of services for raising the standards of living. Nowadays, it becomes clear that robots are coming to stay, as it is shown by their remarkable application to an increasing number of tasks where they collaborate with humans or release them from tedious or hazardous chores, such as surveillance, health care, companion, entertainment, household maintenance, etcetera [97]. Figure 1.1 depicts some examples of modern robots aimed at performing some of these tasks. Common to all these robotic applications is the necessity of building representations of the working environment, commonly referred to as *maps*, which permit a mobile robot to be aware of its surroundings in order to navigate avoiding obstacles, localize itself with respect to a given reference frame, store relevant information about spatial elements for accomplishing its goals, etc.

Traditional spatial representations, like geometric, topological, or hybrid maps, are extensively used due to the core skills they provide, *i.e.* navigation and localization. Nevertheless, the execution of high-level tasks, like the ones involved in the aforementioned scenarios, calls for more sophisticated representations closer to the way in which humans interpret and behave within their environments. *Semantic maps* came out to cope with this need, permitting a robot to *understand* not only the spatial aspects of its workspace, but also the meaning of its elements (objects and rooms) and how humans interact with them, *e.g.* functionalities, events, and relations. This is

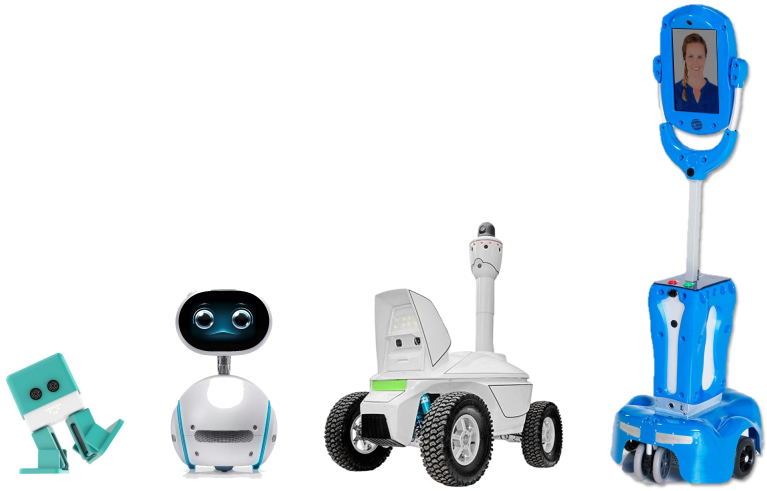


Figure 1.1: Examples of state-of-the-art robots successfully applied to different tasks. From left to right: the educational robot Zowi, the companion and entertainment robot Zenbo, the security patrol robot S5, and the Giraff robot employed in telehealth-care applications.

achieved by considering meta-information, commonly referred to as *common-sense* or *Semantic Knowledge* (SK), concerning the types of elements (and their relations) to be found in the robot workspace. Pieces of this information, typically encoded into a *Knowledge Base* (KB), could be: blankets are often stored in cupboards; lights must be switched off after the working day; teddies make kids happier; fragile crockery should not be cleaned in the dishwasher.

1.1 Motivation

Typically, semantic maps are populated with *crispy information*, *e.g.* an object is a blanket or not. This is due to the weakness of traditional semantic representations to handle uncertainty, which forces the use of recognition algorithms providing a crispy outcome, probably by thresholding a probabilistic result. For example, a recognition algorithm[1] stating that an object can be a blanket with a probability of 0.52, and a carpet with 0.48, might yield a unique outcome by considering the object as a blanket and neglecting the other, high probable, hypothesis. This crispy stance clearly compromises the robot operation: the uncertainty coming from sources like the robot sensory system and the employed models is being disregarded when the recognition results are stored in the semantic map. So, despite the results clamor for a disambiguation, our lovely grandma could end up with a rugged carpet on top of her. Therefore, it becomes clear the necessity of levering probabilistic techniques for populating and

[1]For the sake of simplicity only two possible object types are considered at this point.

maintaining semantic maps, as well as to adapt semantic representations for managing uncertain information, which would permit a mobile robot to operate in a more coherent and efficient way.

As an illustrative example of the convenience of dealing with uncertain information, let's suppose an scenario where a servant robot right landed from the lab into its new home is commanded to bring the slippers to the grandma. In the absence of spatial information, the robot could infer (according to the loaded KB) that the most probable location for slippers is a bedroom. During the preliminary setup, the robot initially recognized a bedroom corresponding to the farthest room from the current grandma location with a probability of 0.45, and 0.43 of being a kitchen[2]. Another room, close to the robot location, has been recognized as a kitchen with a probability of 0.48, and as a bedroom with 0.47. The utilization of only the most probable interpretation, *modus operandi* of traditional, crispy semantic maps, would lead to the exploration of the farthest room having a 45% of being the correct place, while the consideration of both interpretations would produce the more logical plan (for the robot battery and the grandma patience) of taking a look at the closer room first.

Although there exist numerous algorithms for the recognition of objects and/or rooms that provide uncertainty measurements about their results, they usually work by individually processing each spatial element according to its geometric/appearance features. In other words, if the most probable type of an object is blanket, it is considered a blanket no matter other objects placed nearby nor its location. Nevertheless, human-made environments are rich in contextual information worth to exploit, *i.e.* the room's layout follows a certain order, and objects are not placed randomly but following certain configurations according to their functionality: *e.g.* a remote control is usually found close to a tv, a corridor connects rooms, or bathtubs are (as indicated by its name) placed at bathrooms. Modeling and leveraging context is useful, for example, to disambiguate uncertain results: following the previous example, if the object is found into a wardrobe it would be more probably a blanket than a carpet, which are usually lying on the floor. This kind of information can be naturally encoded in KBs, however, its exploitation for contextual object/room recognition, also managing uncertainties, is not straightforward.

Probabilistic Graphical Models (PGMs) have been a widely resorted tool for modeling and exploiting contextual relations, while dealing with uncertainty. They work with a graph-based representation, where nodes stand for random variables and edges link variables showing some type of relation. For example, in the case of the object recognition problem, each object in the scene is represented by a random variable that takes values from the set of possible object types (table, book, couch, etc.), and nodes whose associated objects are close to each other in the scene are linked by an edge. This representation supports the efficient execution of probabilistic inference methods, which permit us to retrieve the scene object recognition results along with a measure of their uncertainty. PGMs have been successfully applied to tasks

[2]Notice that the sum of both probabilities is 0.88. The remaining probabilities, up to a total of 1, correspond to other possible room types: corridor, bedroom, living room. etc.

like image denoising, natural language processing, activity recognition, etc. However, they exhibit a number of limitations that could prevent their utilization for populating semantic maps: they become computationally intractable when the complexity of the problem increases, *i.e.* the number of objects/rooms in the environment and their types augments, they need a considerable amount of training data to tune successful models, and they are unable to detect incoherent results as well as to learn from experience.

1.2 Contributions

This thesis contributes to overcome some of the aforementioned limitations of traditional semantic maps by resorting to probabilistic techniques. Concretely, the goals of the thesis, which resulted in the development of those techniques, were stated as:

- **Development of reliable recognition methods:** To provide contextual object/room recognition algorithms able to exploit contextual relations and handle uncertainty, in close synergy with KBs, also offering a number of desirable features like scalability, efficiency, detection of wrong results, learning from experience, etc.
- **Enhancement of traditional representations to manage uncertainty:** To accommodate the probabilistic outcomes of such algorithms into a novel semantic map representation, in such a way that a robot could have a grounded belief about the certainty of its understanding of the surroundings, hence operating in a coherent fashion.

Thereby, the contributions of this thesis can be grouped into two major topics: contextual scene understanding, and semantic mapping.

1.2.1 Contributions to contextual scene understanding

The first set of contributions, presented in the papers [114, 121, 122, 115, 116, 119, 117] focuses on the *scene object and/or room recognition problems*. To overcome these problems is crucial for the proper building of the semantic representations sought. Probabilistic Graphical Models, concretely Conditional Random Fields (CRF), are used to model those issues from a holistic stance, considering the contextual relations among objects and/or rooms, and to natively deal with uncertainty. Their suitability for the problem at hand has been verified through a comprehensive evaluation of PGMs trained and exploited by the most popular learning and probabilistic inference algorithms.

These CRFs work in synergy with KBs, a mutually beneficial relationship which permits to keep their advantages and mitigate their limitations:

- KBs provide CRFs with the capabilities to: reduce their complexity, exploit prior information, verbalize their outcome, generate an arbitrary number of training samples, detect incoherent results, and learn from experience.

- CRFs enables KBs to handle uncertainty and exploit contextual relations in a holistic and principled manner.

The developed algorithms have been compared with other cutting-edge solutions employing state-of-the-art datasets. Additionally, a dataset consisting of 25 rooms from our facilities, called *UMA-Offices*, has been collected and made public. An open-source library, called *Undirected Probabilistic Graphical Models in C++* (UPGMpp), has been also implemented for working with PGMs paying attention to the special requirements of software targeted at robotic applications.

1.2.2 Contributions to semantic mapping

The goal of the second group of contributions, presented in the papers [123, 118, 120], is to accommodate the probabilistic outcome of the previous techniques into a semantic map representation. For that, the so-called *Multiversal Semantic Map* (*MvSmap*) representation has been developed. This map turns such outcome into different interpretations of the robot workspace, coined universes, which are annotated with their probability of being the true ones. This permits the robot to consider not only the most probable universe, but other ones also showing a high probability, hence unlocking a more coherent and efficient operation. Techniques to keep the number of possible universes tractable in complex environments, crowded of objects and rooms, has been also studied.

The suitability of this map as well as its capacity to efficiently handle uncertain information have been tested with a novel dataset, *Robot@Home*, collected by a mobile robot surveying a number of apartments. The *Object Labeling Toolkit* (OLT), publicly available for the researcher community, has been developed to effortlessly process datasets compounded of sequences of sensory information, such as *Robot@Home*.

1.2.3 Publications

The present thesis encompasses the following publications:

Journals

- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, and Javier Gonzalez-Jimenez.* **Building Multiversal Semantic Maps for Mobile Robot Operation**. Submitted to Knowledge-Based Systems (2016).
- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, and Javier Gonzalez-Jimenez.* **A Survey on Learning Approaches for Undirected Graphical Models. Application to Scene Object Recognition**. In International Journal of Approximate Reasoning, accepted (2016).
- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, and Javier Gonzalez-Jimenez.* **Robot@Home, a Robotic Dataset for Semantic Mapping of Home Environments**. Submitted to International Journal of Robotics Research (2016).

- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, and Javier Gonzalez-Jimenez.* **Scene Object Recognition for Mobile Robots Through Semantic Knowledge and Probabilistic Graphical Models**. In Expert Systems with Applications, vol. 42, no. 22, pp. 8805–8816, (2015).

- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, and Javier Gonzalez-Jimenez.* **Exploiting Semantic Knowledge for Robot Object Recognition**. In Knowledge-Based Systems, vol. 86, pp. 131–142, (2015).

Conference proceedings

- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, and Javier Gonzalez-Jimenez.* **Probability and Common-Sense: Tandem Towards Robust Robotic Object Recognition in Ambient Assisted Living**. In 10th International Conference on Ubiquitous Computing & Ambient Intelligence, Las Palmas de Gran Canaria, Spain, (2016).

- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, and Javier Gonzalez-Jimenez.* **Joint Categorization of Objects and Rooms for Mobile Robots**. In IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, (2015).

- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, and Javier Gonzalez-Jimenez.* **OLT: A Toolkit for Object Labeling Applied to Robotic RGB-D Datasets**. In European Conference on Mobile Robots (ECMR), Lincoln, UK, (2015).

- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, and Javier Gonzalez-Jimenez.* **UPGMpp: a Software Library for Contextual Object Recognition**. In 3rd. Workshop on Recognition and Action for Scene Understanding (REACTS), Valletta, Malta, (2015).

- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, and Javier Gonzalez-Jimenez.* **Mobile Robot Object Recognition through the Synergy of Probabilistic Graphical Models and Semantic Knowledge**. In European Conference on Artificial Intelligence, Workshop on Cognitive Robotics (CogRob), Prague, Czech Republic, (2014).

1.3 Thesis framework

This thesis is the result of 5 years of work by the author as a member of the Machine Perception and intelligent Robotics (MAPIR) research group[3], part of the Department of System Engineering and Automation of the University of Málaga. This research has been mainly funded by the FPU (*Formación de Profesorado Universitario*) grant program, supported by the Spanish Education Ministry.

[3] http://mapir.isa.uma.es/

During this period, the author successfully completed the doctoral program in Mechatronics Engineering, coordinated by the Department of System Engineering and Automation, where he obtained a strong background knowledge concerning the four fundamental pillars of robotics: control systems, electronic systems, mechanical systems, and computers. This academic education was completed with different courses, like the "Writing in the sciences" course imparted by the Stanford University, and with the participation in the First Örebro Winter School on "Artificial Intelligence and Robotics", which aimed to bring closer two fields strongly correlated like Artificial Intelligence and Robotics. This school also made possible to meet other researchers in the same and other related fields.

The author also completed a three months research stay at the Knowledge-Based Systems Research Group[4], of the University of Osnabrück, in 2014, under the supervision of Prof. Dr. Joachim Hertzberg. During this time, research focused on the analysis and implementation of different algorithms for efficiently handling PGMs, as well as in their application to online object recognition in mobile robots. In this great experience, cooperations with researchers of the group were also established.

Besides, it is also worth to mention that the author has been active in the review process of papers/articles from prestigious conferences and journals, like in the case of the International Conference on Robotics and Automation (ICRA, 2014, 2015, 2016), the International Conference on Intelligent Robots and Systems (IROS, 2015), or the Association for the Advancement of Artificial Intelligence and the Intelligent Service Robotics journals.

The FPU grant also offered the opportunity to collaborate as an assistant lecturer with the Department of System Engineering and Automation. Concretely, the author taught on 'Robotics" at the faculty of Computer Science, in the University of Málaga. He also co-supervised the bachelor thesis of a student, David Zúñiga Noël, entitled "Visual SLAM with RGB-D Cameras Based on Pose Graph Optimization".

In addition to the research concerning this thesis, the author has been also involved in other projects within the MAPIR group, some of them with related topics:

- **TCS: Tunnel Continuous Setout** (Nov'08 – Jul'11): this project focuses on the development of a system for the automatic setting-out of tunnel sections to be perforated. The system prototype, which takes the same name as the project, combines a scanning device that surveys the excavation front and a laser projector that continuously displays the actual tunnel section. The most challenging part of the project was the implementation of calibration techniques for retrieving the accurate location of all the system components.

- **ExCITE: Enabling SoCial Interaction Through Embodiment** (Jul'10 – Jun'13): The author's role in this project was related to the development of technical improvements for the Giraff telepresence platform: a safer and easier driving, including auto-docking to the recharging station, obstacle detection, and displaying the robot position in a sketch map of the visited place. A robotic

[4] www.inf.uos.de/kbs/

architecture called Navigation Assistant (NAS) was also implemented to fulfill these particular needs.

- **Taroth: New developments toward a Robot at Home** (Jan'12 – Dec'15): this project pursuits the three following targets: 1) improving dependability of the robot motion, 2) integrating and exploiting semantics to improve robot autonomy and interaction with humans, and 3) developing a robot software architecture that can manage Ambient Assisted Living services related to entertainment, domotics, social networking, safety, etc.

- **IRO: Improvement of the sensorial and autonomous capability of Robots through Olfaction** (Jan'14 – Feb'19): the research in this project is targeted at the investigation of mechanisms to use odor information in problems such as object recognition and scene-activity understanding, paying special attention to the role of semantics within the robot perception and decision-making processes, aiming to improve the robot capabilities in terms of efficiency, autonomy and usefulness.

From the author's work in these projects arose a number of additional publications:

Journals

- *Javier Gonzalez-Jimenez, Vicente Arévalo, Cipriano Galindo, and Jose-Raul Ruiz-Sarmiento*. **An Automated Surveying and Marking System for Continuous Setting-out of Tunnels**. In Computer-Aided Civil and Infrastructure Engineering, vol. 31, no. 3, pp. 219–228, (2016).

Conference proceedings

- *David Zuñiga-Noël, Jose-Raul Ruiz-Sarmiento, and Javier Gonzalez-Jimenez*. **Detección de Lugares con Cámaras RGB-D. Aplicación a Cierre de Bucles en SLAM**. In XXXVII Jornadas de Automática, Madrid, Spain, (2016).

- *Javier Gonzalez-Jimenez, Jose-Raul Ruiz-Sarmiento, and Cipriano Galindo*. **Improving 2D Reactive Navigators with Kinect**. In 10th International Conference on Informatics in Control, Automation and Robotics (ICINCO), Reykjavic, (Iceland, 2013).

- *Javier Gonzalez-Jimenez, Cipriano Galindo, Francisco Melendez-Fernandez, and Jose-Raul Ruiz-Sarmiento*. **Building and Exploiting Maps in a Telepresence Robotic Application**. In 10th International Conference on Informatics in Control, Automation and Robotics (ICINCO), Reykjavic, Iceland, (2013).

- *Javier Gonzalez-Jimenez, Cipriano Galindo, and Jose-Raul Ruiz-Sarmiento*. **Technical Improvements of the Giraff Telepresence Robot Based on Users' Evaluation**. In The 21st IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), Paris, France, (2012).

- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, and Javier Gonzalez-Jimenez.* **Cámaras basadas en tiempo de vuelo. Uso en la mejora de métodos de detección de caras.** In XXXII Jornadas de Automática, Sevilla, Spain, (2011).

Technical reports

- *Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, and Javier Gonzalez-Jimenez.* **Experimental Study of the Performance of the Kinect Range Camera for Mobile Robotics**. Universidad de Malaga, Andalucia Tech, Departamento de Ingenieria de Sistemas y Automatica, (2013).

1.4 Thesis outline

Besides the introductory chapter, the remaining ones in the first part of this thesis, **Part I: Thesis description**, are organized as follows:

Chapter 2: Theoretical background gives brief notions of the theory behind two frameworks constantly resorted in this thesis: Probabilistic Graphical Models and Knowledge Base representations, so the non-expert readers in this field can get the basic background for a proper understanding of the next chapters. The author has tried his best to make the reading of this chapter as pleasant as possible.

Chapter 3: Contextual scene understanding describes the traditional approaches followed for the recognition of objects and rooms by a mobile robot, and how they are related to the presented contributions exploiting contextual information. Details about the synergy of PGMs and Semantic Knowledge for scene understanding are provided. This chapter also discusses the datasets used to test the developed techniques, including the *UMA-Offices* one, as well as the implemented software in this respect: the *Undirected Probabilistic Graphical Models in C++* library.

Chapter 4: Semantic Mapping outlines the semantic map representations traditionally used in mobile robotics, and describes the thesis contribution for a representation handling uncertain information: the *Multiversal Semantic Map*. The virtues of this map have been checked against a novel dataset, *Robot@Home*, whose features are described in this chapter along with those of the software used for its processing: the *Object Labeling Toolkit*.

Chapter 5: Summary of included papers lists the papers that make up the second part of the thesis, **Part II: Included papers**, giving a brief description of their content and contributions.

Chapter 6: Conclusions and future work discusses the conclusions drawn from the work done in this thesis, as well as the research lines still open an possible extensions.

2 Theoretical background

This chapter briefly covers the theory behind two frameworks that have been essential for the research in this thesis. The first one is Probabilistic Graphical Models, used to holistically model the object and/or room recognition problems from a probabilistic stance. The second framework is Knowledge Bases, employed to encode Semantic Knowledge of the domain at hand for its posterior exploitation with different purposes. The synergy between both frameworks enables the design of sophisticated techniques to manage semantic maps.

2.1 Probabilistic Graphical Models

Probabilistic Graphical Models (PGMs) [65, 12] suppose a widespread framework from the Machine Learning field to efficiently model and exploit contextual relations, aiming to predict multiple, somehow dependent, random variables. These models are usually employed to deal with complex systems that involve uncertainty, which mainly arises from the limitations on the motion and sensory systems of the robot.

PGMs rely on a graph representation $G = (V, E)$, where the set V represents the random variables of the problem as nodes, while the edges $E \subseteq V \times V$ relate variables that are dependent in some way. This graph-based representation permit PGMs to compactly encode complex distributions over high-dimensional spaces, and to support the execution of probabilistic inference techniques for the prediction of the variable values. Thus, PGMs are strongly based on principles from graph theory and probability theory.

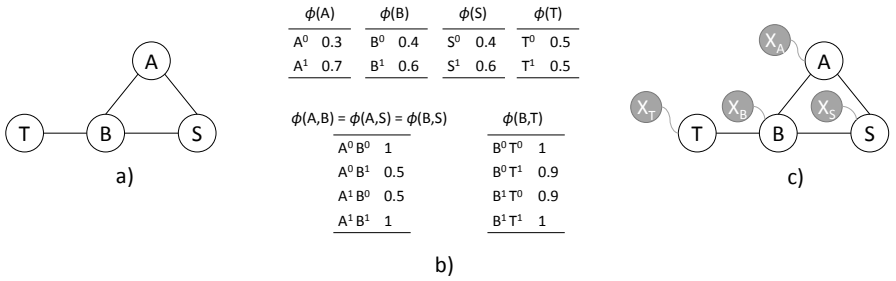


Figure 2.1: a) Graph representation of the MRF *happiness* model. b) Factors defined over such a graph. c) CRF representation including measures about different aspects.

PGMs have been successfully applied to a variety of domains like medicine, computer vision, robotics, etc. Depending on the types of edges, PGMs can be grouped on Directed or Undirected models. On the one hand, Directed Graphical Models, also called Bayesian Networks (BNs) [98], model the dependencies among nodes through directed edges, encoding *causality* relations. These models have been utilized with notable success in problems like medical diagnosis [84], biology [159], weather forecasting [1], or robotic localization and map building [14]. On the other hand, Undirected Graphical Models (UGMs), also called Markov Random Fields (MRFs) [63], employ undirected edges to define *symmetric* relations among random variables. This approach has reached a remarkable success in computer vision [50].

The choice between BNs and MRFs largely depends on the target application, since they are able to encode different types of dependencies (*e.g.* BNs can define induced dependencies, while MRFs are able to represent cyclic dependencies). In the case of the object/room recognition problem, the more suitable framework is such of MRFs, since the nature of the relations among objects and rooms is symmetric, and they can also exhibit loops, which are non trivial to model within the BNs framework. In its turn, the discriminative variant of MRFs, called Conditional Random Fields (CRFs) [70], are more appropriate in classification problems where the random variables are conditioned to observed data [59, 69]. The next section shows an example to illustrate the differences among these models.

2.1.1 The happiness example

Let's suppose the family formed by **B**ob, **S**ophie, and **A**lice presented in the introductory chapter, and a mobile robot with the goal of modeling their *happiness state* through a MRF. As human beings, we empathize with each other, and we are directly affected and affect the well-being and emotional state of our relatives, so it makes sense to take into account these relationships when trying to predict the happiness state of a person. PGMs model this in a principled way. Figure 2.1-a) shows the graph representation exemplifying the relations among the happiness state of each

family member, also including our lovely grandma, called **T**ess, who have nice conversations with Bob in the elevator. From this representation it can be inferred that the happiness of Alice, Bob, or Sophie directly influences the feelings of the other family members, while Tess has only influence and is influenced by Bob.

At this point, instead of modeling the whole probability distribution $P(\mathbf{y})$ (with $\mathbf{y} = [A, B, S, T]$), MRFs break it down into smaller pieces through the utilization of factors, *i.e.* functions defined over different parts of the graph. The first row of Figure 2.1-b) shows factors defined over the nodes of the graph, which are commonly called *unary factors*, stating the likelihood of these nodes to take certain values. Let's simplify the happiness of a person to two possible states, unhappy (0) and happy (1). Having a closer look at these factors, we can see for example that Alice is more probable to be happy than Tess. In its turn, the second row shows factors defined over pair of nodes, called *pairwise factors*, that set the likelihood about those nodes taking a certain values combination. The defined factors tell us that Bob, Alice and Sophie are prone to share their happiness, and although Bob and Tess are also inclined to have the same state, this influence is weaker. The values defined in a factor have not to sum up 1, since they are not probabilities.

Exhaustively defining $P(\mathbf{y})$ in this toy example requires the codification of $2^4 = 16$ probabilities. In this case, the MRF codification through factors does not save so much work, however, in more realistic scenarios with dozens, hundreds or thousands of random variables their utilization becomes crucial to keep the problem tractable. For example, a scenario with 20 binary random variables entails the definition of $2^{20} \simeq 10^6$ probabilities.

Thus, according to the Hammersley-Clifford theorem [48], the probability $P(\mathbf{y})$ can be factorized over the graph G as a product of factors $\phi(\cdot)$:

$$p(\mathbf{y}) = \frac{1}{Z} \prod_{c \in C} \phi(y_c) \tag{2.1}$$

where C is the set of maximal cliques[1] of the graph G, and $Z(\cdot)$ is the so-called partition function that plays a normalization role so $\sum_{\xi(\mathbf{y})} p(\mathbf{y}) = 1$, being $\xi(\mathbf{y})$ a possible assignment to the variables in $\mathbf{y}$. Therefore, the computation of the partition function is needed for computing the probability of a given assignment.

This way to define factors is rigid and naive: the happiness of a person can hardly be modeled by writing in stone his tendency to be happy, and it is additionally influenced by a number of (hopefully measurable) daily aspects: the sleeping hours, the success at work, hours spent with family and friends, etc. These aspects could be also included in the MRF graph as additional random variables, although the modeling of their probabilities and relations tend to be needlessly complex. Conditional Random Fields (CRF) [70] avoid the need to model them by conditioning the probability distribution over $\mathbf{y}$ to the values of these aspects, referred to as *features*. Thus, a CRF

[1] A maximal clique is a fully-connected subgraph that can not be enlarged by including an adjacent node.

works directly with the distribution $p(\mathbf{y} \mid \mathbf{x})$, where $\mathbf{x}$ is the vector of observed features. Figure 2.1-c) shows the graph representation of a CRF considering this information. Additionally, instead of defining by hand the factors for each possible content of $\mathbf{x}$, they are parametrized through a vector of weights θ that are learned during the training phase of the CRF. Thus, the probability $p(\mathbf{y} \mid \mathbf{x})$ can be retrieved by:

$$p(\mathbf{y} \mid \mathbf{x}; \theta) = \frac{1}{Z(\mathbf{x}, \theta)} \prod_{c \in C} \phi(y_c, x_c, \theta_c) \tag{2.2}$$

The parametrized factors can be formulated in different ways depending on the application. For example, in recognition problems, unary factors are often defined as $\phi_u(y_i, x_i, \theta) = \sum_{l \in L} \delta(y_i = l)\theta_l f(x_i)$, where $f(x_i)$ computes a vector of features that characterizes the object x_i (*e.g.* size, shape, color, etc.), θ_l is the vector of weights for the class l obtained during the training phase, and $\delta(y_i = l)$ is the Kronecker delta function, which takes value 1 when $y_i = l$ and 0 otherwise. Pairwise factors are defined in a similar way, but considering a function that computes a vector of contextual features (*e.g.* difference of color, difference of orientation, etc.).

2.1.2 Learning the models

Training a CRF model for a given domain requires estimating the parameters θ, in such a way that they maximize the likelihood in Eq.2.2 with respect to a certain i.i.d. training dataset $D = [d^1, \ldots d^m]$, that is:

$$\max_{\theta} L_p(\theta : D) = \max_{\theta} \prod_{i=1}^{m} p(\mathbf{y}^i \mid \mathbf{x}^i; \theta) \tag{2.3}$$

where each training sample $d^i = (\mathbf{y}^i, \mathbf{x}^i)$ consists of a number of observed features from the elements of the problem at hand ($\mathbf{x}^i$), the people whose happiness is to be estimated in our example, and the corresponding ground truth information about their classification ($\mathbf{y}^i$), *i.e.* if they are happy (1) or not (0).

The optimization in Eq.2.3 is also known as Maximum Likelihood Estimation (MLE), and requires the computation of the partition function $Z(\cdot)$, which in practice is *NP*-hard, hence an intractable problem. Two major approaches stand out to overcome this concern: (i) the definition of alternative, tractable objective functions, or (ii) the estimation of the likelihood by approximate inference algorithms [68, 66, 96]. The performance of methods from both options highly differs depending on the domain of the problem at hand, *i.e.* the nature and internal structure of the data to work with. Therefore, for a certain application, a thorough study is needed in order to obtain a successful model, which motivates the analysis described in Chapter 3.

2.1.3 Probabilistic inference

Once a CRF is trained, and its graph representation modeling a given problem is built, it can be exploited by probabilistic inference methods to perform different probability

queries. At this point, two types of queries are specially relevant: the *Maximum a Posteriori* (MAP) query, and the *Marginal* query. The goal of the MAP query is to find the most probable assignment $\hat{\mathbf{y}}$ to the variables in $\mathbf{y}$, *i.e.* :

$$\hat{y} = arg\max_{\mathbf{y}} p(\mathbf{y} \mid \mathbf{x}; \theta) \tag{2.4}$$

Once again, the computation of the partition function $Z(\cdot)$ is needed, but since given a certain CRF graph its value remains constant, this expression can be simplified by:

$$\hat{y} = arg\max_{\mathbf{y}} \prod_{c \in C} \exp(\langle \phi(x_c, y_c), \theta \rangle) \tag{2.5}$$

Nevertheless, this task checks every possible assignment to the variables in $\mathbf{y}$, so it is still unfeasible for real applications. An usual way to address this issue is the utilization of approximate methods, like the *max-product* version of Loopy Belief Propagation (LBP) [150], Iterated Conditional Models (ICM) [11], or Graph Cuts [15].

On the other hand, the Marginal query, which can be performed by, for example, the *sum-product* version of LBP [155], provides us beliefs about the possible assignations to the variables $\mathbf{y}$. In other words, this query yields the marginal probabilities for each element taking different values, as well as the compatibility of these assignments with respect to the values of contextually related elements. Notice that the most probable MAP assignment to a random variable can differ from the highest marginal probability. Additionally, with this query is also possible to estimate the probability of a certain assignment to the variables in $\mathbf{y}$.

2.2 Knowledge bases

Knowledge base (KBs) is the term used in Artificial intelligence (AI) to describe one of the two parts of a knowledge-based system, which is in charge of encoding semantic or common-sense knowledge about a particular domain in a computer-readable fashion. The other system part is a reasoning engine able to infer new information or detect inconsistencies in the KB. In the happiness example, a KB could encode the types of relations among persons, the different factors that affect their happiness, etc. (see Section 2.2.2), which are typically modeled through Ontologies. Knowledge-based systems have been a pivotal component for semantic mapping, as they permit a mobile robot to perform efficiently according to the information collected from the environment.

2.2.1 Ontologies

An Ontology is commonly defined as a representation of a conceptualization related to a knowledge domain, which accounts for a number of concepts arranged hierarchically, relations among them, and instances of such concepts, also called individuals [144]. Example of concepts could be `Person` or `Happiness`, while `Person`

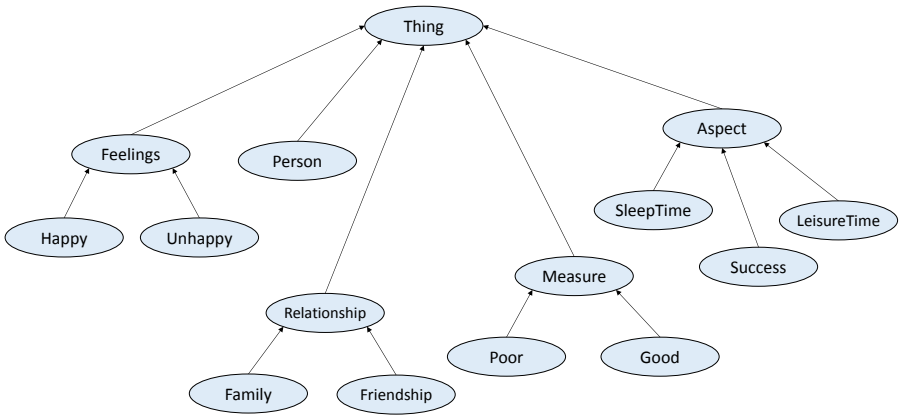


Figure 2.2: Hierarchy of concepts for the *happiness* domain.

`hasState Happiness` could be a relation stating the happiness of a person. Thus, the happiness of `Bob`, an individual of the concept `Person`, could be codified by `Bob hasState Happy`.

The process of obtaining and codifying Semantic Knowledge can be tackled in different ways. For example, web mining knowledge acquisition systems can be used as mechanisms to obtain information about the domain of discourse [158]. Available common-sense Knowledge Bases, like ConceptNet [134] or Open Mind Indoor Common Sense [46], can be also analyzed to retrieve this information. Another valuable option is the utilization of internet search engines, like Google's image search [29], or image repositories like Flickr [99], for extracting knowledge from user-uploaded information. Semantic Knowledge can be also codified through an human elicitation process, which supposes a truly and effortless encoding of a large number of concepts and relations between them. In contrast to online search or web mining-engine based methodologies, this source of semantic information (a person or a group of people) is trustworthy, so the uncertain about the validity of the information is reduced [119].

2.2.2 Happiness from an Ontological stance

Figure 2.2 shows an example of hierarchy of concepts from an Ontology modeling the *happiness* domain. The root concept is `Thing`, with 5 children codifying information about: the possible states of happiness, the person concept itself, different types of relationships among people, possible aspects that affect happiness, and measurements of those aspects. Using this Ontology, one can define, for example, that a happy person has a `Good SleepTime`, `Success` at work, and `LeisureTime`. Thus, if a `Person` shows these properties, a logical reasoner, like Pellet [133], FaCT++ [143], or Racer [47], can be used to automatically infer that such a person is happy.

Contextual relations among concepts or instances can be also defined. For example, `Bob hasFamilyRealtion Alice` sets that Bob and Alice are relatives. This way of inferring crispy information and defining crispy relations and properties, although useful in some domains, has limitations. The major one is the lack of mechanisms to manage uncertainty or providing beliefs about the inference results, which prevent its application to problems where their consideration is a must.

3 Contextual scene understanding

This section deals with the developed techniques for contextually recognizing objects and rooms. After an introduction, it discusses the related work that can be found in the literature, describes the datasets used as a testbed to evaluate such techniques, and concludes with the description of the contributions done in this regard.

3.1 Introduction

The ability to be aware of the objects and rooms in the robot surroundings, as well as of their types, is vital for a successful robot operation. Object/room recognition techniques are core components of semantic mapping systems, which are in charge of yielding the type of the spatial elements captured by the robot sensory system. As a consequence of this, a number of recognition approaches have been proposed for populating semantic maps.

Recognition methods often rely on RGB, and more recently on RGB-D information to perceive the robot environment and process the spatial elements therein. For that, the captured images are segmented into such spatial elements, which are individually processed in order to retrieve their type, *e.g.* counter, cabinet, microwave, kitchen, bathroom, etc., through a number of appearance and/or geometric features. The utilization of RGB and depth information entail a number of challenges as changing lighting conditions, cluttered room layouts, occlusions, or changing viewpoints, which can produce ambiguous recognition results. Recognition techniques also face other sources of uncertainty, like those coming from the own sensory system (*e.g.* sensor noise) or from the defined models. Given the effect that ambiguous recognition results stored in a semantic map may have on the robot operation (recall the lovely

grandma and the carpet), recognition techniques integrated into these systems have to tackle them.

In the works that conform this thesis a number of recognition techniques that address this uncertainty issues have been proposed, also striving to decrease the ambiguity of the recognition results by exploiting contextual relations. PGMs are employed for that, in close cooperation with KBs in the form of Ontologies in order to enhance their performance. These techniques are also able to provide a measure about the uncertainty of their results, which is crucial for the semantic mapping framework presented in the next chapter.

3.2 Related work

A vast literature exists around the recognition of objects and/or rooms. This section starts by briefly discussing traditional approaches addressing this issue, and the good reasons for contextually modeling these problems. Then, popular works exploiting context through PGMs are presented, as well as some alternatives exploring the utilization of Semantic Knowledge. Finally, the datasets applicable to the evaluation of the proposed recognition techniques are reviewed, as well as related software applications.

Traditional scene object/room recognition

Scene object recognition is a widely studied topic in computer vision and robotics. Recognition systems have traditionally relied on the features of the objects/room like their geometry or appearance due to their acceptable performance. Regarding object recognition, a popular example is the work by Viola and Jones [146], where an integral image representation is used to encode the appearance of a certain object category, and is exploited by a cascade classifier over a sliding window to detect the occurrences of such object type in intensity images. Another well known approach is the utilization of image descriptors, like Scale-Invariant Feature Transform (SIFT) [74], Speeded-Up Robust Features (SURF) [64], or Local Binary Pattern (LBP) [20], to capture the appearance of objects, and its posterior exploitation by classifiers like Supported Vector Machines (SVMs) [100] or Bag-of-Words [85, 52]. Other works study the automatic learning of low level features, *e.g.* using neuronal networks, as is the case of Bai *et al.* [8]. The work by Zhang *et al.* [157] provides a comprehensive review of methods following this approach.

On the other hand, a considerable number of works also tackle the room categorization problem through the exploitation of their geometry or appearance, like the one by Mozos *et al.* [80] which employs range data to classify spaces according to a set of geometric features. Also popular are works resorting to global descriptors of intensity images, like the *gist* of the scene proposed by Oliva and Torralba [91], those resorting to local descriptors like the aforementioned SIFT and SURF [6, 81], or the works combining both types of cues, global and local, pursuing a more robust performance [149, 101].

Despite the success of local recognition systems for certain applications, their integration into mobile robots arises a number of additional issues to be tackled [119, 93]. One of the most significant ones is the fact that they can lead to ambiguous recognitions, i.e. they are prone to fail in identifying classes with similar features, as analyzed in [92, 19, 38, 115]. This is mainly due to only relying on features of the objects/rooms themselves, disregarding valuable contextual information that is also available. Therefore, a significant, growing body of current research aiming to overcome this issue is considering contextual information of the scene objects in addition to their usually employed individual features. Some works have attempted to exploit this information by providing ad-hoc or preliminary solutions, like in [78], where the co-occurrence of objects appearing in distinct types of rooms are implicitly modeled. However, these works lack a consistent theoretical background, compromising, among others, their comparison, generalization, re-usability, or scalability. Moreover, their output consists of a set of objects' labels, which do not carry any semantic information profitable by high-level AI robotic components. Well grounded alternatives for modeling/exploiting contextual relations are *Probabilistic Graphical Models* and *Semantic Knowledge*, whose combination is exploited in this thesis with the goal of mitigating their drawbacks and boosting their virtues.

Contextual Recognition through PGMs

Probabilistic Graphical Models (PGMs) in general, and Undirected Graphical Models (UGMs) in particular, have became popular frameworks to model and exploit contextual relations in combination with probabilistic inference methods [65]. Contextual relations can be of different nature, involving objects and/or rooms. On the one hand, objects are not placed randomly within the robot workspace, but following configurations that make sense from a human point of view, *e.g.* carpets are on the floor, remote controls can be found close to televisions, and pillows are normally placed on beds. The earliest works using this information were based on intensity information of the scene, like [152], where the context between pixels in a given RGB image is modeled by a discriminative Conditional Random Field (CRF). Another work, also relying on intensity images, is the presented in [106] that proposes a CRF framework that incorporates hidden variables for part-based object recognition. The work in [79] also builds part-based models of objects, and represents their interrelations with a PGM. More recent is the work presented in [33] which employs stereo intensity images in a CRF formulation. Three-dimensional information from stereo enables the exploitation of meaningful geometric properties of objects and relations. However, stereo systems are unable to perform on surfaces/objects showing an uniform intensity, which can negatively affect the recognition performance.

With the emergence of inexpensive 3D sensors, like Kinect, a new batch of approaches have appeared leveraging the dense and relatively accurate data provided by these devices. For example, the work presented in [4] builds a model isomorphic to a Markov Random Field (MRF) according to the segmented regions from a scene point cloud and their relations. The authors did the tedious work of gathering information

from 24 office and 28 home environments, and manually labeled the different object classes. Interestingly, it is shown in [111] that the accuracy of a MRF in charge of assigning object classes to a set of superpixels increases as the amount of available training data augments. In [145] a meshed representation of the scene is built on the basis of a number of depth estimates, and a CRF is defined to classify mesh faces. CRFs are also used in [60] and [154], where Decision Tree Fields [87] and Regression Tree Fields [56] are studied as a source of potentials for the PGM. The CRF structure for representing the scenes in [154] is similar but less expressive than the one presented here. In that work, a CRF is used to classify the main components of a facility, namely clutters, walls, floors and ceilings.

On the other hand, object–room relations also supposes a useful source of information: objects are located in rooms according to their functionality, so the presence of an object of a certain type is a hint for the categorization of the room and, likewise, the category of a room is a good indicator of the object categories that can be found therein. Thus, recent works have explored the joint categorization of objects and rooms leveraging both, object–object and object–room contextual relations. CRFs have proven to be a suitable choice for modeling this holistic approach, as it has been shown in the works by Rogers and Christensen [113] or Lin *et al.* [73].

Despite their virtues, PGMs shows a number of drawbacks, like the necessity of large and comprehensive datasets for training, their high complexity when modeling real world problems, or their inability to detect incoherent results and learn from experience. The contributions in this section aim to mitigate those issues with the utilization of Semantic Knowledge.

Semantic Knowledge for modeling context

A different trend in the literature resorts to Semantic Knowledge for both recognizing objects and exploiting their contextual information. For example, the work described in Günter *et al.* [45] codifies contextual information in an Ontology, combined with a set of rules defined with the Semantic Web Rule Language [53], to generate objects' candidate classes. These hypotheses are subsequently validated through a matching process with CAD models. Another example is presented in Nüchter and Hertzberg [88], which defines a constraint network in Prolog to classify the main structural surfaces of buildings, i.e. walls, floors, ceiling and doors, using contextual relations like orthogonal, parallel, above, etc. In Galindo *et al.* [35], data codified into an Ontology about scene objects and their relations are used to infer new high-level information. The work introduced by Durand *et al.* [21] recognizes segmented regions that have been previously characterized through a set of features in RGB images. These features are defined in an Ontology, and their usual values for the different object types are learned by symbolic supervised machine learning tools. In this case, a specific procedure matches characterized regions with semantically defined concepts, but although the authors propose the use of contextual relations, they are neither defined nor exploited. An Ontology is also used in Maillot *et al.* [25] for the recognition of isolated objects and their subparts, which manually establishes the as-

sociation between geometric features and numeric values. This Ontology is populated through machine learning techniques like Perceptrons and Support Vector Machines.

A common characteristic of these approaches based on Semantic Knowledge is that they show limitations in quantifying the uncertainty of their results, and in exploiting the encoded contextual relations. The presented contributions face these issues through collaboration with a CRF, which provides the mobile robot with a recognition system endowed with a probabilistic inference mechanism, able to manage uncertainty and adequately exploit contextual relations.

Related software applications

Most contextual-based object recognition works rely on an ad-hoc implementations of both the PGMs framework and inference algorithms [4, 111, 145, 154]. This makes it difficult to conduct a fair comparison between state-of-the-art works, even when they report results resorting to the same dataset. There are some publicly available software libraries implementing this framework [89, 129], but they are not suited for the contextual object recognition problem (e.g. they only handle *chain-structured* models), or their applicability to this issue is limited. Regarding Semantic Knowledge related applications, there exist a number of mature software for codifying and managing this information in Ontologies, as is the case of Protégè [43] or Fluent Editor [17], as well as logical reasoners like Pellet [133], HermiT [41], FaCT++ [143], or Racer [47].

Applicable RGB-D datasets

The irruption of proposals exploiting RGB-D information has been accompanied with public datasets that offer common benchmarking resources for comparing these works. Among them we can find Berkely-3D [57], Cornell-RGBD [5], NYUv1 [130], NYUv2 [131], TUW [3], SUN3D [153], or ViDRILO [75]. Specially popular are Cornell-RGBD, which is employed in several works aforementioned [4, 60, 54], and NYUv2 used in [151, 119, 116, 117]. The next section reports the datasets employed in this thesis.

3.3 Testbed

Three datasets containing RGB-D information have been used to assess the performance of the contributions in this chapter: UMA-Offices [119], NYUv2 [131] and Cornell-RGBD [5]. This section briefly describe the last two datasets, while details about UMA-Offices are provided in Section 3.4.1.

NYUv2 contains a total of 1,449 labeled pairs of both intensity and depth images, and has been extensively used in the literature (*e.g.* [151, 119, 116, 117]) due to its challenging, cluttered scenes from commercial and residential buildings. Although the number and type of objects and rooms we have considered differs from one work to other, typically 208 scenes corresponding to home facilities have been employed,

as well as 24 object categories appearing in such environments, e.g: bottle, cabinet, counter, faucet, floor, mirror, sink, toilet, towel, table, sofa, book, etc. It is worth to mention that the provided images only capture a portion of the scene, so the contained contextual relations are somehow limited. An evidence of this is given by the total number of extracted relations, $1,345$, when compared with the number of objects, 1,295. This is an average of 6.25 objects and 6.47 relations per scene.

The Cornell-RGBD repository has 24 labeled office scenes and 28 home labeled scenes built from the registration of RGB-D images. As opposed to NYUv2, the provided data inspect a larger portion of the scene, resulting in a richer set of available contextual information. This feature has motivated its utilization in a variety of works (e.g [4, 60, 54]). As before, the home scenes have been selected, which sum up a total of 764 object instances and 2,911 contextual relations among them, averaging 27.29 objects and 103.96 relations per scene. We have used the same 17 categories as in the work that presented this dataset [4].

3.4 Contributions

This section describes the developed techniques for an object/room recognition framework through the synergy of PGMs and Semantic Knowledge. It starts with the description of the UMA-Offices dataset, specially collected for testing such techniques, and continues with an overview of the Undirected Probabilistic Graphical Models in C++ library, implemented for efficiently handling PGMs in robotic applications, as well as an analysis of PGM learning strategies. Then, a brief review of those techniques is provided, along with references to papers and online resources with further information.

3.4.1 UMA-Offices dataset

Office facilities are one of the typical application domains for mobile robots. To test the developed techniques in such environments, the UMA-Offices dataset, compounded of 25 office scenes from the University of Málaga, has been collected. Sensory data included in this dataset was acquired by Rhodon, a mobile robot endowed with an RGB-D device mounted on a Pan-Tilt unit, which permits it to perceive the world from a human-like point of view (see Figure 3.1-left). In this repository, the plane-based mapping algorithm by Fernandez-Moral *et al.* [31] was used to build a 3D representation of the scenes (see Figure 3.1-right), as well as to extract planar patches characterized through a number of features (*e.g.* size, orientation, position or contextual relations). In total, 170 object instances were labeled from the following categories: floor, wall, table top, table side, chair back rest, chair seat, and computer screen. Table 3.1 lists the features of this and the other two datasets used as testbeds.

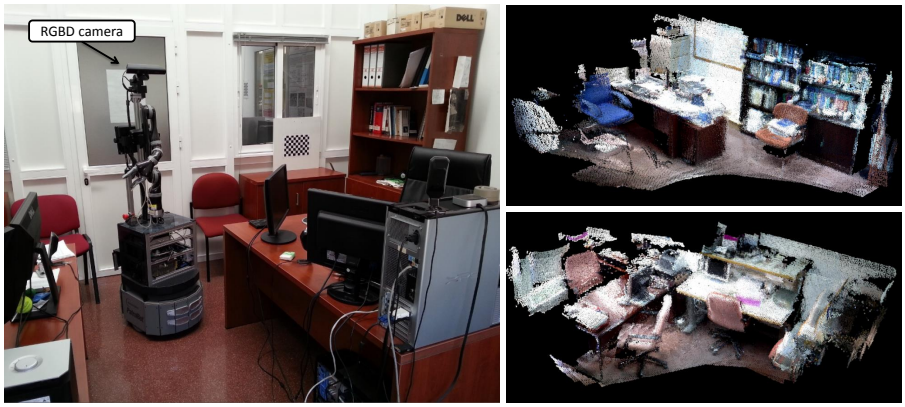


Figure 3.1: Left Rhodon robot from the MAPIR Group capturing RGB-D images from an office. Right, two point clouds from the UMA-Offices dataset.

Table 3.1: Principal characteristics of the three discussed datasets, UMA-Offices, NYUv2 and Cornell-RGBD.

Properties Dataset	UMA-Offices	NYUv2	Cornell-RGBD
#scenes	25	208	24
#obj. categories	7	24	17
#objects	170	1,345	764
#relations	305	1,295	2,911
mean #objects	6.8	6.25	27.29
mean #relations	12.2	6.47	103.96
type of objects	planar surfaces	arbitrary shapes	arbitrary shapes

3.4.2 The UPGMpp library

The study of the software used by state-of-the-art recognition methods employing CRFs arose the lack of public solutions especially focused and optimized for that goal. The utilization of efficient software is a must, since the computational resources in typical robotic platforms are limited given the different modules of the robotic architecture (navigation, localization, etc.) that compete for them.

For that reason, the Undirected Probabilistic Graphical Models in C++ library (UPGMpp, see Figure 3.2) has been developed as open-source[1] for the efficient building, training and managing of undirected PGMs. Its main features are:

- It works with discrete random variables.
- Handles first order (local or unary) and second order (pairwise) relations.

[1] http://mapir.isa.uma.es/work/upgmpp-library

Figure 3.2: UPGMpp logo.

- Nodes (random variables) of different types can appear and interact in the same PGM (for example, nodes representing objects, rooms, facilities, etc.).
- If the value of a random variable is known, such an evidence can be considered.
- It supports PGMs with an arbitrary structure (including graphs with loops).

UPGMpp is fully implemented in C++, and resorts to the also open-source project libLBFGS [82] for performing numerical optimization, and to the Eigen library [44] for fast matrix operations. Boost library [128] is used to avoid unnecessary re-copy of data across the implemented methods by means of shared smart pointers. This library is also employed for serialization purposes, which adds the possibility of storing/loading graphs from/to files, enabling the long-term life of PGMs beyond execution time. Additionally, the Open Multi-Processing API (OpenMP) [94] was employed to speed-up the execution of a number of algorithms through parallelization techniques. Further implementation information and other details can be found in the work by Ruiz-Sarmiento *et al.* [115], which is included in this thesis.

The methods currently available for managing Undirected PGMs are:

Maximum a Posteriori (MAP) inference: Iterated Conditional Modes (ICM) [11], Greedy ICM, Exact Inference, Loopy Belief Propagation (LBP) [150], Tree Reparametrization Belief Propagation (TRBP) [148], Residual Belief Propagation (RBP) [24], α-expansions and α-β Swaps Graph Cuts [15].

Marginal inference: (sum-prduct) Loopy Belief Propagation [155], Tree Reparametrization Belief Propagation [148], Residual Belief Propagation [24].

Learning objective functions: Pseudo-likelihood [11], Score-matching [55], Piecewise-likelihood [136, 135], Marginal-based approximation [68], MAP-based approximation [65].

Learning optimization methods: Stochastic Gradient Descent (SGD) [83], quasi-Newton Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) [86].

As a proof of the efficiency achieved by the library, and as reported in [115], different inference methods were executed on scenes from the NYUv2 dataset, which averages 6.25 objects and 6.47 contextual relations per scenario (see Table 3.1), reaching the ICM inference method a mean execution time of 0.46ms, the LBP one 2.16ms, and the α-expansions method 7.78ms.

3.4.3 Testing CRF learning approaches

The learning and probabilistic inference methods implemented in UPGMpp have been successfully applied to a variety of problems, however, their performance highly depends on the peculiarities of the application domain [68, 66, 96, 32]. A study of this for the scene object recognition result was missing in the literature, so this gap was covered through an empirical analysis of the most popular strategies. Concretely, two families of objective functions have been explored: pseudo-likelihood, and approximate inference algorithms, including Marginal and Maximum a Posteriori methods: sum-product and max-product LBP, ICM, and Graph-cuts. Two approaches for the optimization of such objectives are also considered: SGD, and L-BFGS.

As a testbed for the conducted analysis the indoor home scenes from the NYUv2 and Cornell-RGBD were employed, with particular features worth to explore: while NYUv2 comprises a high number of labeled images (we have used 208 from home environments) that capture the objects and relations from portions of scenes, Cornell-RGBD provides a lower number of scenes (28 from homes) but fully covering the inspected place, similarly to the contributed UMA-Offices dataset, which results in a considerably larger number of perceived objects and relations.

The conducted study focused on two facets of the learning methods: the recognition performance of the trained CRFs, and the required computational time. To measure the CRFs performance different MAP inference methods were executed over the learned models, and their recognition results compared with the ground-truth information provided by the datasets. The computational time needed by each learning method to converge was also analyzed, studying the advantage of parallelization techniques. Finally, the scalability of the learning methods according to different factors was also studied.

Briefly, the conducted study yielded the following conclusions, which greatly help in deciding the learning strategy to be chosen and the configuration according to the target application (for a complete conclusions' list, please refer to [121]):

- CRF models learned from Cornell-RGBD data were more prone to over-fit their parameters than those working with NYUv2. This is due to the higher complexity of the scenes from the Cornell-RGBD.
- The Marginal inference – SGD strategy yielded the highest recognition performance in both datasets: 79.85% in NYUv2 and 67.27% in Cornell-RGBD.
- The PL – L-BFGS strategy was the most robust, providing acceptable results in all the CRF configurations studied.
- LBP was the winning method for testing, reaching the best results when dealing with CRFs with edges and normalized features.
- In general, the computational time is reduced, ranging from the 24.43s. (on average) with the PL – SGD strategy, up to the 71.03s. with the Marginal inference – SGD one.

- L-BFGS and SGD benefited from parallelization techniques in OpenMP, achieving a speed-up factor of ~ 3.5 for PL – L-BFGS, and ~ 5 for Marginal inference -– SGD using 8 CPU-cores.

Concerning the scalability of the studied strategies, it has been analyzed how the utilization of different number of training samples and object categories affect their performance. These experiments reported that the computational time required for learning scales considerably better in both cases when PL – L-BFGS was used, being its growth even sub-linear in some cases. Regarding recognition success, the Marginal inference – SGD option achieved the best outcome.

3.4.4 Exploiting Semantic Knowledge for CRF learning

PGMs in general, and CRFs in particular, need a vast amount of training data in order to reliably encode the gist of the domain at hand. However, the collection of that information is an arduous, time-consuming, and – in some domains – an intractable task that consists of moving the robot from one scene to another, gathering the data, and post-processing it accordingly to the type of information expected by the training algorithms. To face this issue, a framework to codify Semantic Knowledge through human elicitation in an Ontology has been developed, defining the domain object classes, their properties, and their relations. The result is used to generate an arbitrary number of training samples for tuning CRFs. These training samples reify prototypal scenarios where objects are represented by a set of geometric primitives, e.g., planar patches or bounding boxes, that fulfill certain geometric properties and relations, like proximity, difference of orientation, etc. This approach exhibit a number of advantages:

- It eliminates the usually complex and high resource-consuming task of collecting the large number of training samples required to tune an accurate and comprehensive model of the domain.
- Ontologies are compact and human-readable knowledge representations. In that way, extending the problem with additional object classes is just reduced to codify the knowledge about the new classes into the Ontology, generate synthetic samples considering the updated semantic information, and train the CRF. This process can be completed in a few minutes, in contrast to the time needed for gathering and processing real data.
- The recognized objects are anchored to semantically defined concepts, they hence can be straightforwardly incorporated to a semantic map for performing high-level tasks [36, 34, 18].

Thus, the proposed framework follows a top-down methodology (see Figure 3.3). The design starts with the definition of an Ontology for the knowledge domain at hand, e.g. an office environment, through human elicitation, stating the typical objects, their geometrical features, and relations. Then, the encoded Semantic Knowledge is used for generating sets of synthetic samples, which replace the real datasets

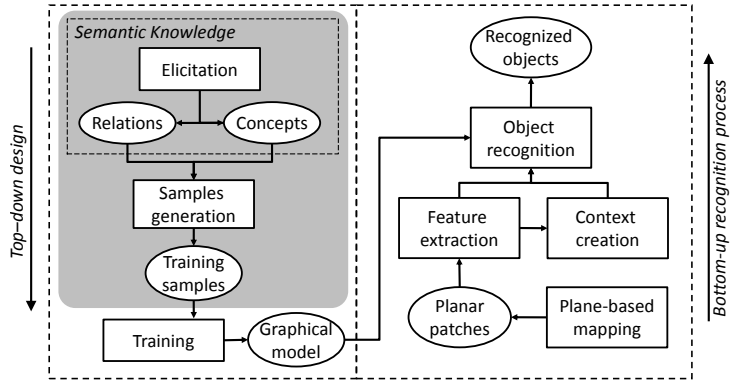


Figure 3.3: Overview of the developed framework for object recognition. The shadowed area delimits the proposed components for the generation of training samples. Boxes represent processes, whereas ovals are generated/consumed data (taken from [116]).

required for training through an algorithm that performs an arbitrary number of times the following steps:

1. **Inclusion of objects in the scene.** The set of objects that appears in the synthetic scene is selected according to their frequency of occurrence codified within the Ontology.
2. **Object characterization.** The geometrical features of the objects included in the previous step, *e.g.* area, centroid height, elongation, orientation, etc. are reified according to their concepts' definitions in the Ontology.
3. **Context creation.** The contextual relations between the included objects are established.
4. **Context characterization.** Different features of those relations are computed, adding valuable contextual information. Examples of these features are: difference between centroid heights, perpendicularity, difference between areas, areas ratio, difference between elongations, etc.

Once the CRF is trained (recall Section 2.1.2), it is integrated into an object recognition framework that works following a bottom-up stance (see Figure 3.3). During the robot operation, a plane-based mapping algorithm [31] extracts planar patches, which are characterized through a number of features, e.g., size, orientation, position or contextual relations. These characterized planar patches feed a probabilistic inference process that yields the recognition results (recall Section 2.1.3).

The results obtained in the conducted evaluations achieved a recognition success of $\sim 90\%$ within the UMA-Offices dataset (see Figure 3.4), and of $\sim 81\%$ and 69.5% using office and home scenes from the NYUv2 dataset respectively, revealing that

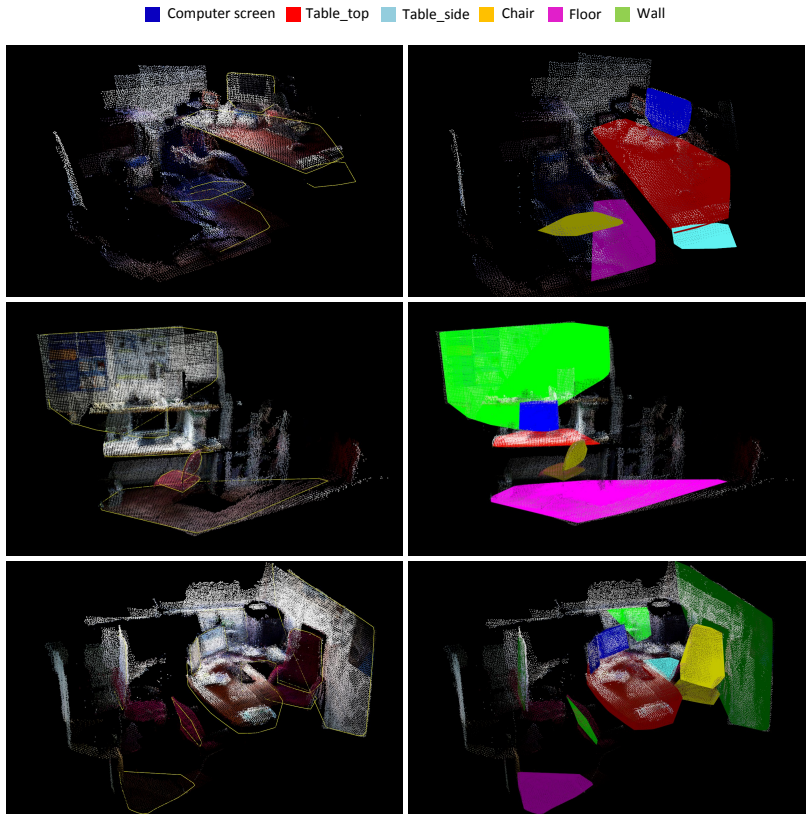


Figure 3.4: Examples of scene object recognitions performed by the proposed framework. Left column, observed scenes from th UMA-Offices dataset with the detected planar patches delimited by yellow lines. Right column, recognition results of such scenes (see [116]).

Semantic Knowledge can be exploited for the suitable training of recognition systems. This approach was also compared with other state-of-the-art approaches based on CRFs, like [154], yielding a substantial improvement.

A number of additional, related issues were also addressed:

- The discriminant capability of different sets of contextual features was studied, showing their positive effect on the system performance.
- The relation between the size of the training datasets and the system performance was analyzed, obtaining the expected conclusions [111]: the larger and the more comprehensive the dataset is, the better the system outcomes are.
- It was also reckoned the computational efficiency, evidencing the suitability of the proposed system for real time robotic applications.

- It was analyzed the time saving gained with the use of human elicitation plus synthetic samples generation processes, resulting 20 times lower than the time spent in collecting real data from the UMA-Offices dataset.

Please refer to [116] for further information about the developed framework, its evaluation, and the reached conclusions.

3.4.5 Including rooms into the equation

The spatial awareness needed by the robot to accomplish high-level tasks must account for the existing close relations among not only objects, but also their typical locations. Thus, the robot should not only tackle the object recognition problem, but also the room recognition one, i.e. to infer the type of space where it is.

Recent publications (e.g. [73, 113]) have shown that the joint modeling of these problems can outperform other methods that address them separately [28, 16, 90, 107, 105]. Holistic approaches exploit the fact that objects are located in rooms according to their functionality, so the presence of an object of a certain type is a hint for the recognition of the room [147, 102, 26]. Likewise, the category of a room is a good indicator of the object types that can be found inside [142]. Besides, objects are not placed randomly, but following configurations that make sense from a human perspective [114, 4, 154]. Thereby, the exploitation of these object-object and object-room contextual clues provides recognition methods with useful information.

For leveraging this information, the framework presented in the previous section has been extended to also consider rooms, recognizing them through the exploitation of their contextual relations. For that, Semantic Knowledge about rooms was codified into the Ontology through human elicitation (see Figure 3.5-top). Figure 3.5-bottom shows the definition of the concept `Microwave` within such Ontology, where we can see, for example, that their orientation is usually horizontal, or that they can be found in kitchens. This Ontology and other resources are available online at: `http://mapir.isa.uma.es/work/objects-rooms-categorization`.

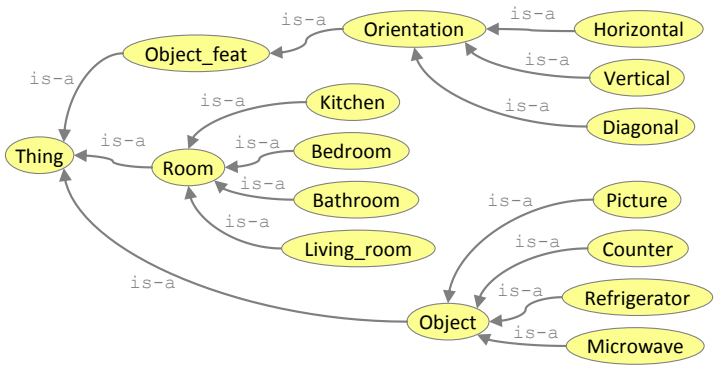
The CRFs employed were also modified in order to consider random variables of different types, *e.g.* taking values from different object types, or from a set of room types, as well as contextual relations of different nature: object-object and object-room relations.

Thereby, two new steps were added to the four-steps algorithm described in the previous section to also generate room-related data. Concretely the new algorithm is:

1. **Room characterization.** The first step is the computation of the room features which, in the used Ontology, includes its volume (m^3) and color hue variation.

2-5. The same four steps as in the original algorithm, but taking into account the type of the room being synthetically generated.

6. **Object-room context characterization.** The relation between the room and its objects is characterized by a fixed value, as it is the training process of the CRF which learns automatically the likelihood of finding an object of a certain



hasBiggestArea **has** MediumLow_Area	var:low
hasColorVariation **has** Medium_ObjectColorVariation	var:high
hasElongation **has** MediumHigh_Elongation	var:veryLow
hasHeight **some** (Medium_Height **or** MediumHigh_Height	var:low
hasOrientation **has** Horizontal	var:null
hasSize **has** MediumLow_ObjectSize	var:low
isPlanar **has** Non_planar	var:medium
isIn **some** Kitchen	freq:veryHigh
isNearTo **some** Counter	freq:veryHihg
isNearTo **some** Refrigerator	freq:medium
isOn **some** Counter	freq:high

Figure 3.5: Top, excerpt of the Ontology used for the codification of Semantic Knowledge about the home domain. Bottom, definition of the concept `Microwave`.

type into a kitchen. Notice that the appearance of an object of a certain type in the room depends on previous steps.

In summary, the above six steps yield the objects, room and contextual features needed to feed the unary and pairwise factors during the training of the CRF. The avid reader can find more information about this process in [117].

The approach has been validated against home scenes from the NYUv2 dataset, reaching a categorization success of $\sim 70\%$ for both objects and rooms. The work by Lin *et al.* [73] also employs CRFs and NYUv2 for validation, and although a fair comparison is not possible since the authors consider a different set of object categories and room types, it permits us to qualitatively confirm the promising performance of the proposed approach, since they achieve a success of $\sim 60.5\%$ and $\sim 58.7\%$ recognizing objects and rooms respectively.

It is worth to mention that the applicability of the framework is not limited to robots working at home environments, but it is suitable to perform in other domains which properties and semantics can be defined by human elicitation, e.g. office facilities or hospitals.

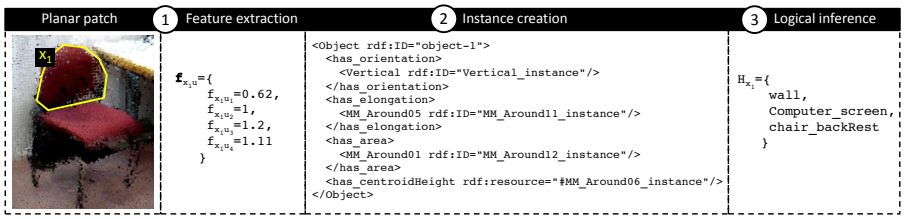


Figure 3.6: Example of hypotheses generation for a given region. New instances are inserted into the Ontology using the OWL language.

3.4.6 Further enhancing CRFs performance: coherence and efficiency

Approximate inference methods executed over CRFs are able to handle complex models and operate in impressive short times, at the expense of a (hopefully) tiny sacrifice in terms of recognition success. Obviously, the utilization of exact inference algorithms is preferable, but the complexity of real models prevents their use. This contribution proposes the exploitation of the Semantic Knowledge encoded in an Ontology to reduce the CRF inference complexity.

Concretely, the Semantic Knowledge is used to generate hypotheses about the most probable belonging classes of the objects according to their features. For example, a horizontal surface with a medium height from the floor could be hypothesized as belonging to the `Chair_seat`, `Table` or `Counter` concepts, but not to `Wall` or `Computer_screen`. These hypotheses are then taken by the CRF as the only possible candidates. This leads to a considerable reduction in the number of combinations, *i.e.* assignments to the random variables, hence decreasing the inference complexity and even enabling, in some cases, exact inference. Moreover, the generation of these hypothesis ensures that the results will be coherent with the information in the Ontology, and consequently, with the Semantic Knowledge that the human encoded about the domain.

The process shown in Figure 3.6 help us to illustrate how hypotheses are generated. First, the object (in this case a chair back) is characterized through a number of features, and a new instance derived from the `Object` concept is inserted into the Ontology, *e.g.* `object-1`, also including a number of properties, or relations, stating such features, *e.g.* `object-1 hasCentroidHeight MM_Around06`. This information is encoded in the Ontology employing the OWL language [10]. Then, a logical reasoner, Pellet [133] in this case, infers a set of concepts that are consistent with the instance definition: `Wall`, `Computer_screen` and `Chair_backRest` in the example. In this way, the CRF only considers that concepts as possible categories for that object, hence decreasing the problem complexity.

Additionally, prior information about the frequency of occurrence of the different object types was also encoded into the Ontology. This type of information permits us

to model that, for example, it is more likely to find a computer than a couch in an office environment, while it is quite unlikely to find an ironing table. This new source of information comes together with a modification to the usual CRF formulation, which can be checked in [114, 119], so it is able to exploit this prior information from the Ontology. This approach enhances even more the expected coherence of the recognition results.

The claimed virtues of these contributions have been thoroughly validated considering the NYUv2 and UMA-Offices datasets. Regarding the recognition success, the evaluation provided the performance of a local object recognition approach as a baseline, which was of $\sim 79\%$ and $\sim 54\%$ for UMA-Offices and NYUv2 respectively, and revealed the progressive increment in the performance and robustness as long as additional information is exploited: contextual information ($\sim 84\%$ and $\sim 59\%$), hypotheses of objects' types ($\sim 93\%$ and $\sim 61\%$), and prior information about object category occurrences ($\sim 94\%$ and $\sim \%65$).

Moreover, an analysis of the complexity reduction of the probabilistic inference process was carried out by considering the most promising object belonging types, including the feasibility of exact inference for the considered datasets. The yielded results are promising, allowing the system to rely on exact inference in all the scenarios within the UMA-Offices dataset, and in a wider variety of them in NYUv2. Further details in this regard can be found in [114, 119].

3.4.7 Learning from experience

Typically, mobile robots employ CRFs that are pre-tuned with a certain dataset in order to recognize a fixed range of object categories. However, this configuration lacks of the flexibility demanded by robots performing in human-like environments, e.g. it is (of course) unable to recognize new types of objects not appearing in the training dataset, or instances of learned ones showing peculiar features, which can lead to an incoherent performance [93]. This section proposes a recognition framework that relies on (surprise) Semantic Knowledge to detect and learn from incoherent recognition results yielded by inference over a CRF.

For example, it can be defined the concept `Fridge` codifying that they are usually high, box-shaped objects, and the `Pill_box` one, stating that they are small boxes related to fridges by `Pill_box placedInto Fridge`. In the proposed framework, the recognition results yielded by probabilistic inference over the CRF are checked for coherence against the Semantic Knowledge. If any of them is detected as incoherent (for example, a middle-size object is classified as a fridge), then it is annotated for its posterior evaluation by the user through a simple dialog. This human-robot interaction is greatly supported by the Ontology, since its content can be verbalized in a straightforward way. Finally, the feedback from the user is back-propagated in order to tune the CRF and the own Ontology accordingly. It is worth to mention that Ontologies also suppose a basic way to understand the robot workspace, enabling the detection of object configurations that can be hazardous, e.g. the pill box found out of the fridge.

More concretely, the recognition pipeline of the recognition framework starts by capturing an image of the scene to be processed to build its CRF graph representation. This graph, along with the pre-trained CRF parameters, is exploited by a probabilistic inference algorithm to provide a set of tentative object recognition results. These results are then inserted as instances in the Ontology, which checks their consistency with respect to the codified Semantic Knowledge by employing a logical reasoner (Pellet). This permits the robot to detect incoherent results that are subsequently evaluated by the user. The evaluation of a conflicting object starts by showing him/her a cropped image of it. Three different scenarios are then possible:

Case 1: the user determines that the recognition result is right. This means that the CRF performed correctly, but the codified common-sense knowledge was somehow too strict. The Ontology learns from this outcome by relaxing the codified object property that produced the inconsistency.

Case 2: the recognition result is wrong, and:

- **Case 2.1:** the object type is already present in the CRF/Ontology. In this case the CRF misclassified the object. To learn from the mistake, the gathered object information is used to re-tune the CRF parameters.
- **Case 2.2:** the object type is new. The relevant information from the object is used to automatically generate a new concept in the Ontology, and the CRF is also re-trained taking into account this new object type.

To perform a proof-of-concept validation of the framework, a robot was deployed into an apartment and commanded to perform a primary task: to check the configuration of the objects in the kitchen. Concretely, during the robot operation, the RGB-D camera was used to capture both intensity and depth images when reaching certain locations in the kitchen. In that setup, the robot detected an inconsistency, which corresponded to a pill box recognized as a cereal box, since such object type was unknown for the robot. This information was then back-propagated to both: (i) the Ontology, where the system created a new concept `Pill_box`, inheriting from the `Object` one, and described it with the information gathered from the human and from the collected sensory data, and (ii) to the CRF model, which re-tuned its parameters according to the new information. The learning success was evaluated in later observations of pill boxes, where the robot was able to successfully recognize this new type of object.

3.5 Discussion

This chapter has described the thesis' contributions to the contextual object and/or room recognition problem. It started with the *UMA-Offices* dataset, a collection of 3D reconstructions of offices from the University of Málaga, which was necessary for evaluating the developed algorithms. Then, the *Undirected Probabilistic Graphical Models in C++* (UPGMpp) library has been presented, which permits the efficient handling of Undirected PGMs when applied to robotic-related applications.

PGMs in general, and CRFs in particular, have proven to be valuable frameworks for the modeling of recognition problems exploiting contextual information, also dealing with uncertainty. The effort needed for the collection and processing of *UMA-Offices* and other sensory data, along with the hungry for comprehensive and large training datasets exhibited by the learning phase of PGMs, motivated the study of alternative training strategies. This led to the utilization of Semantic Knowledge stored within an Ontology to remove the necessity of a real dataset. This is specially useful in domains where it is difficult, or even infeasible the collection of large amounts of data. Ontologies also provide the recognition system with an structured, human readable representation ready-to-use for high-level robotic tasks.

Semantic knowledge has been further exploited for reducing the complexity of the probabilistic inference processes over the CRFs, as well as to provide prior knowledge about the frequency of occurrence of the object classes of the domain at hand. This information is incorporated into the usual CRF formulation in order to enhance its performance. It has been also leveraged for detecting incoherent recognition results, by considering a logical reasoner that checks the consistency of the CRF outcome with respect to the encoded knowledge. This also allows the recognition system, including an user in the loop (supervised learning), to learn from experience by automatically adapting its internal representations.

These contributions make up a probabilistic recognition system which is able to: (i) exploit contextual relations, (ii) handle uncertainty, (iii) leverage prior knowledge about the domain at hand, (iv) detect incoherent results, (v) learn from experience, and (vi) verbalize its outcome. In addition to these features, the system can also provide a measure about the uncertainty of its results. Finally, the system has been integrated into a semantic mapping framework specially suited for taking advantage of these features, as shown in the next chapter.

4 Semantic Mapping

This chapter outlines the thesis's contributions to the semantic mapping field. It starts with a brief introduction to the problem and a discussion of relevant works in the literature. Then, it describes: a toolkit for labeling sequential RGB-D datasets, the Robot@Home repository processed by that toolkit, and finally the Multiversal Semantic Map, a novel representation evaluated through Robot@Home.

4.1 Introduction

Despite the possibilities of geometric and/or topological maps when applied to mobile robot applications, the planning and execution of high-level tasks like "bring me the red cup from the kitchen's counter" or "show the customer off-season clothing, specially pants, please" demands more sophisticated maps. Humans share semantic knowledge about concepts like *red*, *cup*, or *off-season clothing*, which must be transferred to robots in order to successfully face these tasks. *Semantic maps* emerged to cope with this need, providing the robot with the capability to *understand*: (i) the spatial aspects of human environments, (ii) the meaning of their elements (objects, rooms, or facilities), and (iii) how humans interact with them (*e.g.* functionalities, events, or relations).

This feature is distinctive and traversal to semantic maps, being the key difference with respect to maps that simply augment metric/topological models with labels to state the type of recognized objects or rooms [108, 22, 76, 127], *e.g.* saying that a portion of sensory data is a cup, without any other information about the *implications* of that. Contrary, semantic maps handle meta-information that models the properties

and relations of relevant concepts therein the domain at hand, codified into a *Knowledge Base* (KB) and stating that, for example, cups are cylindrical-shaped objects usually found in kitchens and useful for containing liquids. Building and maintaining semantic maps involve the symbol grounding problem [49, 18], *i.e.* linking portions of the sensory data gathered by the robot (percepts), represented by symbols (*e.g.* `object-1` or `room-1`), to concepts in the KB by means of some recognition and tracking method. These representations usually reckon on off-the-shelf recognition methods to individually ground percepts to particular concepts, which disregard the valuable contextual relations between the workspace elements: a rich source of information intrinsic to human-made environments (for example that night-stands are usually in bedrooms and close to beds).

Semantic maps generally support the execution of reasoning engines, providing the robot with inference capabilities for efficient navigation, object search, or proactiveness [36], among others. Typically, such engines are based on logical reasoners that work with crispy information (*e.g.* a percept is identified as a cup or not). The information encoded in the KB, along with that inferred by logical reasoners, is then available for a task planning algorithm dealing with this type of knowledge and orchestrating the aforementioned tasks [35]. Although crispy knowledge-based semantic maps can be suitable in some setups, especially in small and controlled scenarios [156], they are also affected by uncertainty coming from different sources like the robot sensory system or the inaccurate modeling of the elements within the robot workspace.

This chapter presents the contributions done for achieving a semantic map representation able to deal with uncertainty, also managing contextual relations, where the techniques outlined in Chapter 3 play a pivotal role (Section 4.3.3). In addition, given the lack of datasets for evaluating mapping systems with those features, we also describe a repository of information especially collected for that goal, the *Robot@Home* dataset (Section 4.3.2), as well as a toolkit developed for the efficient processing of this type of repositories, the *Object Labeling Toolkit* (Section 4.3.1).

4.2 Related work

This section reviews the most relevant works addressing some issues related to the semantic mapping problem, starting with a discussion about popular semantic representations (Section 4.2), continuing with an analysis of the datasets that are suitable as a testbed for such approaches (Section 4.2), and finishing with a discussion on available tools for managing datasets (Section 4.2).

Semantic mapping approaches

In the last decade, a number of works have appeared in the literature contributing different semantic map representations. One of the earliest works in this regard is the one by Galindo *et al.* [37], where a multi-hierarchical representation models, on the one hand, the concepts of the domain of discourse through an ontology, and on the

other hand, the elements from the current workspace in the form of a spatial hierarchy that ranges from sensory data to abstract symbols. NeoClassic is the chosen system for knowledge representation and reasoning through Description Logics (DL), while the employed recognition system is limited to the classification of simple shape primitives, like boxes or cylinders, as furniture, e.g. a red box represents a couch. The potential of this representation was further explored in posterior works, *e.g.* for improving the capabilities and efficiency of task planners [35], or for the autonomous generation of robot goals [36]. A similar approach is proposed in Zender *et al.* [156], where the multi-hierarchical representation is replaced by a single hierarchy ranging from sensor-based maps to a conceptual abstraction, which is encoded in a Web Ontology Language (OWL)–DL ontology defining an office domain. To categorize objects, they rely on a SIFT-based approach, while rooms are grounded according to the objects detected therein. In Nüchter and Hertzberg [88] a constraint network implemented in Prolog is used to both codify the properties and relations among the different planar surfaces in a building (wall, floor, ceiling, and door) and classify them, while two different approaches are considered for object recognition: a SVM-based classifier relying on contour-based features, and a Viola and Jones cascade of classifiers reckoning on range and reflectance data.

These works set out a clear road for the utilization of ontologies to codify semantic knowledge, which has been further explored in more recent research. An example of this is the work by Tenorth *et al.* [138], which presents a system for the acquisition, representation, and use of semantic maps called KnowRob-Map, where Bayesian Logic Networks are used to predict the location of objects according to their usual relations. The system is implemented in SWI-Prolog, and the robot's knowledge is represented in an OWL-DL ontology. In this case, the recognition algorithm classifies planar surfaces in kitchen environments as tables, cupboards, drawers, ovens and dishwashers [127]. The same map type and recognition method is employed in Pangercic *et al.* [95], where the authors focus on the codification of object features and functionalities relevant to the robot operation in such environments. The paper by Riazuelo *et al.* [112] describes the RoboEarth cloud semantic mapping which also uses an ontology for codifying concepts and relations, and rely on a Simultaneous Localization and Mapping (SLAM) algorithm for representing the scene geometry and object locations. The recognition method resorts to SURF features, and performs by only considering the object types that are probable to appear in a given scene (the room type is known beforehand). In Günther *et al.* [45], the authors employ an OWL-DL ontology in combination with rules defined in the Semantic Web Rule Language (SWRL) to categorize planar surfaces.

It has been also explored the utilization of humans for assisting during the semantic map building process through a situated dialog Examples of works addressing this are those by Bastianelli *et al.* [9], Gemignani *et al.* [40], or the aforementioned one by Zender *et al.* [156]. The main motivation of these works is to avoid the utilization of recognition algorithms, given the numerous challenges that they have to face. However, they themselves argue that the more critical improvement of their proposals would arise from a tighter interaction with cutting-edge recognition techniques.

The interested reader can refer to the survey by Kostavelis and Gasteratos [67] for an additional, comprehensive review of semantic mapping approaches for robotic tasks.

The semantic mapping techniques discussed so far rely on crispy categorizations of the perceived spatial elements, *e.g.* an object is a cereal box or not, a room is a kitchen or not, etc., which is typically exploited by (logical) reasoners and planners for performing a variety of robotic tasks. As commented before, these approaches: (i) can lead to an incoherent robot operation due to ambiguous recognition results, and (ii) exhibit limitations to fully exploit the contextual relations among spatial elements. The contributions in the previous chapter propose a solution for probabilistic symbol recognition to cope with both, the uncertainty inherent to the recognition process, and the contextual relations among spatial elements. Perhaps the closet work to this approach addressing semantic mapping is the one by Pronobis and Jensfelt [103], which employs a Chain Graph (a graphical model mixing directed and undirected relations) to model the grounding problem from a probabilistic stance, but that fails at fully exploiting contextual relations. This thesis contributes, among others, a novel representation called Multiversal Semantic Map (*MvSmap*), in order to accommodate and further exploit the outcome of the probabilistic symbol grounding.

Suitable datasets

Datasets containing sensory data are needed for a thorough evaluation of semantic mapping techniques, since they set a common framework for their fair comparison. Mobile robots have traditionally resorted to intensity images to categorize objects and/or rooms, which motivated the collection of datasets providing this kind of information [27, 125, 124]. Nowadays, the tendency is for the datasets to also include depth information [57, 5, 72], given the proved benefits of exploiting morphological and spatial information in assisting recognition methods [114]. These datasets can be roughly classified as: *object-centric*, *view-centric*, and *place-centric*.

Object-centric datasets, like ACCV [51], RGBD Dataset [72, 71], KIT object models [62], or BigBIRD [132], provide RGB-D observations in which a unique object spans over each image. The exploitation of these images for robotic recognition exhibits some drawbacks: (i) they are not representative of the typical images gathered by a robot at a real environment, (ii) they prevent the utilization of valuable contextual information of objects, and (iii) they are not suitable for the room recognition problem. These shortcomings also narrow their utilization by semantic mapping benchmarks.

On the other hand, *view-centric* datasets as Berkeley-3D [57], Cornell-RGBD [5], NYU [130, 131], TUW [3], or UBC VRS [77], consist of isolated RGB-D images, or a sequence of them, which cover a partial view of the working environment. This information permits the exploitation of contextual information but only from a local, reduced perspective, since information of the entire scene is not collected. Therefore, their use for contextual recognition is still limited, as well as their utilization for semantic mapping purposes.

Table 4.1: Summary of related datasets (CR: Collected by a robot, DT: Dataset type, EOC: Enables object context exploitation, ERC: Enables room categorization).

Dataset	CR	DT	EOC	ERC
ACCV [51]		*object-centric*		
Berkeley-3D [57]		*view-centric*	✓(local)	✓(limited)
BigBIRD [132]		*object-centric*		
Cornell-RGBD [5]	✓	*view-centric*	✓(local)	✓(limited)
KIT object models [62]		*object-centric*		
Multi-sensor 3D Object Dataset [39]		*object-centric*		
NYUv1 [130]		*view-centric*	✓(local)	✓(limited)
NYUv2 [131]		*view-centric*	✓(local)	✓(limited)
RGBD Dataset [72]		*object-centric*		
RGBD Dataset 2 [71]		*object-centric*		
TUW [3]	✓	*view-centric*	✓(local)	✓(limited)
SUN3D [153]		*place-centric*	✓	✓
UBC VRS [77]	✓	*view-centric*	✓(local)	
Robot@Home	✓	*place-centric*	✓	✓

Finally, *place-centric* datasets like SUN3D [153] provide comprehensive information from the inspected room, or even the entire work environment, typically through the registration of RGB-D images. This type of datasets conforms the best option as a testbed for semantic mapping taking advantage of both depth and contextual information, albeit, unfortunately their number is quite limited. A dataset worth to mention at this point is ViDRILO [75], which comprises 5 sequences of RGB-D observations of two office buildings collected by a robot combining *object* and *environment-centric* perspectives. This dataset annotates each observation with its room type and the objects found within it, although this labeling is not per-pixel and the number of object categories is reduced. Table 4.1 shows a summary of datasets applicable to the semantic problem and their characteristics, which also includes the one contributed by this thesis: the *place-centric* Robot@Home dataset.

Available dataset management tools

The tedious object labeling task within RGB-D datasets is carried out in different ways. Some works resort to *Amazon Mechanical Turk* (AMT) to label their intensity images [57, 130, 131], usually through a labeling tool like LabelMe [125], but this merely divides the workload, and the annotated information still needs to be thoroughly checked to fix incoherent labels. Another approach is the manual labeling of *key intensity frames* from a sequence, propagating these labels to the remaining RGB-D observations [77, 153], but this is only suitable for sequences with simple sensor trajectories, and additionally shows the same limitations as the AMT option. There are also works that reconstruct a 3D representation of the inspected scene and annotate the objects appearing on it [5], but there is not a *labeling feedback* to the RGB-D

Figure 4.1: OLT logo.

observations' sequence(s). In the works by Lai *et al.* [72, 71] the ground truth annotations over a reconstructed scene are also propagated to the individual RGB-D observations employing an ad-hoc software which, to the best of the author knowledge, is not publicly available. In the next section it is described an open source solution conveniently divided into configurable components, which provides the robotic community with a number of functionalities towards an efficient labeling of arbitrarily large collections of RGB-D data.

4.3 Contributions

Three contributions are outlined in this chapter, all of them in the scope of the semantic mapping problem. First, the Object Labeling Toolkit (OLT) is described. It consists of a set of software solutions for the labeling of sequential RGB-D datasets, especially relevant to semantic mapping. Then, we describe a novel *place-centric* dataset, named Robot@Home, which contains raw and processed data from domestic settings compiled by a mobile robot. Finally, the *Multiversal Semantic Map* is presented, an environment representation able to handle uncertainty and contextual relations, in which the contributions of the previous chapter are integrated.

4.3.1 The Object Labeling Toolkit

A comprehensive dataset is a valuable benchmark tool for tuning, testing, and comparing robotic algorithms and systems in a convenient and fair way. Although public datasets consisting of intensity images [27, 125, 124] have largely helped researchers to push ahead the state-of-the-art in object recognition or scene interpretation, nowadays new particularly oriented datasets are required given the increasing number of capabilities and applications that are demanded to a mobile robot, e.g. semantic mapping [104], high-level decision making [36], or contextual object recognition [116, 114, 115, 119].

RGB-D cameras have become a key source of information for such *robotic* datasets. Although the sensory data of these datasets may be conveniently gathered by the mobile robot itself, human supervision is still needed to segment objects and to

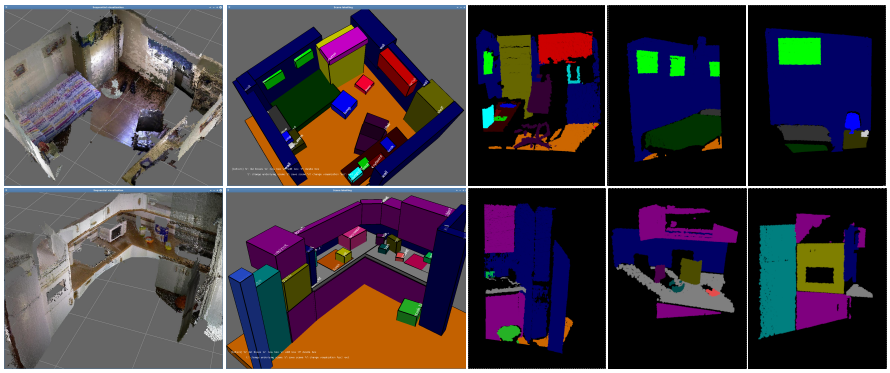


Figure 4.2: First column, reconstructed scenes from two RGB-D sequences. Second column, labeled reconstructed scenes. Third-fifth columns, examples of individual point clouds from RGB-D observations labeled by the propagation of the annotations within the reconstructed scenes.

label them, i.e. to add annotations over portions of the observed data as belonging to a certain object class, e.g. floor, table, lamp, etc. This is the motivation for the development of the *Object Labeling Toolkit* (OLT, see Figure 4.1), *i.e.* to provide the robotic community with a tool to efficiently label datasets compound of sequences of RGB-D observations, gathered from an arbitrary number of RGB-D sensors. OLT is publicly available under a GNU General Public License at: `http://mapir.isa.uma.es/work/object-labeling-toolkit`.

For achieving such efficient labeling, the toolkit builds a 3D reconstruction of each RGB-D sequence within a given dataset, and allows the user to graphically label objects within that reconstruction (see the two first columns in Figure 4.2). Then, this ground truth annotations are automatically propagated to all the RGB-D observations without requiring human supervision, resulting in a dense labeling of both intensity and depth data (see the three last columns of Figure 4.2). More information about this pipeline can be found in the publication by Ruiz-Sarmiento *et al.* [118].

OLT comprises a number of software components covering the following functionality: i) dataset pre-processing, ii) localization of RGB-D observation poses, iii) 3D scene reconstruction, iv) labeling of the reconstructed scene, and v) automatic propagation of annotated labels. Some of these functionalities can exploit additional information coming from sensors usually present in a robotic platform, e.g. the robot pose estimation computed from 2D laser scans. All the components are highly customizable in order to fit the particularities of robotic datasets, and can be easily expandable to integrate other algorithms of interest. The toolkit resorts to the Mobile Robot Programing Toolkit (MRPT [58]) and the Point Cloud Library (PCL [126]) for point cloud registration and smoothing algorithms, and for data representation and vi-

sualization purposes. The most time-consuming components of OLT have been also parallelized employing OpenMP.

Aiming to illustrate the toolkit suitability, it was utilized for segmenting and labeling a robotic dataset from a home environment (indeed, a part of the Robot@Home dataset, see Figure 4.2) consisting of 77 RGB-D observations. Regarding the time spent in labeling, the human operator needed 2 hours to annotate both the kitchen and the bedroom scenes, spending on average 2 minutes per object (this has been reduced to 1 minute in the last toolkit version). To compare this with the labeling of all the RGB-D observations individually, it was followed the typical intensity image labeling approach and they were annotated 5 non-consecutive observations from each sequence, extrapolating the results to the whole dataset. This yielded a total of $\sim$3 hours needed for the labeling of the kitchen sequence, and $\sim$7 hours for the bedroom, which clearly illustrated the benefits of the toolkit utilization. When following such a typical approach problems appeared to accurately label the objects' boundaries, and with objects partially occluded and/or with an unclear belonging class, drawbacks that are mitigated with the utilization of the proposed toolkit.

4.3.2 Robot@Home dataset

The Robot-at-Home (Robot@Home) dataset, is a collection of raw and processed data from five domestic settings compiled by the commercial mobile robot Giraff, equipped with 4 RGB-D cameras and a 2D laser scanner, Its main purpose is to serve as a testbed for semantic mapping algorithms through the recognition of objects and/or rooms, so it is publicly available at `http://mapir.isa.uma.es/work/robot-at-home-dataset`. This dataset is unique in three aspects: (i) the sensory system employed for its gathering, (ii) the diversity and amount of provided data, and (iii) the availability of dense ground truth information.

The provided data were captured with a rig of 4 RGB-D sensors with an overall field of view of 180° horizontally and 58° vertically, and with a 2D laser scanner (see Fig. 4.3). In order to yield accurate information within the dataset, the sensors mounted on the robot were calibrated both intrinsically and extrinsically [30, 42, 137]. Detailed information concerning this calibration in particular, and about the dataset in general, can be found in the paper by Ruiz-Sarmiento *et al.* [123].

This robotic platform was employed to explore 5 dwelling apartments, which have been named as *anto*, *alma*, *pare*, *rx2*, and *sarmis*. In this way, a total of 36 rooms were completely inspected (some of them several times), so the dataset is rich in contextual information of objects and rooms. This is a valuable feature, missing in most of the state-of-the-art datasets, which can be exploited by, for instance, semantic mapping systems that leverage relationships like *pillows are usually on beds* or *ovens are not in bathrooms*. This information was processed by OLT, which also supposes a mechanism to conveniently access and manage the data.

The ground-truth information provided by OLT comes in two flavors. On the one hand, it is provided (per-point) annotations of the categories of the main objects and rooms appearing in the scenes reconstructed from the RGB-D sequences (recall the

Figure 4.3: Giraff robot while collecting sensory information. The basic robotic platform was endowed with a rig of 4 RGB-D sensors mounted on the *robot's neck*, and a 2D laser scanner on its base.

second column of Figure 4.2). A total of ∼1,900 objects belonging to 157 different categories were manually labeled from the 36 visited rooms. These rooms are also labeled as belonging to one of 8 possible types: bathroom, bedroom, kitchen, living-room, etc. On the other hand, Robot@Home also includes (per-pixel) annotations of the objects appearing in the 69,000+ gathered RGB-D images. The objects and rooms are also annotated with identifiers, so they can be individually tracked along the video sequences.

Summarizing, the content of the dataset, which comes in different formats accessible by the open source Mobile Robot Programming Toolkit[1] (MRPT), as well as in (human readable) plain text files and PNG images, is as follows:

- **81** sequences of observations containing ∼75min. of recorded data. The total number of observations is **87,000+** (18,000+ laser scans and 69,000+ RGB-D images), which are saved in *rawlog* format as well as in plain text (see the three first rows of Figure 4.4).
- **41** 2D geometric maps saved in text files (36 for individual rooms, and 5 maps covering each apartment, see fourth row of Figure 4.4).
- **72** 3D reconstructed scenes in *scene* format and plain text (see fifth row of Figure 4.4).
- **72** Labeled 3D reconstructed scenes in *scene* format and plain text, containing ∼**1,900** labeled objects (see sixth row of Figure 4.4).
- **72** Labeled RGB-D sequences in *rawlog* format and plain text (see seventh row of Figure 4.4).

[1] http://www.mrpt.org

Figure 4.4: Excerpts of information provided by Robot@Home. From top to bottom, examples of 2D laser scans, RGB images, depth images, 2D geometric maps, reconstructed rooms, labeled reconstructed rooms, and labeled depth information. *Taken from [123].*

Moreover, a number of particular characteristics have been intentionally included in each scenario to provide additional data for testing different object recognition algorithms and techniques. Concretely,

- **Inclusion of distinctive objects.** A number of patterns/objects have been placed at different rooms within these houses, concretely: teddies in *alma*, fruits in *anto*, numerical patterns in *pare* and geometric patterns in *rx2*.
- **Varying lighting conditions**. Each of the three sessions in *sarmis* house was conducted at a different time of the day, which means that the objects were visualized under different lighting conditions.
- **Varying sets of objects**. In these three sessions, the set of objects placed in each room from session to session differs, with objects dis/appearing as well as being moved.

Although its main application is the aforementioned semantic mapping, it can be also useful for the recognition of instances of objects/rooms, object segmentation, or data compression/transmission algorithms. Moreover, typical robotic tasks like 3D map building, localization, or SLAM can be tested with Robot@Home, since the robot localization can be accurately estimated from the sequence of 2D scans. Finally, the distinctive patterns and objects placed on purpose can be used, for example, to test object-finding algorithms.

4.3.3 Multiversal Semantic Maps

The third contribution of this chapter is a novel semantic map representation, called *Multiverse Semantic Map* (*MvSmap*). This representation handles uncertainty by considering the different combinations of possible groundings of objects and rooms in the robot workspace, or *universes*, as instances of ontologies with belief annotations on their grounded concepts and relations. These beliefs are provided by the probabilistic recognition techniques described in Chapter 3. According to them, it also encodes the probability of each ontology instance being the right one. Thus, *MvSmaps* can be exploited by logical reasoners performing over such ontologies, as well as by probabilistic reasoners working with the CRF representation. This ability to manage different semantic interpretations of the robot workspace, which can be leveraged by probabilistic conditional planners (*e.g.* those in [61] or [2]), is crucial for a coherent robot operation.

The proposed *MvSmap* (see Figure 4.5) is inspired by the multi-hierarchical semantic map presented in Galindo *et al.* [37]. This map considers two separated but tightly related hierarchical representations containing: (i) the semantic, meta-information about the domain at hand, *e.g.* refrigerators keep food cold and are usually found in kitchens, and (ii) the factual, spatial knowledge acquired by the robot and its implemented algorithms from a certain workspace, *e.g.* `obj-1` is perceived and recognized as a refrigerator. These hierarchies are called terminological

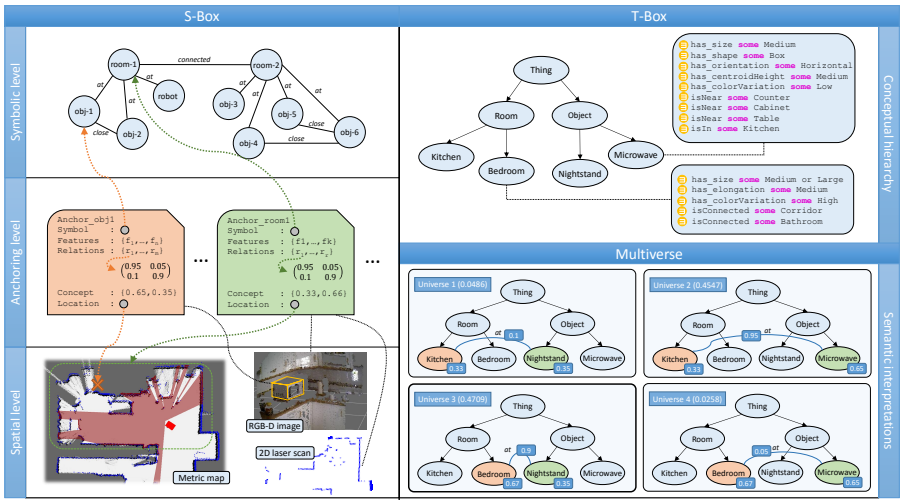


Figure 4.5: Example of Multiversal Semantic Map for a simple scenario.

box (see *T-Box* in Figure 4.5) and spatial box (see *S-Box* in Figure 4.5), respectively, names borrowed from the common structure of hybrid knowledge representation systems [7].

MvSmaps enhance this representation by including uncertainty, in the form of *beliefs*, about the groundings (recognitions) of the spatial elements in the S-Box to concepts in the T-Box. For example, a perceived object, represented by the symbol `obj-1`, could be grounded by the robot as a microwave or a nightstand with beliefs 0.65 and 0.35, respectively, or it might think that a room (`room-1`) is a kitchen or a bedroom with beliefs 0.33 and 0.66. Moreover, in this representation the relations among the spatial elements play a pivotal role, and they have also associated compatibility values in the form of beliefs. To illustrate this, if `obj-1` was found in `room-1`, *MvSmaps* can state that the compatibility of `obj-1` and `room-1` being grounded to microwave and kitchen respectively is 0.95, while to microwave and bedroom is 0.05. These belief values are provided by the proposed probabilistic inference techniques.

Furthermore, *MvSmaps* assign a probability value to each possible set of groundings, creating a *multiverse*, *i.e.* a set of universes stating different explanations of the robot environment (see *Multiverse* in Figure 4.5). An universe codifies the joint probability of the observed spatial elements being grounded to certain concepts, hence providing a global sense of certainty about the robot understanding of the environment. Thus, following the previous example, an universe can represent that `obj-1` is a microwave and `room-1` is a kitchen, while a parallel universe states that `obj-1` is a nightstand and `room-1` is a bedroom, both explanations annotated with different probabilities. Thereby, the robot performance is not limited to the utilization of

the most probable universe, like traditional semantic maps do, but it can also consider other possible explanations with different semantic interpretations, resulting in a more coherent robot operation.

The symbol grounding problem, *i.e.* linking portions of sensory data, represented by symbols (*e.g.* `obj-1` or `room-2`), to concepts in the KB (*e.g.* `Microwave` or `Kitchen`), is faced by an anchoring process [18] that relies on the proposed recognition techniques and a simple tracking algorithm to make the symbols and their groundings consistent over time. In a nutshell, the result of this process is a set of the so-called anchors, which keep geometric/appearance information about the spatial elements (location, features, relations, etc.) and establish links to their symbolic representation. Additionally, in a *MvSmap*, anchors are in charge of storing the beliefs about the grounding of their respective symbols, as well as their compatibility with respect to the grounding of related elements.

Given the ingredients of *MvSmaps* previously provided, a *Multiversal Semantic Map* can be formally defined by the quintuple $\mathscr{M}v\mathscr{S}map = \{\mathscr{R}, \mathscr{A}, \mathscr{Y}, \mathscr{O}, \mathscr{M}\}$, where:

- $\mathscr{R}$ is a metric map of the environment, providing a global reference frame for the observed spatial elements (objects and rooms).
- $\mathscr{A}$ is a set of anchors internally representing such spatial elements, and linking them with the set of symbols in $\mathscr{Y}$.
- $\mathscr{Y}$ is the set of symbols that represent the spatial elements as instances of concepts from the ontology $\mathscr{O}$.
- $\mathscr{O}$ is an ontology codifying the semantic knowledge of the domain at hand.
- $\mathscr{M}$ encodes the multiverse, containing the set of universes.

Notice that the traditional T-Box and S-Box are defined in a *MvSmap* by $\mathscr{O}$ and $\{\mathscr{R}, \mathscr{A}, \mathscr{Y}\}$ respectively. Since the robot is usually provided with the ontology $\mathscr{O}$ beforehand, building a *MvSmap* consists of creating and maintaining the remaining elements in the map definition.

The suitability of the proposed semantic map representation was assessed with the challenging Robot@Home dataset. On the one hand, the reported success while grounding object and room symbols respectively without considering contextual relations was of $\sim 73.5\%$ and $\sim 57.5\%$, whereas including them these figures increased up to a success of $\sim 81.5\%$ and 91.5%. They have been also evaluated some of the most popular classifiers also resorting to individual object/room features, namely: Supported Vector Machines, Naive Bayes, Decision Trees, Random Forests, and Nearest Neighbors, demonstrating the reported results the higher success of CRF approaches.

On the other hand, they were also shown two sample scenarios of different complexity where it was illustrated the building of *MvSmaps* according to the information gathered by a mobile robot (see Figure 4.6). For a detailed description of this results,

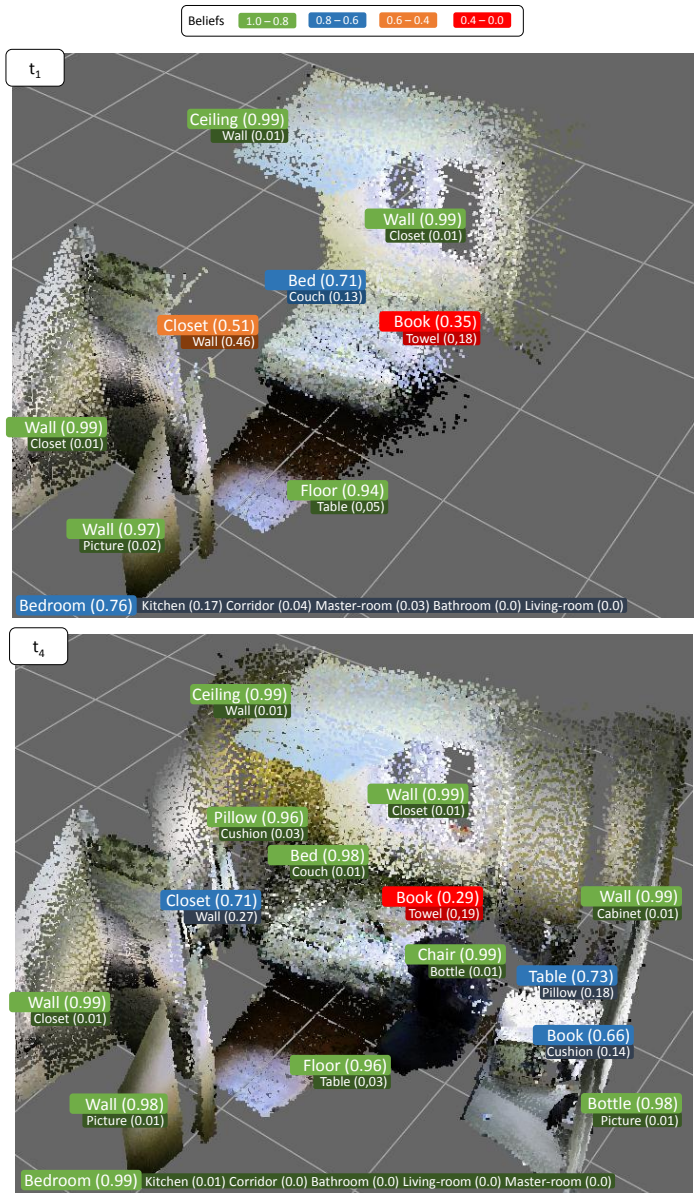


Figure 4.6: Grounding results and their belief values for the spatial elements perceived during the robot exploration of a bedroom from Robot@Home at two time instants: t_1 and t_4.

as well as of the building of *MvSmaps*, please refer to the work by Ruiz-Sarmiento *et al.* [120].

The main purpose of the proposed *MvSmap* is to provide a mobile robot with a rich representation of its environment, empowering the efficient and coherent execution of high-level tasks. *MvSmaps* can be exploited for traditional semantic map applications by considering only an universe, albeit its potential to measure the (un)certainty of the robot understanding can be exploited for an intelligent, more efficient robotic operation. A clear example of this arises when considering the work by Galindo and Saffiotti [36], which envisages an application of semantic maps where they encode information about how things should be, also called norms, allowing the robot to infer deviations from these norms and act accordingly. The typical norm example is that "towels must be in bathrooms", so if a towel is detected, for example, on the floor of the living room, a plan is generated to bring it to the bathroom. This approach works with crispy information, *e.g.* an object is a towel or not. Instead, the consideration of a *MvSmap* would permit the robot to behave more coherently, for example gathering additional information if the belief of an object symbol being grounded to `Towel` is 0.55 while to `Carpet` is 0.45. In this example, a crispy approach could end up with a carpet in our bathroom, or a towel in our living room. Other applications where *MvSmaps* could be useful are task planning, planning with incomplete information, navigation, object search, human-robot interaction, or robotic localization.

4.4 Discussion

This chapter has outlined the thesis' contributions to the semantic mapping field. A novel semantic representation, called *Multiversal Semantic Map* (*MvSmap*), has been described, which was designed to take advantage of the outcome for probabilistic recognition techniques. This permits the robot to propagate the uncertainty coming from different sources like its sensory system, or its internal models of the spatial elements, to the recognition results. *MvSmaps* also allow the tracking and exploitation of contextual relations among the elements in the robot workspace. The utilization of the uncertainty concerning the types of recognized spatial elements enables the robot to consider different semantic interpretations of its environment, resulting in a more coherent operation.

Additionally, it has been also described the *Robot@Home* dataset, a large repository of data collected by a mobile robot in domestic settings. The provided raw data come from two different types of sensors: a 2D laser scan mounted on the robot base, and a rig of 4 RGB-D cameras on the robot's neck. The processed information includes 2D and 3D reconstructions of the fully inspected houses, as well ground truth annotations about the type of the objects and rooms therein. Thus, this dataset is rich in contextual relations among spatial objects given the wide coverage of the provided data, so it is specially suitable for the evaluation of semantic mapping systems. To evaluate the proposed *MvSmaps*, the recognition techniques in the previous chapter

were integrated into a semantic mapping system building those representations, and *Robot@Home* was used as a testbed.

Robot@Home contains a huge number of observations whose processing by traditional techniques is prohibitive. Thereby, it was developed the *Object Labeling Toolkit* (OLT), a set of software components that greatly minimizes the operator intervention for processing sequential RGB-D observations. The developed/integrated algorithms for image processing, point cloud registration, scene reconstruction, scene labeling, and automatic propagation of labels to individual observations, really helped to keep the effort low for processing *Robot@Home*. Both dataset and toolkit are publicly available.

5 Summary of included papers

*This chapter outlines the content of the included papers, available at the second part of the thesis **Part II: Included papers**, as well as the author's contributions to each of them.*

5.1 Paper A: Learning CRFs with data from Semantic Knowledge

Outline: This paper studies the applicability of CRFs trained with synthetic data, generated from Semantic Knowledge, for contextually modeling the scene object recognition problem. The proposed learning approach aims at avoiding the collection of real data for training object recognition systems, which is a highly time-consuming, cumbersome, and even unfeasible task, since the gathered information must be representative enough of the domain at hand. To face this issue, Semantic Knowledge is represented by means of an Ontology, which defines the domain object classes, their properties, and their relations, and is used to generate synthetic training samples for tuning CRFs. The suitability of the learning approach has to be assessed through real datasets, so UMA-Offices and NYUv2 conformed the benchmark for answering questions like: *How much do the context relations contribute to the recognition performance?*, *How much does the size of the training dataset affect the recognition performance?*, or *Do the generated synthetic data capture actual object properties and relations?*.

Contribution by the author: Studied the state-of-the-art approaches for addressing the scene object recognition problem through Probabilistic Graphical Models or

Semantic Knowledge. Designed the way in which the relevant information can be encoded in an Ontology for its posterior exploitation. Implemented the algorithm for the automatic generation of an arbitrary number of synthetic training samples. Processed the UMA-Offices dataset, and performed the experiments to demonstrate the suitability of the approach.

5.2 Paper B: Joint recognition of objects and rooms

Outline: This work extends the previous one by including rooms in the equation. Motivated by recent studies that highlight the convenience of jointly modeling the object and room recognition problems (in view of the mutual influence between the types of the recognized rooms and the the types of the objects therein), the ontology defined in Paper A is augmented to also consider room classes, their attributes, and relations among them as well as among objects and rooms: *e.g.* that bedrooms are usually connected to corridors and beds can be found therein. The CRF models are also conveniently adapted for dealing with different types of random variables (taking values from object or room types) and contextual relations. To validate the approach the paper resorts to home scenes from the NYUv2 dataset.

Contribution by the author: Studied state-of-the-art techniques for jointly modeling the object and room recognition problems. Designed the expansion of the Ontology in the previous paper, as well as of the CRF formulation and the algorithm implemented for generating synthetic training samples. Performed the experiments to support the paper claims.

5.3 Paper C: Exploiting Semantic Knowledge for a coherent and efficient recognition

Outline: The complexity of CRF models increases considerably when applied to cluttered scenarios. This implies the utilization of approximate inference methods for retrieving the recognition results, which in some cases supposes a decrease in the recognition success when compared with exact inference solutions. This paper proposes the utilization of Semantic Knowledge to decrease the CRF inference complexity. This knowledge, encoded in an Ontology, is exploited for the generation of hypotheses about the most probable belonging classes of the objects according to their features. For example, a planar, vertical surface could be a wall or a screen, but not a table. Then, these hypotheses are considered by the CRF as the only possible candidate types. The consequence of this is a considerable reduction in the number of possible assignments, decreasing the inference complexity, even enabling exact inference in some cases. Additionally, prior information about the frequency of occurrence of the different object classes is also encoded into the Ontology. This information reveals that, for example, it is more likely to encounter a computer than a couch in an

office environment, while it is quite unlikely to find an ironing table. A modification to the usual CRF formulation is proposed to exploit such source of prior information. The gain in efficiency and coherence by this approach is measured against the UMA-Offices and NYUv2 datasets.

Contribution by the author: Designed the framework for, employing the hypotheses generated by logical inference over the ontology, reduce the complexity of the CRF model. Adapted the CRF formulation to also consider prior information about the frequency of occurrence of the different object types from the Ontology. Evaluated the achieved complexity reduction and enhanced recognition coherence with two different repositories.

5.4 Paper D: UPGMpp library for managing PGMs

Outline: This paper presents the Undirected Probabilistic Graphical Models in C++ (UPGMpp) library, a software package for working with Undirected PGMs, as is the case of CRFs. The library was specially designed and implemented for efficiently tackling the object/room recognition problem. The paper describes how to apply UPGMpp to this issue, and overviews its three main software packages: *base* (implements the functionality for building and managing PGM graphs), *training* (permits the definition of training datasets to tune a PGM), and *inference* (implements algorithms to perform inference queries over PGMs). To show the flexibility and usability of the library, the paper describes the processes needed for training and testing (performing inference) CRFs, including code snippets, and reports the recognition results yielded by the implemented inference methods dealing with information from the NYUv2 repository. Execution time performance is also discussed.

Contribution by the author: Studied the theory behind Undirected PGMs, as well as related libraries and software solutions for dealing with them. Designed and implemented the library packages, with the goal of being efficient, versatile, extensible, and easy to use. Made the library publicly available. Exemplified how to use the library, and measured its success and execution time performance.

5.5 Paper E: OLT toolkit for managing sequential RGB-D datasets

Outline: In this work it is presented the Object Labeling Toolkit (OLT), a set of software components for the efficient labeling of datasets compound of sequences of RGB-D observations, gathered from an arbitrary number of sensors of that type. For that, the toolkit builds a 3D reconstruction of the scene explored in each RGB-D sequence, and allows the user to graphically label objects within that reconstruction.

Once the scene is labeled, such annotations are automatically propagated to each observation in the sequence. The paper describes its main components, namely: *dataset pre-processing*, *2D map building*, *localization of observation poses*, *sequential visualization*, *scene labeling*, and *labels propagation*, of which only *scene labeling* requires a human operator. It is also depicted the toolkit usage for effortlessly labeling two sequences of observations, also analyzing its virtues with respect to a typical labeling approach.

Contribution by the author: Designed the toolkit and its components. Studied and implemented/adapted techniques for processing RGB and depth images, building 2D geometric maps, building 3D reconstructions, visualizing and interacting with reconstructions, and automatically propagating information through a sequence of sensory data. Compared the time saved when employing the toolkit with respect to a typical labeling approach.

5.6 Paper F: Semantic Map representation handling uncertainty

Outline: This paper proposes a semantic map representation that handles uncertainty, also taking advantage of contextual relations among spatial elements (objects and rooms), coined Multiversal Semantic Map (*MvSmap*). The paper reports a comprehensive survey on semantic mapping approaches, as well as on grounding techniques for populating those maps. *MvSmaps* are described in detail and formally defined, along with the algorithms involved in their building, where the recognition techniques presented in previous works play a pivotal role. Moreover, this paper includes algorithms for efficiently tackling the uncertainty modeled by these maps. The novel Robot@Home dataset is used for both, testing the symbol grounding success, as well as illustrating the building of *MvSmaps* from scenarios with different complexity.

Contribution by the author: Designed the Multiversal Semantic Map representation for storing and managing uncertain information. Integrated the previously developed object and room recognition techniques within a symbol grounding process. Designed and implemented the pipeline for building *MvSmaps* according to the information perceived by a mobile robot. Processed the Robot@Home dataset for being useful for testing symbol grounding algorithms, as well as for illustrating the building of *MvSmaps*.

6 Conclusions and future work

Reaching the end of the thesis, it is time to draw conclusions and think about the future.

This thesis has explored and made contributions to the fascinating world of semantic mapping applied to mobile robots. This type of maps aims to provide a robot with *a sense of understanding* of what is going on in its surroundings, which sets the basis for an intelligent, autonomous, and efficient operation. Particular emphasis has been placed on the population of semantic maps with information about the spatial elements in the robot workspace, namely objects and rooms, through the combination of techniques from *Machine Learning* and *Artificial Intelligence*. These fields are at a great point, evidenced by a growing number of studies and successful applications, as recently commented by Ralf Herbrich – Amazon's director of machine learning – *"We're in a golden age of machine learning and AI, ... , as a scientific community, we are still a long way from being able to do things the way humans do things, but we're solving unbelievably complex problems every day and making incredibly rapid progress."*. In the author's opinion, the research of systems exploiting the synergy of these two fields, boosting their advantages and mitigating their limitations, can lead to remarkable advances profitable by the robotic community. That is the case of the techniques developed in this thesis.

In order to be aware of its surroundings, a mobile robot must be able to recognize the elements that are observed through its sensory system. The second chapter of this thesis described the contributions done in this regard, which focused on the combination of *Conditional Random Fields* (CRFs), a discriminative, undirected variant of

Probabilistic Graphical Models (PGMs), and *Semantic Knowledge* of the domain at hand codified in an *Ontology*. These two frameworks have reached a notable success in different classification applications.

CRFs master the modeling of contextual relations among spatial elements, also handling the uncertainty coming from the robot sensory system and the employed models, and supporting the execution of probabilistic inference methods. Precisely, one of the earliest contributions of this thesis was the *Undirected Probabilistic Graphical Models in C++* (UPGMpp) library, developed as a consequence of the lack of software tools for handling Undirected PGMs in general, and CRFs in particular, providing the features demanded by a recognition system running on board of a mobile robot. This library, which is publicly available, implements popular algorithms for building, learning and performing inference over graphical models. The possible choices of training and inference methods for CRFs motivated the thorough study of different learning strategies, in order to find the most successful configuration for the scene object/room recognition problem. This study provided valuable conclusions, not only for the appropriate utilization of these models in the remaining contributions, but also for those in the robotic community aiming to quickly set-up a working-system as successful as possible for such problem.

Despite their successful utilization in different fields, CRFs exhibit a number of shortcomings when applied to recognition. First of all, to be properly tuned, they require a considerable amount of training data comprehensively covering the elements within the domain at hand. The collection of a dataset is a tedious, heavily time-consuming, and (in some domains) unfeasible task, as the author experienced when processing the *UMA-Offices* dataset. Such dataset, consisting of 25 scenes captured by a mobile robot from office facilities within the *University of Málaga*, was collected to evaluate the developed recognition techniques in conjunction with other state-of-the-art repositories containing information from the trending topic sensors, *#RGB-D_cameras*. To avoid the dependency of datasets containing real data, it was shown how Semantic Knowledge, conveniently codified in an Ontology, can be used to effortlessly generate an arbitrary number of training samples representative of the domain at hand. Ontologies provide a natural way to encode Semantic Knowledge, and suppose a compact, human-readable, and ready-to-use representations in high-level reasoning tasks. However, they are unable to handle uncertainty, and it is difficult to fill the gap between the low level sensory data and the codified information without introducing additional ad-hoc processes. Their synergy with CRFs removes these limitations, setting a mutual benefit relationship.

This thesis has exhibited that Ontologies have much to offer to its marriage with CRFs. For example, they have been employed to generate hypotheses about the possible types of the objects/rooms within a scene, drastically reducing in that way the complexity of the CRFs modeling such scene. This increases the efficiency of approximate inference methods over CRFs, also broaden the scenarios where exact inference is feasible. Notice that the efficiency of the recognition method is key for the proper robot operation, since it must share the (usually limited) robot resources with other algorithms in execution like those performing navigation or localization. On-

tologies may encode different types of information about the elements of the domain of discourse, and this has been leveraged to codify the frequency of occurrence of the different object classes. The usual CRF formulation has been accordingly adapted to exploit this source of prior information, allowing these models to achieve more coherent recognition results. Encoded Semantic Knowledge has been also used to detect incoherences in such results, and learn from them in collaboration with a human. This approach overcomes the CRF inability to learn from experience, and permits it to improve its performance and robustness in the long-term operation within home environments.

Once the mobile robot was able to recognize the elements in its surroundings with guarantees, such recognition framework was integrated into a semantic mapping system. For that, it was designed the *Multiversal Semantic Map* (*MvSmap*), a representation of the robot workspace able to accommodate and take advantage of the probabilistic outcome of the developed recognition techniques. This map considers different interpretations of the spatial elements, called *universes*, as instantiations of Ontologies, creating a *multiverse*. These Ontologies are further annotated with the probabilities yielded by the recognition framework, as well as with their probability of being the true one. Thereby, the robot performance is not limited to the utilization of the most probable universe, like traditional semantic maps do, but it can also consider other possible explanations with different semantic interpretations, resulting in a more coherent robot operation. A way to keep the complexity of the multiverse tractable has been also presented, enabling its utilization in complex environments.

Two additional resources related to semantic mapping have been also made public. The first one is a dataset, coined *Robot@Home*, containing among others: 87,000+ time-stamped observations gathered by a mobile robot endowed with a rig of 4 RGB-D cameras and a 2D laser scanner, 3D reconstructions and 2D geometric maps of fully explored houses, topological information about the connectivity of rooms, and ground truth annotations about the type of the surveyed rooms and objects. The dataset is rich in contextual information of the contained spatial elements, a valuable feature missing in most of the state-of-the-art datasets, which can be exploited by semantic mapping systems. The second contribution in this regard is the *Object Labeling Toolkit* (OLT), a set of software components to efficiently process sequences of sensory information, including RGB-D observations. Such components are highly customizable and expandable, facilitating the integration of already-developed algorithms, and have proven to drastically reduce the time and effort needed for processing that type of datasets.

As a final remark, it is worth to say that although all the techniques described in this thesis have been assessed with data repositories from domestic and office environments, their utilization is not restricted to these domains, but they can be exploited in any scenario exhibiting rich semantic information as hospitals, shopping centers, or other human-like environments. Moreover, their use is not restricted to mobile robot applications, but they could be exported to other fields that would benefit from the exploitation of semantic maps as assistance to visual impaired or elderly people, augmented reality, and more applications to appear in the era of portable devices able

to execute this kind of techniques. Nowadays, in fact, our smartphones are almost as powerful as our desktop computers. The research efforts in semantic mapping, along with the new technological advances, ensure the emergence of breakthrough and exciting applications. Stay tuned!

Future work

The work done in this thesis leaves a number of research lines open. Some of the most interesting ones are outlined below.

Hypotheses generation. The generation of hypotheses employing the information encoded in the Ontology could be so restrictive in some situations, mainly with objects showing unusual properties. Let's suppose a scene with a book placed on the floor. In that situation the logical reasoner does not yield the type `Book` as a hypothesis, given that its height largely differs from the expected one. An option could be to consider the result of the logical inference as a score to be introduced in the CRF formulation, at the cost of compromising the exact inference option.

Exploitation of MvSmaps. The real potential of *Multiversal Semantic Maps* (in the author's opinion) is still to come. Several proof-of-concept applications have been designed and tested, but it should be studied the benefits of this representation in real world problems like efficient navigation and object search, robot localization, task planning with uncertain/incomplete information, etc.

Learning from experience. There is significant room to explore possible improvements to the proposed system for learning from experience. Firstly, it should be conducted a thorough evaluation of the system with complex CRFs and ontologies, including information from objects and rooms, during long periods of time. Since the human is in the *learning loop*, it could be also studied how possible incorrect indications by the user affect the performance. The system could also benefit from a study of when would be more appropriate to ask the user about inconsistent results in order to not bother him/her.

Further development of UPGMpp. Some additional features regarding the performance of the UPGMpp library could be explored. For example, although the most time-consuming parts of the library are parallelized through OpenMP, some repetitive operations intensively employing data could also benefit for a parallelization at a lower level, aiming to also take advantage of GPU cores through, for example, CUDA or OpenCL. Visualization tools for inspecting the underlying graphs would be also useful for understanding what is on in the code and during execution. The implementation of sampling techniques for drawing samples from the probability defined by a PGM (like Markov Chain Monte Carlo), are also in the spotlight. Of course, any contribution to UPGMpp from the computer vision or robotic communities is welcome.

Improvements in OLT. The incorporation of algorithms for a globally consistent alignment of the RGB-D observations used to reconstruct a scene would lead to even more accurate models. The user experience could be also improved with the addition of geometric primitives like spheres or cylinders to the currently used one (boxes) to segment and label scenes. Moreover, the time needed for labeling would be reduced if an initial segmentation of the scene as well as tentative labels for the objects/rooms therein are provided beforehand.

Bibliography

[1] B. Abramson, J. Brown, W. Edwards, A. Murphy, and R. L. Winkler. Hailfinder: A bayesian system for forecasting severe weather. *International Journal of Forecasting*, 12(1):57–71, 1996.

[2] Al-Moadhen, A. Abdulhadi, M. Packianather, R. Setchi, and R. Qiu. Robot task planning in deterministic and probabilistic conditions using semantic knowledge base. *International Journal of Knowledge and Systems Science (IJKSS)*, 7(1):56–77, Jan. 2016.

[3] A. Aldoma, T. Faulhammer, and M. Vincze. Automation of "ground truth" annotation for multi-view rgb-d object instance recognition datasets. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pages 5016–5023, Sept 2014.

[4] A. Anand, H. S. Koppula, T. Joachims, and A. Saxena. Contextually guided semantic labeling and search for three-dimensional point clouds. *In the International Journal of Robotics Research*, 32(1):19–34, Jan. 2013.

[5] A. Anand, H. S. Koppula, T. Joachims, and A. Saxena. Contextually guided semantic labeling and search for three-dimensional point clouds. *In The International Journal of Robotics Research*, 32(1):19–34, Jan. 2013.

[6] H. Andreasson, A. Treptow, and T. Duckett. Localization for mobile robots using panoramic vision, local features and particle filter. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 3348–3353, April 2005.

[7] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, New York, NY, USA, 2007.

[8] J. Bai, Y. Wu, J. Zhang, and F. Chen. Subset based deep learning for rgb-d object recognition. *Neurocomputing*, 165(0):280 – 292, 2015.

[9] E. Bastianelli, D. D. Bloisi, R. Capobianco, F. Cossu, G. Gemignani, L. Iocchi, and D. Nardi. On-line semantic mapping. In *Advanced Robotics (ICAR), 2013 16th International Conference on*, pages 1–6, Nov 2013.

[10] S. Bechhofer, F. van Harmelen, J. Hendler, I. Horrocks, D. L. McGuinness, P. F. Patel-Schneider, and L. A. Stein. OWL Web Ontology Language reference. W3C Recommendation, 2004.

[11] J. Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society. Series B (Methodological)*, 48(3):259–302, 1986.

[12] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[13] J. Blanco, J. González, and J.-A. Fernández-Madrigal. Subjective local maps for hybrid metric-topological {SLAM}. *Robotics and Autonomous Systems*, 57(1):64 – 74, 2009.

[14] J.-L. Blanco, J.-A. Fernández-Madrigal, and J. González-Jiménez. Towards a unified bayesian approach to hybrid metric-topological slam. *IEEE Transactions on Robotics*, 24(2):259–270, 2008.

[15] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(11):1222–1239, Nov 2001.

[16] L. Chang, M. M. Duarte, L. Sucar, and E. F. Morales. A bayesian approach for object classification based on clusters of SIFT local features. *Expert Systems with Applications*, 39(2):1679 – 1686, 2012.

[17] Cognitum. Fluent editor home page. http://www.cognitum.eu/semantics/FluentEditor/, 2016. [Online; accessed 16-September-2016].

[18] S. Coradeschi and A. Saffiotti. An introduction to the anchoring problem. *Robotics and Autonomous Systems*, 43(2-3):85–96, 2003.

[19] S. Divvala, D. Hoiem, J. Hays, A. Efros, and M. Hebert. An empirical study of context in object detection. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1271–1278, June 2009.

[20] F. Dornaika, A. Bosaghzadeh, H. Salmane, and Y. Ruichek. Graph-based semi-supervised learning with local binary patterns for holistic object categorization. *Expert Systems with Applications*, 41(17):7744 – 7753, 2014.

[21] N. Durand, S. Derivaux, G. Forestier, C. Wemmert, P. Gancarski, O. Boussaid, and A. Puissant. Ontology-based object recognition for remote sensing image interpretation. In *Tools with Artificial Intelligence, 2007. ICTAI 2007. 19th IEEE International Conference on*, volume 1, pages 472–479, Oct 2007.

[22] S. Ekvall, D. Kragic, and P. Jensfelt. Object detection and mapping for service robot tasks. *Robotica*, 25(2):175–187, Mar. 2007.

[23] A. Elfes. Sonar-based real-world mapping and navigation. *IEEE Journal on Robotics and Automation*, 3(3):249–265, June 1987.

[24] G. Elidan, I. McGraw, and D. Koller. Residual belief propagation: Informed scheduling for asynchronous message passing. In *Proceedings of the Twenty-second Conference on Uncertainty in AI (UAI)*, Boston, Massachussetts, July 2006.

[25] N. Eric Maillot and M. Thonnat. Ontology based complex object recognition. *Image Vision Comput.*, 26(1):102–113, Jan. 2008.

[26] P. Espinace, T. Kollar, A. Soto, and N. Roy. Indoor scene recognition through object detection. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 1406–1413. IEEE, 2010.

[27] M. Everingham, L. van Gool, C. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010.

[28] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, 32(9):1627–1645, 2010.

[29] R. Fergus, L. Fei-Fei, P. Perona, and A. Zisserman. Learning object categories from google's image search. In *IEEE International Conference on Computer Vision (ICCV 2005)*, volume 2, pages 1816–1823 Vol. 2, 2005.

[30] E. Fernandez-Moral, J. González-Jiménez, P. Rives, and V. Arévalo. Extrinsic calibration of a set of range cameras in 5 seconds without pattern. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2014)*, Chicago, USA, September 2014.

[31] E. Fernandez-Moral, W. Mayol-Cuevas, V. Arevalo, and J. Gonzalez-Jimenez. Fast place recognition with plane-based maps. In *IEEE International Conference on Robotics and Automation (ICRA 2013)*, pages 2719–2724, 2013.

[32] T. Finley and T. Joachims. Training structural svms when exact inference is intractable. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pages 304–311, New York, NY, USA, 2008. ACM.

[33] G. Floros and B. Leibe. Joint 2d-3d temporally consistent semantic segmentation of street scenes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2012)*, pages 2823–2830, 2012.

[34] C. Galindo, J. Fernandez-Madrigal, J. Gonzalez, and A. Saffiotti. Using semantic information for improving efficiency of robot task planning. In *IEEE International Conference on Robotics and Automation (ICRA), Workshop on Semantic Information in Robotics*, Rome, Italy, 2007.

[35] C. Galindo, J. Fernandez-Madrigal, J. Gonzalez, and A. Saffiotti. Robot task planning using semantic maps. *Robotics and Autonomous Systems*, 56(11):955–966, 2008.

[36] C. Galindo and A. Saffiotti. Inferring robot goals from violations of semantic knowledge. *Robotics and Autonomous Systems*, 61(10):1131–1143, 2013.

[37] C. Galindo, A. Saffiotti, S. Coradeschi, P. Buschka, J. A. Fernandez-Madrigal, and J. Gonzalez. Multi-hierarchical semantic maps for mobile robotics. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2278–2283, Aug 2005.

[38] C. Galleguillos and S. Belongie. Context based object categorization: A critical survey. *Computer Vision and Image Understanding*, 114(6):712–722, June 2010.

[39] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, J. Garcia-Rodriguez, J. Azorin-Lopez, M. Saval-Calvo, and M. Cazorla. Multi-sensor 3d object dataset for object recognition with full pose estimation. *Neural Computing and Applications*, pages 1–12, 2016.

[40] G. Gemignani, D. Nardi, D. D. Bloisi, R. Capobianco, and L. Iocchi. Interactive semantic mapping: Experimental evaluation. In A. M. Hsieh, O. Khatib, and V. Kumar, editors, *Experimental Robotics: The 14th International Symposium on Experimental Robotics*, volume 109 of *Springer Tracts in Advanced Robotics*, pages 339–355. Springer International Publishing, 2016.

[41] B. Glimm, I. Horrocks, B. Motik, G. Stoilos, and Z. Wang. Hermit: an owl 2 reasoner. *Journal of Automated Reasoning*, 53(3):245–269, 2014.

[42] R. Gómez-Ojeda, J. Briales, E. Fernández-Moral, and J. González-Jiménez. Extrinsic calibration of a 2d laser-rangefinder and a camera based on scene corners. In *IEEE International Conference on Robotics and Automation (ICRA)*, Seattle, USA, 2015.

[43] R. Gonçalves, M. Horridge, M. Musen, C. Nyulas, S. Tu, and T. Tudorache. Protégè home page. http://protege.stanford.edu/, 2015. [Online; accessed 26-June-2015].

[44] G. Guennebaud, B. Jacob, et al. Eigen v3. http://eigen.tuxfamily.org, 2010.

[45] M. Günther, T. Wiemann, S. Albrecht, and J. Hertzberg. Building semantic object maps from sparse and noisy 3d data. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2013)*, pages 2228–2233, 2013.

[46] R. Gupta and M. J. Kochenderfer. Common sense data acquisition for indoor mobile robots. In *Proceedings of the 19th National Conference on Artifical Intelligence*, AAAI'04, pages 605–610. AAAI Press, 2004.

[47] V. Haarslev, K. Hidde, R. Möller, and M. Wessel. The racerpro knowledge representation and reasoning system. *Semantic Web Journal*, 3(3):267–277, 2012.

[48] J. Hammersley and P. Clifford. Markov fields on finite graphs and lattices, 1971. Unpublished manuscript.

[49] S. Harnad. The symbol grounding problem. *Phys. D*, 42(1-3):335–346, June 1990.

[50] K. Held, E. R. Kops, B. J. Krause, W. M. Wells, R. Kikinis, and H.-W. Muller-Gartner. Markov random field segmentation of brain mr images. *IEEE transactions on medical imaging*, 16(6):878–886, 1997.

[51] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab. Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *Proceedings of the 11th Asian Conference on Computer Vision - Volume Part I*, ACCV'12, pages 548–562, Berlin, Heidelberg, 2013. Springer-Verlag.

[52] W. L. Hoo, C. H. Lim, and C. S. Chan. Keybook: Unbias object recognition using keywords. *Expert Systems with Applications*, 42(8):3991 – 3999, 2015.

[53] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean. SWRL: A semantic web rule language combining OWL and RuleML. W3C Member Submission, World Wide Web Consortium, 2004.

[54] F. Husain, L. Dellen, and C. Torras. Recognizing point clouds using conditional random fields. In *Pattern Recognition (ICPR), 2014 22nd International Conference on*, pages 4257–4262, Aug 2014.

[55] A. Hyvärinen. Estimation of non-normalized statistical models by score matching. *The Journal of Machine Learning Research*, 6:695–709, Dec. 2005.

[56] J. Jancsary, S. Nowozin, T. Sharp, and C. Rother. Regression tree fields - an efficient, non-parametric approach to image labeling problems. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2012)*, pages 2376–2383, 2012.

[57] A. Janoch, S. Karayev, Y. Jia, J. T. Barron, M. Fritz, K. Saenko, and T. Darrell. A category-level 3-d object dataset: Putting the kinect to work. In *1st Workshop on Consumer Depth Cameras for Computer Vision (ICCV workshop)*, November 2011.

[58] J.L. Blanco Claraco. Mobile Robot Programming Toolkit (MRPT). `http://www.mrpt.org`, 2015. [Online; accessed 28-April-2015].

[59] A. Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. *Advances in neural information processing systems*, 14:841, 2002.

[60] O. Kahler and I. Reid. Efficient 3d scene labeling using fields of trees. In *2013 IEEE International Conference on Computer Vision*, pages 3064–3071, Dec 2013.

[61] L. Karlsson. Conditional progressive planning under uncertainty. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence - Volume 1*, IJCAI'01, pages 431–436, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc.

[62] A. Kasper, Z. Xue, and R. Dillmann. The kit object models database: An object model database for object recognition, localization and manipulation in service robotics. *The International Journal of Robotics Research*, 31(8):927–934, 2012.

[63] R. Kindermann, J. L. Snell, et al. *Markov random fields and their applications*, volume 1. American Mathematical Society Providence, RI, 1980.

[64] J. Knopp, M. Prasad, G. Willems, R. Timofte, and L. Van Gool. Hough transform and 3d surf for robust three dimensional classification. In *Proceedings of the 11th European Conference on Computer Vision: Part VI*, ECCV'10, pages 589–602, Berlin, Heidelberg, 2010. Springer-Verlag.

[65] D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.

[66] F. Korč and W. Förstner. Approximate parameter learning in conditional random fields: An empirical investigation. In *Proceedings of the 30th DAGM Symposium on Pattern Recognition*, pages 11–20, Berlin, Heidelberg, 2008. Springer-Verlag.

[67] I. Kostavelis and A. Gasteratos. Semantic mapping for mobile robotics tasks: A survey. *Robotics and Autonomous Systems*, 66:86–103, 2015.

[68] S. Kumar, J. August, and M. Hebert. Exploiting inference for approximate parameter learning in discriminative fields: An empirical study. In *Proceedings of the 5th International Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition*, EMMCVPR'05, pages 153–168, Berlin, Heidelberg, 2005. Springer-Verlag.

[69] S. Kumar and M. Hebert. Discriminative random fields. *Int. J. Comput. Vision*, 68(2):179–201, June 2006.

[70] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc.

[71] K. Lai, L. Bo, and D. Fox. Unsupervised feature learning for 3d scene labeling. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 3050–3057, May 2014.

[72] 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*, pages 1817–1824, May 2011.

[73] D. Lin, S. Fidler, and R. Urtasun. Holistic scene understanding for 3d object detection with rgbd cameras. *IEEE International Conference on Computer Vision*, 0:1417–1424, 2013.

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

[75] J. Martinez-Gomez, M. Cazorla, I. Garcia-Varea, and V. Morell. Vidrilo: The visual and depth robot indoor localization with objects information dataset. *International Journal of Robotics Research*, 2015.

[76] D. Meger, P.-E. Forssén, K. Lai, S. Helmer, S. McCann, T. Southey, M. Baumann, J. J. Little, and D. G. Lowe. Curious george: An attentive semantic robot. *Robots and Autonomous Systems*, 56(6):503–511, June 2008.

[77] D. Meger and J. J. Little. The UBC visual robot survey: A benchmark for robot category recognition. In *Experimental Robotics - The 13th International Symposium on Experimental Robotics, ISER 2012, June 18-21, 2012, Québec City, Canada*, pages 979–991, 2012.

[78] M. L. Mekhalfi, F. Melgani, Y. Bazi, and N. Alajlan. Toward an assisted indoor scene perception for blind people with image multilabeling strategies. *Expert Systems with Applications*, 42(6):2907 – 2918, 2015.

[79] R. Mottaghi, A. Ranganathan, and A. L. Yuille. A compositional approach to learning part-based models of objects. In *IEEE International Conference on Computer Vision Workshops (ICCV 2011 Workshops)*, pages 561–568, 2011.

[80] O. Mozos, C. Stachniss, and W. Burgard. Supervised learning of places from range data using adaboost. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 1730–1735, April 2005.

[81] A. C. Murillo, J. J. Guerrero, and C. Sagues. Surf features for efficient robot localization with omnidirectional images. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 3901–3907, April 2007.

[82] J. N. N. Okazaki. libLBFGS: a library of Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS). `http://www.chokkan.org/software/liblbfgs/`, 2015. [Online; accessed 14-September-2015].

[83] Y. Nesterov. *Introductory lectures on convex optimization : a basic course*. Applied optimization. Springer US, 2004.

[84] D. Nikovski. Constructing bayesian networks for medical diagnosis from incomplete and partially correct statistics. *IEEE Transactions on Knowledge and Data Engineering*, 12(4):509–516, 2000.

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

[86] J. Nocedal. Updating quasi-newton matrices with limited storage. *Mathematics of computation*, 35(151):773–782, 1980.

[87] S. Nowozin, C. Rother, S. Bagon, T. Sharp, B. Yao, and P. Kohli. Decision tree fields. In *IEEE International Conference on Computer Vision (ICCV 2011)*, pages 1668–1675, 2011.

[88] A. Nüchter and J. Hertzberg. Towards semantic maps for mobile robots. *Robots and Autonomous Systems*, 56(11):915–926, 2008.

[89] N. Okazaki. Crfsuite: a fast implementation of conditional random fields (crfs). http://www.chokkan.org/software/crfsuite/. [Online; accessed 28-April-2015].

[90] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International journal of computer vision*, 42(3):145–175, 2001.

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

[92] A. Oliva and A. Torralba. The role of context in object recognition. *Trends in Cognitive Sciences*, 11(12):520–527, Dec. 2007.

[93] M. Oliveira, L. S. Lopes, G. H. Lim, S. H. Kasaei, A. D. Sappa, and A. M. Tomé. Concurrent learning of visual codebooks and object categories in open-ended domains. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 2488–2495, Sept 2015.

[94] OpenMP Architecture Review Board: OpenMP API Specification for Parallel Programming. `http://openmp.org/wp/`. [Online; accessed 14-April-2016].

[95] D. Pangercic, B. Pitzer, M. Tenorth, and M. Beetz. Semantic object maps for robotic housework - representation, acquisition and use. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4644–4651, Oct 2012.

[96] S. Parise and M. Welling. Learning in markov random fields: An empirical study. In *Proceedings of the Joint Statistical Meeting*, JSM2005, 2005.

[97] S. Payr, F. Werner, and K. Werner. Potential of robotics for ambient assisted living. Technical report, Austrian Research Institute for Artificial Intelligence, 2015.

[98] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1988.

[99] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, June 2007.

[100] M. Pontil and A. Verri. Support vector machines for 3d object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(6):637–646, Jun 1998.

[101] A. Pronobis and B. Caputo. Confidence-based cue integration for visual place recognition. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2394–2401, Oct 2007.

[102] A. Pronobis and P. Jensfelt. Hierarchical multi-modal place categorization. In *European Conference on Mobile Robots (ECMR)*, pages 159–164, 2011.

[103] A. Pronobis and P. Jensfelt. Large-scale semantic mapping and reasoning with heterogeneous modalities. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 3515–3522, May 2012.

[104] A. Pronobis, P. Jensfelt, K. Sjöö, H. Zender, G.-J. M. Kruijff, O. M. Mozos, and W. Burgard. Semantic modelling of space. In H. I. Christensen, G.-J. M. Kruijff, and J. L. Wyatt, editors, *Cognitive Systems*, volume 8 of *Cognitive Systems Monographs*, pages 165–221. Springer Berlin Heidelberg, 2010.

[105] A. Pronobis, O. M. Mozos, B. Caputo, and P. Jensfelt. Multi-modal semantic place classification. *The International Journal of Robotics Research*, 2009.

[106] A. Quattoni, M. Collins, and T. Darrell. Conditional random fields for object recognition. In *Advances in Neural Information Processing Systems*, pages 1097–1104. MIT Press, 2004.

[107] A. Quattoni and A. Torralba. Recognizing indoor scenes. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 413–420. IEEE, 2009.

[108] A. Ranganathan and F. Dellaert. Semantic modeling of places using objects. In *Robotics: Science and Systems Conference III (RSS)*. MIT Press, 2007.

[109] A. Ranganathan, E. Menegatti, and F. Dellaert. Bayesian inference in the space of topological maps. *IEEE Transactions on Robotics*, 22(1):92–107, Feb 2006.

[110] E. Remolina and B. Kuipers. Towards a general theory of topological maps. *Artificial Intelligence*, 152(1):47–104, Jan. 2004.

[111] X. Ren, L. Bo, and D. Fox. Rgb-(d) scene labeling: Features and algorithms. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2012)*, pages 2759–2766, 2012.

[112] L. Riazuelo, M. Tenorth, D. D. Marco, M. Salas, D. Gálvez-López, L. Mösenlechner, L. Kunze, M. Beetz, J. D. Tardós, L. Montano, and J. M. M. Montiel. Roboearth semantic mapping: A cloud enabled knowledge-based approach. *IEEE Transactions on Automation Science and Engineering*, 12(2):432–443, April 2015.

[113] J. G. Rogers and H. I. Christensen. A conditional random field model for place and object classification. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 1766–1772, May 2012.

[114] J. Ruiz-Sarmiento, C. Galindo, and J. González-Jiménez. Mobile robot object recognition through the synergy of probabilistic graphical models and semantic knowledge. In *European Conf. on Artificial Intelligence. Workshop on Cognitive Robotics*, 2014.

[115] J. Ruiz-Sarmiento, C. Galindo, and J. González-Jiménez. UPGMpp: a Software Library for Contextual Object Recognition. In *3rd. Workshop on Recognition and Action for Scene Understanding*, 2015.

[116] J. R. Ruiz-Sarmiento, C. Galindo, and J. González-Jiménez. Exploiting semantic knowledge for robot object recognition. *Knowledge-Based Systems*, 86:131–142, 2015.

[117] J. R. Ruiz-Sarmiento, C. Galindo, and J. González-Jiménez. Joint categorization of objects and rooms for mobile robots. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015.

[118] J. R. Ruiz-Sarmiento, C. Galindo, and J. González-Jiménez. OLT: A Toolkit for Object Labeling Applied to Robotic RGB-D Datasets. In *European Conference on Mobile Robots*, 2015.

[119] J. R. Ruiz-Sarmiento, C. Galindo, and J. González-Jiménez. Scene object recognition for mobile robots through semantic knowledge and probabilistic graphical models. *Expert Systems with Applications*, 42(22):8805–8816, 2015.

[120] J. R. Ruiz-Sarmiento, C. Galindo, and J. González-Jiménez. Building Multiversal Semantic Maps for Mobile Robot Operation. *Submitted*, 2016.

[121] J. R. Ruiz-Sarmiento, C. Galindo, and J. González-Jiménez. A survey on learning approaches for undirected graphical models. Application to scene object recognition. *International Journal of Approximate Reasoning (Aceppted)*, 2016.

[122] J. R. Ruiz-Sarmiento, C. Galindo, and J. González-Jiménez. Probability and common-sense: Tandem towards robust robotic object recognition in ambient assisted living. *10th International Conference on Ubiquitous Computing and Ambient Intelligence*, 2016.

[123] J. R. Ruiz-Sarmiento, C. Galindo, and J. González-Jiménez. Robot@home, a robotic dataset for semantic mapping of home environments. *Submitted*, 2016.

[124] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, and L. Fei-Fei. Imagenet large scale visual recognition challenge. *CoRR*, abs/1409.0575, 2014.

[125] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. Labelme: A database and web-based tool for image annotation. *Int. J. Comput. Vision*, 77(1-3):157–173, May 2008.

[126] R. B. Rusu and S. Cousins. 3D is here: Point Cloud Library (PCL). In *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, May 9-13 2011.

[127] R. B. Rusu, Z. C. Marton, N. Blodow, M. Dolha, and M. Beetz. Towards 3d point cloud based object maps for household environments. *Robotics and Autonomous Systems*, 56(11):927 – 941, 2008. Semantic Knowledge in Robotics.

[128] B. Schling. *The Boost C++ Libraries*. XML Press, 2011.

[129] M. Schmidt. UGM: Matlab Code for Undirected Graphical Models. `http://www.cs.ubc.ca/~schmidtm/Software/UGM.html`, 2015. [Online; accessed 28-April-2015].

[130] N. Silberman and R. Fergus. Indoor scene segmentation using a structured light sensor. In *Proceedings of the International Conf. on Computer Vision - Workshop on 3D Representation and Recognition*, 2011.

[131] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor Segmentation and Support Inference from RGBD Images. In *Proc. of the 12th European Conference on Computer Vision (ECCV 2012)*, pages 746–760, 2012.

[132] A. Singh, J. Sha, K. Narayan, T. Achim, and P. Abbeel. Bigbird: A large-scale 3d database of object instances. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 509–516, May 2014.

[133] E. Sirin, B. Parsia, B. C. Grau, A. Kalyanpur, and Y. Katz. Pellet: A practical owl-dl reasoner. *Web Semantics: Science, Services and Agents on the World Wide Web*, 5(2):51–53, June 2007.

[134] R. Speer and C. Havasi. Conceptnet 5: a large semantic network for relational knowledge. In *The People's Web Meets NLP. Theory and Applications of Natural Language*, pages 161—176. Springer, 2013.

[135] C. Sutton and A. McCallum. Piecewise pseudolikelihood for efficient training of conditional random fields. In *Proceedings of the 24th international conference on Machine learning*, pages 863–870. ACM, 2007.

[136] C. A. Sutton and A. Mccallum. Piecewise Training for Undirected Models. In *Proceedings of the 21st Conference on Uncertainty in Artificial Intelligence (UAI-05)*, pages 568–575, 2005.

[137] A. Teichman, S. Miller, and S. Thrun. Unsupervised intrinsic calibration of depth sensors via slam. In *Proceedings of Robotics: Science and Systems*, Berlin, Germany, June 2013.

[138] M. Tenorth, L. Kunze, D. Jain, and M. Beetz. Knowrob-map - knowledge-linked semantic object maps. In *2010 10th IEEE-RAS International Conference on Humanoid Robots*, pages 430–435, Dec 2010.

[139] S. Thrun. Learning metric-topological maps for indoor mobile robot navigation. *Artificial Intelligence*, 99(1):21 – 71, 1998.

[140] S. Thrun. Learning occupancy grid maps with forward sensor models. *Autonomous Robots*, 15(2):111–127, Sept. 2003.

[141] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. Intelligent robotics and autonomous agents. MIT Press, 2005.

[142] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin. Context-based vision system for place and object recognition. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 273–280. IEEE, 2003.

[143] D. Tsarkov and I. Horrocks. *FaCT++ Description Logic Reasoner: System Description*, pages 292–297. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[144] M. Uschold and M. Gruninger. Ontologies: principles, methods and applications. *The Knowledge Engineering Review*, 11:93–136, 1996.

[145] J. Valentin, S. Sengupta, J. Warrell, A. Shahrokni, and P. Torr. Mesh based semantic modelling for indoor and outdoor scenes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2013)*, pages 2067–2074, 2013.

[146] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001)*, volume 1, pages 511–518, 2001.

[147] P. Viswanathan, T. Southey, J. Little, and A. Mackworth. Place classification using visual object categorization and global information. In *Computer and Robot Vision (CRV), 2011 Canadian Conference on*, pages 1–7. IEEE, 2011.

[148] M. Wainwright, T. Jaakkola, and A. Willsky. Tree-based reparameterization framework for analysis of sum-product and related algorithms. *IEEE Transactions on Information Theory*, 49(5):1120–1146, May 2003.

[149] C. Weiss, H. Tamimi, A. Masselli, and A. Zell. A hybrid approach for vision-based outdoor robot localization using global and local image features. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1047–1052, Oct 2007.

[150] Y. Weiss and W. T. Freeman. On the optimality of solutions of the max-product belief-propagation algorithm in arbitrary graphs. *IEEE Trans. Inf. Theor.*, 47(2):736–744, Sept. 2006.

[151] D. Wolf, J. Prankl, and M. Vincze. Fast semantic segmentation of 3d point clouds using a dense crf with learned parameters. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, Seattle, WA, USA, 2015.

[152] Y. Xiang, X. Zhou, Z. Liu, T.-S. Chua, and C.-W. Ngo. Semantic context modeling with maximal margin conditional random fields for automatic image annotation. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3368–3375, 2010.

[153] J. Xiao, A. Owens, and A. Torralba. Sun3d: A database of big spaces reconstructed using sfm and object labels. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 1625–1632, Dec 2013.

[154] X. Xiong and D. Huber. Using context to create semantic 3d models of indoor environments. In *In Proceedings of the British Machine Vision Conference (BMVC 2010)*, pages 45.1–11, 2010.

[155] J. S. Yedidia, W. T. Freeman, and Y. Weiss. Generalized Belief Propagation. In *Advances Neural Information Processing Systems*, volume 13, pages 689–695, 2001.

[156] H. Zender, O. M. Mozos, P. Jensfelt, G.-J. Kruijff, and W. Burgard. Conceptual spatial representations for indoor mobile robots. *Robotics and Autonomous Systems*, 56(6):493 – 502, 2008. From Sensors to Human Spatial Concepts.

[157] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. In *2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06)*, pages 13–13, June 2006.

[158] K. Zhou, M. Zillich, H. Zender, and M. Vincze. Web mining driven object locality knowledge acquisition for efficient robot behavior. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2012)*, pages 3962–3969, 2012.

[159] M. Zou and S. D. Conzen. A new dynamic bayesian network (dbn) approach for identifying gene regulatory networks from time course microarray data. *Bioinformatics*, 21(1):71–79, 2005.

Part II

Included papers

Exploiting Semantic Knowledge for Robot Object Recognition

Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, Javier Gonzalez-Jimenez

Published in Knowledge-Based Systems, 2015.

Exploiting Semantic Knowledge for Robot Object Recognition

J.R. Ruiz-Sarmiento, C. Galindo and J. Gonzalez-Jimenez

Machine Perception and Intelligent Robotics Group, System Engineering and Auto. Dept., University of Málaga, Campus de Teatinos, 29071, Málaga, Spain.

This paper presents a novel approach that exploits semantic knowledge to enhance the object recognition capability of autonomous robots. Semantic knowledge is a rich source of information, naturally gathered from humans (elicitation), which can encode both objects' geometrical/appearance properties and contextual relations. This kind of information can be exploited in a variety of robotics skills, especially for robots performing in human environments. In this paper we propose the use of semantic knowledge to eliminate the need of collecting large datasets for the training stages required in typical recognition approaches. Concretely, semantic knowledge encoded in an ontology is used to synthetically and effortless generate an arbitrary number of training samples for tuning Probabilistic Graphical Models (PGMs). We then employ these PGMs to classify patches extracted from 3D point clouds gathered from office environments within the UMA-offices dataset, achieving a $\sim 90\%$ of recognition success, and from office and home scenes within the NYU2 dataset, yielding a success of $\sim 81\%$ and $\sim 69.5\%$ respectively. Additionally, a comparison with state-of-the-art recognition methods also based on graphical models has been carried out, revealing that our semantic-based training approach can compete with, and even outperform, those trained with a considerable number of real samples.

Keywords: Semantic Knowledge, Human Elicitation, Object Recognition, Probabilistic Graphical Models, Autonomous Robots

1 Introduction

Object recognition is one of the key abilities of a mobile robot intended to perform high-level tasks in human environments, where objects are usually placed according to their functionality, e.g., tv-sets are in front of couches, night tables are near beds, etc. As reported by other authors [11], the exploitation of these contextual relations, that can be seen as a form of *semantic knowledge*, can improve the performance of traditional object recognition methods which only rely on sensorial features.

To illustrate the benefits of using semantics, let's consider a robot coping with the task of recognizing the objects placed in its surroundings. This may become complex for a number of reasons, including the large number of possible object classes and features to extract, their similarity, etc. Suppose now that the robot knows that it is in an office and has some semantic knowledge related to that particular domain, for example the type of objects usually present in a typical office environment and their

contextual relations. This simplifies the recognition problem, drastically reducing the range of possible objects classes, and even more importantly, enabling the recognition system to exploit particular object relations to gain in effectiveness and robustness. For instance, an object that resembles an office table according to its geometry can be more confidently recognized as such if objects typically found near it, e.g. a computer screen and/or a chair, are also detected and fulfill certain contextual relations, for example, the computer screen is on the table and the chair is close to it.

In this work we present a novel approach that exploits semantic knowledge encoded by *human elicitation* to train *Probabilistic Graphical Models* (PGMs) [16] for object recognition. PGMs form a machine learning framework that is widely applied to object recognition given its capabilities for modelling both uncertainty and objects relations. These systems need a vast amount of training data in order to reliably encode the gist of the domain at hand, however, the gathering of that information is an arduous, time-consuming, and – in some domains – not a tractable task. To face this issue, we codify semantic knowledge by means of an ontology [30], which defines the domain object classes, their properties, and their relations, and use it to generate training samples for a Conditional Random Field (CRF) [16]. These training samples reify prototypal scenarios where objects are represented by a set of geometric primitives, e.g., planar patches or bounding boxes, that fulfill certain geometric properties and relations, like proximity, difference of orientation, etc.

Aiming to show the performance of CRFs trained with the proposed approach, they have been integrated into an object recognition framework. This framework operates by processing point clouds provided by a RGB-D camera, in order to extract geometric primitives (see figure A.1-a), which are then recognized as belonging to a certain object class through an inference process over the trained CRF. We have obtained promising results in office and home environments, employing both planar patches and bounding boxes as geometric primitives, though our methodology can be applied to other scenarios and sensorial data types.

In the literature, PGMs are used, in general, to learn the properties of the different object classes and their contextual relations using data from previously collected datasets. In contrast, the work presented here drives this learning phase by providing synthetic training samples extracted from the semantic knowledge of the domain at hand. This knowledge can be naturally provided by humans and encoded into an ontology, and exhibits three advantages with respect to other related approaches:

- It eliminates the usually complex and high resource-consuming task of collecting the large number of training samples required to tune an accurate and comprehensive model of the domain.
- Ontologies are compact and human-readable knowledge representations. In that way, extending the problem with additional object classes is just reduced to codify the knowledge about the new classes into the ontology, generate synthetic samples considering the updated semantic information, and train the CRF. This process can be completed in a few minutes, in contrast to the time needed for gathering and processing real data.

- The recognized objects are anchored to semantically defined concepts, which is useful for robot high-level tasks like reasoning or task planning [10, 8, 4].

We have conducted an evaluation of our work employing two datasets: one from our facilities, called UMA-offices, which counts 25 office environments, and the NYU2 dataset [28], from which we have extracted 61 offices and 200 home scenes. The performance of CRFs trained with our methodology have been also compared with two state-of-the-art methods, namely i) a standard formulation of CRFs trained and tested with real data [16], and ii) the CRF presented in [34]. The results show that our approach can compete with, and even outperform, those trained with a considerable number of real samples.

In the next section we put our proposal in the context of other related works. Section 3 introduces probabilistic graphical models applied to object recognition, while in section 4 we present the proposed method to train these models using semantic knowledge. In section 5, the evaluation results of the method considering two datasets comprising office and home environments are shown, and a comparison with other state-of-the-art approaches is presented. Finally, section 6 ends with some conclusions and future work.

2 Related work

Object recognition is a key topic in robotics and computer vision that, in many cases, has been successfully addressed by *only* using the visual features of isolated objects, i.e. without considering information from the rest of the scene. Some remarkable examples are the Viola and Jones boosted cascade of classifiers [32], the SIFT object recognition algorithm [19] or the Bag of Features [21] models. However, the current trend also considers the exploitation of contextual information between objects, aiming to improve the recognition results (see [11]).

Throughout this section, we discuss related works on object recognition systems that resort to graphical models or semantic knowledge to model contextual information. Also, some works reporting different alternatives to the utilization of ontologies as a source of semantic information for object recognition are commented.

2.1 Probabilistic Graphical Models

Probabilistic Graphical Models (PGMs) [16] is one of the most resorted frameworks to manage contextual information. The earliest works using this tool for object recognition are based on intensity information of the scene, like [33], where the context between pixels in a given RGB image is modelled by a discriminative Conditional Random Field (CRF). Another work also relying on intensity images is the presented in [25] that proposes a CRF framework which incorporates hidden variables for part-based object recognition. The work in [20] also builds part-based models of objects, and represents their interrelations with a PGM. More recent is the work

presented in [7] which employs stereo intensity images in a CRF formulation. Three-dimensional information from stereo enables the exploitation of meaningful geometric properties of objects and relations. However, stereo systems are unable to perform on surfaces/objects showing an uniform intensity, which can negatively affect the recognition performance.

With the emergence of inexpensive 3D sensors, like Kinect, a new batch of approaches have appeared leveraging the dense and relatively accurate data provided by these devices. For example, the work presented in [1] builds a model isomorphic to a Markov Random Field (MRF) according to the segmented regions from a scene point cloud and their relations. The authors did the tedious work of gathering information from 24 office and 28 home environments, and manually labelled the different object classes. Interestingly, it is shown in [26] that the accuracy of a MRF in charge of assigning object classes to a set of superpixels increases as the amount of available training data augments. In [31] a meshed representation of the scene is built on the basis of a number of depth estimates, and a CRF is defined to classify mesh faces. CRFs are also used in [15] and [34], where Decision Tree Fields [23] and Regression Tree Fields [14] are studied as a source of potentials for the PGM. The CRF structure for representing the scenes in [34] is similar to the one presented here. In that work, a CRF is used to classify the main components of a facility, namely clutters, walls, floors and ceilings.

All the methods mentioned above require the collection of large datasets that adequately capture the variability of the domain, which can be a tedious, repetitive, and time-consuming task that consists of moving the robot from one scene to another, gathering the data, and post-processing it accordingly to the type of information expected by the training algorithms. The claim of this work is the utilization of semantic knowledge codified into an ontology as a valuable source of information for the generation of synthetic training samples that, being representative of the domain, also can capture its variability.

2.2 Semantic Knowledge

In the literature, some alternatives to PGMs for object context modelling have been also reported. For example, in [12] a system relying on an ontology plus rules defined into the Semantic Web Rule Language [13] is used to generate object hypotheses. These hypotheses are subsequently checked in a matching process with CAD models. Another example is [24], where a constraint network implemented in Prolog classifies the main structural surfaces, i.e. walls, floors, ceilings and doors, using contextual relations like orthogonal, parallel, above, etc. Nevertheless, these methodologies are unable to handle uncertainty, and exhibit difficulties to leverage all the potential of the contextual relations.

2.3 Alternative sources of information

Additionally to the use of semantic knowledge, other sources of information can be also considered to codify and manage the knowledge from a given domain. For example, in [35], a web mining knowledge acquisition system is presented as a mechanism to obtain information about the location of objects. In [5] the authors describe PGMs that are trained with images from the Google's image search engine. They reported that the high percentage of low quality search results (e.g images where the object of interest appears occluded or is missing, cartoons instead of real objects, etc.) represents a serious drop in the recognition performance. Knowledge bases, like ConceptNet [29], and language models, like TypeDM [2], have been also studied for visual recognition tasks in [18], concluding that they can be inconsistent with the expectation of the presence of objects in the real world if insufficient objects and/or relations are included. Another example of exploitation of encoded information about objects' relations is [17], where the search of a given object is directed by a previously learnt Gaussian Mixture Model (GMM).

In comparison with those methods, the codification of the domain knowledge through human elicitation as presented in this work enables a truly and effortless encoding of a large number of objects' features and relations between them. Moreover, since the source of semantic information (a person or a group of people) is trustworthy, in contrast to online search or web mining-engine based methodologies, there is less uncertainty about the validity of the information being managed. This enables the use of such a semantic information for generating training data which is well representative of the domain. In addition, the use of an ontology to structure that knowledge permits the robot to take advantage of it for other high level applications [9, 10].

3 Scene object recognition through Conditional Random Fields

Conditional Random Fields (CRF) [16] are a particular case of Probabilistic Graphical Models that relies on conditional probability distributions. When applied to object recognition, a CRF computes the posterior $P(\mathbf{y}|\mathbf{x})$, where $\mathbf{x} = [x_1, x_2, ..., x_n]$ are observations of elements in the scene, and $\mathbf{y} = [y_1, y_2, \ldots, y_n]$ are random variables representing the classes of these elements from the set L of the possible object classes. Figure A.1-b shows an example where $L = \{computer\_screen, table, chair\_back, chair\_rest, floor, wall\}$.

The posterior $P(\mathbf{y}|\mathbf{x})$ can be calculated by computing the probability of each possible assignation to the variables in $\mathbf{y}$ conditioned to $\mathbf{x}$, which can become unfeasible if the number of possible assignations is high. CRFs overcome this issue by compactly encoding $P(\mathbf{y}|\mathbf{x})$ through a graph structure that captures the dependence relations among random variables. Concretely, a CRF factorizes $P(\mathbf{y}|\mathbf{x})$ over an undirected graph $H = (V, E)$, where V is a set of nodes, one per each random variable in $\mathbf{y}$, and E is the set of edges linking nodes that are contextually related. These relations are established according to the semantics of the domain and the geometry of the scene.

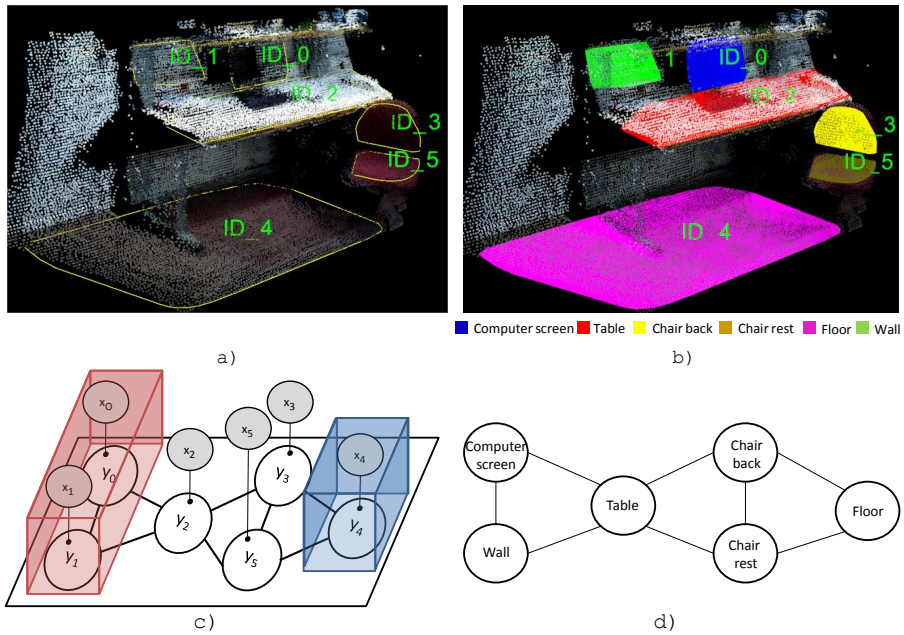


Figure A.1: a) Example of a scene segmented into planar patches (labeled with an ID and delimited by yellow lines). b) Scene objects recognized by our method. c) Graphical model built for the planar patches shown in a). Each patch is associated to a node $y_0, .., y_5$, whose probabilistic distributions are conditioned to their respective patch observations $x_0, .., x_5$ (observation x_i corresponds to patch ID_i). Near patches are linked by an edge. The blue box encapsules the scope of a particular unary factor, while the red one shows the scope of a pairwise factor. d) The resultant graphical model after the execution of the recognition method, when random variables take a value according to their most probable assignation.

For example, in the CRF structure of figure A.1-c defined from the observations in figure A.1-a, the nodes y_3 and y_5 are linked due to the proximity in the scene of their related observed planar patches ID_3 and ID_5. The intuition behind this is that only the neighbors of an object will directly influence its recognition, as stated by the Markov properties [16].

According to the Hammersley-Clifford theorem [16], the factorization of $P(\mathbf{y}|\mathbf{x})$ over a CRF can be expressed as a product of factors. A factor is a function associated to a random variable or a set of variables that represents a probability distribution over it/them. In this work we consider two types of factors: *unary* and *pairwise* (see figure A.1-c). Unary factors encode knowledge about the properties of the object itself and therefore affect to single nodes. On the other hand, pairwise factors act over connected variables, and encapsulate knowledge about the objects' relations. In other words, unary factors model how likely an object y_i belongs to a certain class in L

Table A.1: Unary and pairwise features used in this work to characterize planar patches of the scene.

id	**Unary features**
l_1	Centroid height from the floor.
l_2	Orientation w.r.t. the horizontal.
l_3	Area of its bounding box.
l_4	Elongation.
id	**Pairwise features**
i_1	Perpendicularity.
i_2	on/under relation.
i_3	Vertical distance of centroids.
i_4	Ratio between areas.
i_5	Ratio between elongations.

based only on the observed properties x_i, whereas pairwise factors state the compatibility of an object assignation with respect to the classes of its neighboring objects.

More concretely, we define an unary factor, denoted by $U(\cdot)$, as a linear model:

$$U(y_i, x_i, \omega) = \sum_{l \in L} \delta(y_i = l)\omega_l f(x_i) \tag{A.1}$$

where $f(x_i)$ computes a vector of features that characterizes the object x_i, ω_l is a vector of weights for the class l obtained during the training phase, and $\delta(y_i = l)$ is the Kronecker delta function, which takes value 1 when $y_i = l$ and 0 otherwise. Table A.1-top shows the unary features used in this work. As an example, let's consider the planar patch *ID_0* representing a computer screen in figure A.1, which corresponds to observation x_0. In this case, the outcome of the $f(\cdot)$ function is $f(x_0) = [1.06, 0, 0.17, 1.83]$, where 1.06 stands for its centroid height, 0 its orientation, and so on.

On the other hand, we define the pairwise factor $I(\cdot)$ as:

$$I(y_i, y_j, x_i, x_j, \theta) = \sum_{l1 \in L} \sum_{l2 \in L} \delta(y_i = l_1, y_j = l_2)\theta_{l_1 l_2} g(x_i, x_j) \tag{A.2}$$

where the function $g(x_i, x_j)$ computes pairwise features between the observations x_i and x_j, and $\theta_{l_1 l_2}$ is a vector of weights for the pair of classes l_1 and l_2. Table A.1-bottom enumerates the pairwise features used to characterize the objects' relations.

For convenience, the product of factors over the posterior probability P can be expressed by means of log-linear models as:

$$P(\mathbf{y}|\mathbf{x}, \omega, \theta) = \frac{1}{Z(\mathbf{x}, \omega, \theta)} e^{-\varepsilon(\mathbf{y}, \mathbf{x}, \omega, \theta)} \tag{A.3}$$

where $Z(\cdot)$ is the normalizing partition function so $\sum_{\xi(\mathbf{y})} p(\mathbf{y}|\mathbf{x}, \omega, \theta) = 1$, being $\xi(\mathbf{y})$ an assignation to the variables in $\mathbf{y}$, and $\varepsilon(\cdot)$ the so-called energy function defined as:

$$\varepsilon(\mathbf{y},\mathbf{x},\omega,\theta) = \sum_{i\in V} U(y_i,x_i,\omega) + \sum_{(i,j)\in E} I(y_i,y_j,x_i,x_j,\theta) \tag{A.4}$$

3.1 Training the model

Training a CRF consists of estimating the vectors of weights ω and θ that maximize the likelihood function:

$$\max_{\omega,\theta} L_P(\omega,\theta|D) = \max_{\omega,\theta} \prod_{d\in D} P(\mathbf{y}_d|\mathbf{x}_d,\omega,\theta) \tag{A.5}$$

where $D = \{d_1,d_2,...,d_m\}$ is a dataset composed of m training samples. Each training sample contains the observations to be recognized $\mathbf{x}_d$ labeled with their ground truth object classes in $\mathbf{y}_d$. Solving equation A.5 requires the calculation of the partition function Z, which becomes computationally intractable in practice. To overcome this problem, it is common to resort to the pseudo-likellihood, instead [16]. It consist of an alternative, tractable objective function for which the estimation of ω and θ converges to those computed by the likelihood one if a sufficient large number of samples is provided.

As commented, the training dataset must be comprehensively enough to accurately capture the characteristics and variability of the domain. At this point, the exploitation of semantic knowledge brings two interesting advantages: (i) it provides synthetic training samples that naturally encode the variability of the domain (as it is shown in section 4.2), and (ii) it eliminates the task of gathering, processing and labelling sensorial data to generate a sufficiently comprehensive dataset.

3.2 Inference

Given the observation of a scene, the graph $H = (V,E)$ is built according to the sensed elements $\mathbf{x}$ and the conditional dependencies between the random variables $\mathbf{y}$, as described above. Thereby, the recognition problem consists of finding the assignation to the variables in $\mathbf{y}$ that maximizes the posterior, that is:

$$\begin{aligned} \hat{y} &= arg\max_{y} P(\mathbf{y}|\mathbf{x},\omega,\theta) \\ &= arg\max_{y} \frac{1}{Z(\mathbf{x},\omega,\theta)} e^{-\varepsilon(\mathbf{y},\mathbf{x},\omega,\theta)} \end{aligned} \tag{A.6}$$

Since the partition function does not depend on the assignations to $\mathbf{y}$, we can simplify this expression to:

$$\hat{y} = arg\max_{y} e^{-\varepsilon(\mathbf{y},\mathbf{x},\omega,\theta)} \tag{A.7}$$

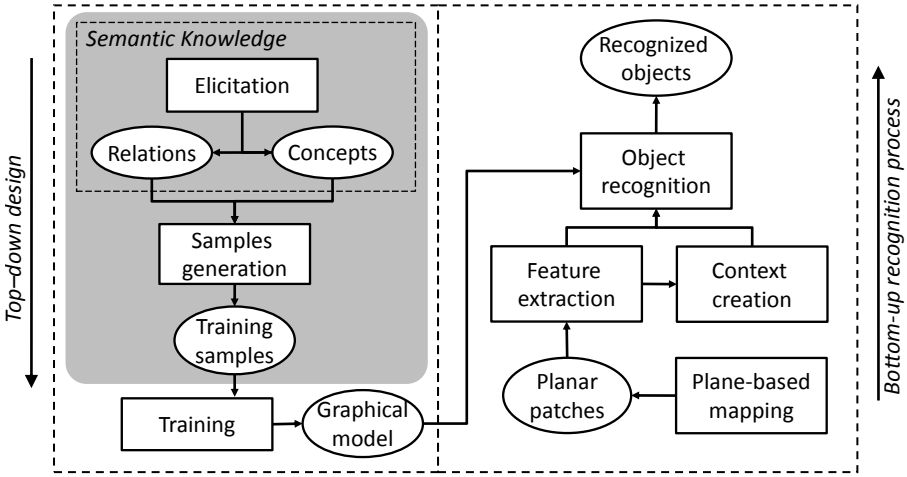


Figure A.2: Overview of the developed framework for object recognition. The shadowed area delimitates the proposed components for the generation of training samples. Boxes represent processes, whereas ovals are generated/consumed data.

This equation is known as the Maximun a Posteriori (MAP) query or Most Probable Explanation (MPE). Although we avoid the computation of the partition function, the exact computation of this query is still unfeasible, as the number of possible configurations is exponential with the number of nodes in V. To overcome this issue, we use the Iterated Conditional Modes (ICM) algorithm [3].

As an illustrative example, figure A.1-d displays the values taken by the nodes of the graph in figure A.1-c after the inference process, and figure A.1-b shows these results in the scene.

4 Using Semantic Knowledge for training

The proposed method for training PGMs according to semantic knowledge follows a *top-down* methodology (see figure A.2). The design starts with the definition of an ontology for the knowledge domain at hand, e.g. an office environment, through human elicitation, stating the typical objects, their geometrical features, and relations. Then, the encoded semantic knowledge is used for generating sets of synthetic samples, which replace the real datasets required for training.

Once the PGM is trained, and aiming to show its performance, it is integrated into an object recognition framework that works following a *bottom-up* stance (see figure A.2). During the robot operation, a plane-based mapping algorithm [6] extracts planar patches, which are characterized through a number of features, e.g., size, orientation, position or contextual relations. These characterized planar patches feed the inference process described in section 3.2.

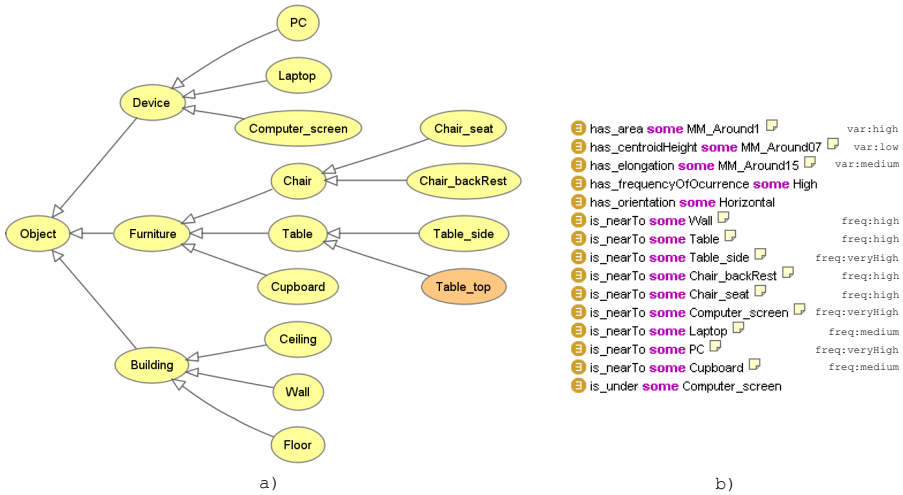


Figure A.3: a) Hierarchy of concepts defined in the office ontology used in this work. b) Definition of the `Table_top` concept based on properties, relations and annotations.

The next section details the process for encoding the semantic knowledge provided by human elicitation into an ontology, and then section 4.2 describes its utilization for generating an arbitrary number of synthetic training samples.

4.1 Ontology definition through Human Elicitation

An ontology is a representation of a conceptualization related to a knowledge domain that consists of a number of *concepts* arranged hierarchically, *relations* among them, and *instances* of concepts, also called *individuals* [30]. For example, an office environment can be represented by an ontology of concepts defining rooms and objects, e.g. `meeting_room`, `office_table` or `printer`, and instantiations of such concepts, e.g. `meeting _room-1`, which refers to a particular meeting room. Ontologies also comprise relations among concepts like "`Object has_location Room`", which establishes that the instances of the concept `Object` are (can be) located at a particular instance of `Room`. For instance, a possible relation can be "`office_table-2 has_location meeting_room-1`". The ontologies used in this work are defined by human elicitation, a process that enables the exploitation of its experience and knowledge[1] for setting the features and relations among the domain concepts.

Figure A.3-a) depicts part of the office ontology defined in our experiments. The root concept is `Object`, with three subconcepts: `Device`, `Furniture` and `Building`, which represents the objects that are typically found in office environments. Notice that the person can vary the granularity of the defined concepts, as it is the case of

[1]Please notice that the source of this information could be also a large number of humans, i.e. crowdsourcing.

Table A.2: Properties defined into the ontology.

Name	Meaning
has_area	Area of the object in m^2.
has_centroidHeight	Height of the object centroid w.r.t. the floor in m.
has_elongation	Ratio between the object length in its two main directions.
has_frequencyOfOccurrence	How often an object appears in the studied environment.
has_orientation	Main orientation of the object.
is_nearTo	An object is near to other one.
is_on	An object is placed on another one.
is_under	An object is placed under another one.

the concept `Table` that has been split into two related concepts: `Table_top` and `Table_side`.

The geometrical properties considered by the human to describe these concepts and their relations are enumerated in table A.2. Such properties can be interpreted as restrictions to be fulfilled by instances belonging to that concepts. Additionally, they compound the minimum set of properties that permits a human to distinguish between the object classes employed during the method evaluation (see section 5). For example, figure A.3-b) shows the definition of the concept `Table_top`, restricting the geometric features and relations considered for a standard table top.

The geometric features defined over the concepts are useful to describe the typical shape, size or relative position of their instances. However, not all the instances of a particular concept have exactly the same appearance in the real world. To quantify objects' variability, the person may also annotate the encoded restrictions with a discrete value from the set $R_A = \{null,\ veryLow,\ low,\ medium, high, veryHigh\}$. Thus, according to the `Table_top` definition given in A.3-b), its height shows a *low* variability around the established value of $0.7m$, indicating that most tables share this typical height. The area, however, can largely vary from the averaged value, i.e. $1m^2$, expressing the differences in size of the tables that can be found in an office. Given that the same set of geometric features is employed for describing all the concepts during the elicitation process, the time needed for their definition scales linearly with the number of object classes. It is also worth to mention that, although the definition of the objects' variability by means of elements of the set R_A could seem subjective (i.e. dependant on the person): the objectiveness can be increased through crowd-sourcing; the crispy values from R_A are relevant but not determinant during the generation of synthetic data – see section 4.2.

Proximity restrictions between objects are also incorporated into the ontology with a value from the R_A set, but with a different meaning. In this case, it is indicated how frequently a particular relation holds. For instance, the person establishes that a computer screen and a table top likely appear close to each other by adding an annotation with the value *veryHigh* (see figure A.3-right). Note that it is not needed to set the proximity relations among all the considered object classes, which would lead to

Concept	has_frequency OfOccurrence	P(appearing)	Sample
Floor	high	0.8	appearing
Wall	high	0.8	appearing
Table_top	veryHigh	0.9	appearing
Table_side	low	0.25	not_appe.
Chair_back	high	0.8	not_appe.
Chair_seat	medium	0.6	appearing
Computer _screen	high	0.8	appearing

is_nearTo	Frequency	P(near)	Sample
Floor	null	0	not_near
Wall	high	0.75	near
Chair_seat	high	0.75	near
Computer_s creen	veryHigh	0.9	near

Figure A.4: Left, example of discrete probability distributions built according to the `has_frequencyOfOcurrece` relation of each concept. These distributions determine which objects are included into the synthetic scenario. Right, context creation for an object of the class `table_top` according to the objects included in the synthetic scenario.

a quadratic increment in the time spent in their definition, but just between the objects that are more commonly encountered together. Thus, extending the previous example, the person could avoid the definition of the relation between computer screens and trash bins, since they seldom appear close in an office.

4.2 Generation of training samples

Upon the semantic knowledge encoded in the ontology, the system generates samples in the form of synthetic scenes following four steps (notice that the stage presented here does not involve the human participation):

1. **Inclusion of objects in the scene**. The set of objects that appear in the synthetic scene is selected according to the relations `has_frecuency- OfOcurrence` defined in the ontology. For that, we use a discrete probability distribution that establishes the likelihood of the presence of each object. For example, following the `Table_top` definition where `has_frecuencyOfOcurrence=high`, such a probability distribution can be defined by the person as $P(Table\_top_{appearing}) = 0.8$ and $P(Table\_top_{notAppearing}) = 0.2$. Samples from these distributions are drawn, yielding the set of objects included in the scene as illustrated in figure A.4-left. In this example the objects included are: parts of the floor and a wall, a table top, a chair seat and a computer screen.

2. **Object characterization**. The geometrical features of the objects included in the synthetic scene in the previous step are reified according to their concepts' definitions in the ontology. To this end, a Gaussian distribution, $N(\mu, \sigma)$, is considered for each defined concept and for each defined geometric property, i.e. `has_area`, `has_centroidHeight`, `has_elongation` and `has_orientation`,

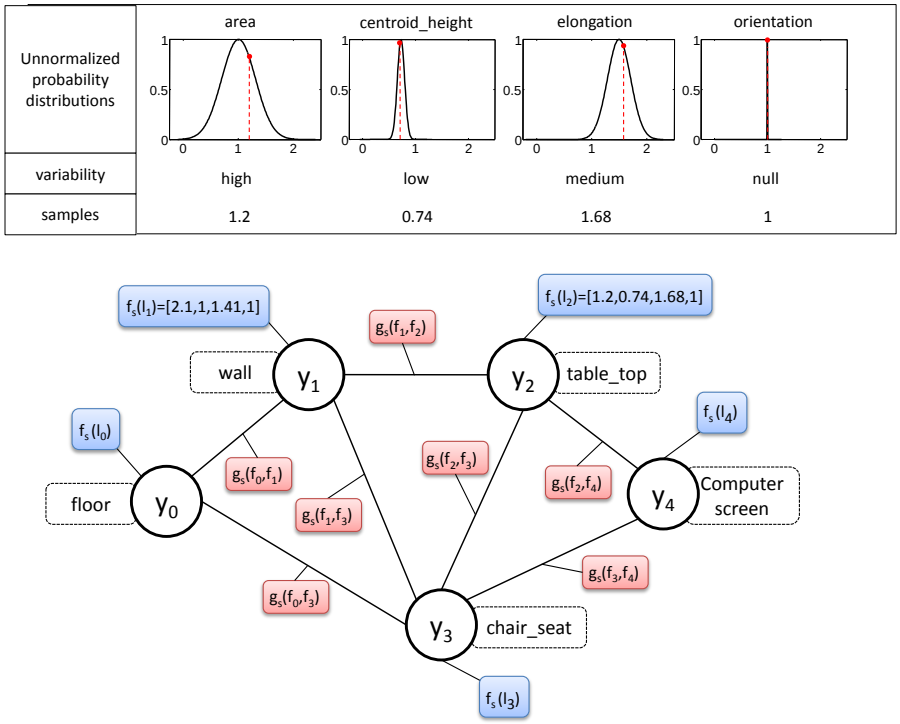


Figure A.5: Top, samples drawn (red lines) from the probability distributions for an object of the class `Table_top`, built according to its geometrical restrictions and the annotated variability in the ontology (see figure A.3-b). Bottom, graphical model that results from the objects included in figure A.4-left and their generated relations.

where the mean μ is the value of that concept for that property in the ontology, and the standard deviation σ is a quantification of the respective annotated variability. For instance, for the `has_area` property of the `Table_top` concept, the person implicitly encoded a Gaussian distribution with $\mu = 1$ and a *high* standard deviation, e.g. $\sigma = 0.75$. Then, samples drawn from these distributions are used as features of the included objects (see figure A.5-top). These synthetic features are computed by the $f_s(l_i)$ function, where l_i is the class of the included object i. This function replaces $f(x_i)$ during the training phase (recall equation A.1 in section 3).

3. **Context creation**. The contextual relations between the included objects are established according to the `is_nearTo` properties and their `frequency` annotations. For example, if the scene contains a `Table_top` and a `Chair_backRest`, they will be placed near one to another into the synthetic scene with a `high` probability, as stated by the ontology. Figure A.4-right shows an example of

the definition of the contextual relations for a `Table_top` object according to the objects previously included in the scene (see figure A.4-left), its `is_nearTo` relations and their `frequency` annotations.

4. **Context characterization**. Different features for the relations established in the previous step are computed, adding valuable contextual information. Examples of these features are: difference between centroid heights, perpendicularity, difference between areas, areas ratio, difference between elongations, etc. To compute them, the information produced in the *objects characterization* step is used. For example, if a `Table_top` with a height of 0.7 m. and a `Chair_backRest` showing a height of 0.32 m. are placed near in a synthetic scenario, their context can be characterized with the difference between the heights of their centroids: 0.38 m.

   Two additional binary features are considered to establish that an object is placed *on* or *under* other, according to the `is_on` and `is_under` relations of the ontology. Notice that these features characterize the context of a pair of objects that have been previously related in the synthetic scenario according to their proximity.

   The set of contextual features for objects (i, j) are yielded by the function $g_s(f_i, f_j)$, where $f_i = [f_s(l_i), l_i]$, being $f_s(l_i)$ the features computed in the *object characterization* step for object *i*, and l_i the class of that object. This function replaces the $g(x_i, x_j)$ one in equation A.2 (section 3).

Figure A.5-bottom shows the components of a synthetic scene produced by the steps described above in the form of a graphical model, compound of nodes representing the included objects, and edges stating their relations. Notice that the characterization of a `Table_top` illustrated in figure A.5-top is in fact carried out by $f_s(l_2)$. As an example of context characterization, let's consider the context established by the objects `wall` (node y_1) and `table_top` (node y_2). Supposing that the contextual features employed are, for instance, difference between centroid heights, perpendicularity, *is on* and *is under*, then such a characterization is generated as $g_s(f_1, f_2) = [0.9, 1, 0, 0]$, which sets that: their centroids are separated by a vertical distance of 0.9 m.; given that the `wall` is vertical and the `table_top` is horizontal they are perpendicular; any object is located on or under the other one.

5 Evaluation

In order to evaluate our approach, we have trained a number of CRFs with synthetic data and assessed their suitability to recognize objects from: i) office scenarios within the UMA-offices dataset (section 5.1), and ii) office and home scenes within the NYU2 dataset [28](section 5.2).

Figure A.6: The mobile robot Rhodon gathering 3D data within an office room.

5.1 Results with the UMA-offices dataset

The UMA-offices dataset was acquired with the mobile robot Rhodon, equipped with a Kinect device mounted on a pan-tilt unit (see figure A.6), and entails 25 office environments from the University of Málaga. In the experiments, seven object classes were considered: $L = \{$*floor, wall, table, table_side, chair_back_rest, chair_seat* and *computer_screen*$\}$, and the ground-truth was provided by an human operator. It is worth to mention that the person that carried out the human elicitation process in the experiments (section 4.1) has worked in different office environments, but he did not visit the offices from the gathered dataset.

In our implementation, we rely on the UGM library [27] for training the CRF using the optimization of the pseudo-likelihood function (see section 3.1). Concretely, a Quasi-Newton method with Limited-Memory BFGS [22] is used, which is able to optimize complex objective functions with a high number of parameters.

The performance of CRFs trained with the proposed method is assessed through the micro/macro precision/recall metrics [1] computed for the results yielded by the recognition process. Briefly, the *precision* of a given class of objects c_i is defined as the percentage of objects recognized as belonging to c_i that really belong to that class. Let $recognized(c_i)$ be the set of objects recognized as belonging to the class c_i, $gt(c_i)$ the set of objects of that class in the ground-truth, and $|\cdot|$ is the cardinality of a set, then the *precision* of the classifier for the class c_i is defined as:

$$precision(c_i) = \frac{|recognized(c_i) \bigcap gt(c_i)|}{|recognized(c_i)|} \tag{A.8}$$

On the other hand, the *recall* of a class c_i expresses the percentage of the objects that belonging to c_i are recognized as members of that class:

$$recall(c_i) = \frac{|recognized(c_i) \bigcap gt(c_i)|}{|gt(c_i)|}. \tag{A.9}$$

Precision and recall are metrics associated to a single class. It is also of interest to know the performance of the proposed method for all the considered classes. This can be measured by adding the so-called macro/micro concepts. *Macro precision/recall* represents the average value of the precision/recall for a number of classes, and it is defined in the following way:

$$macro\_precision = \frac{\sum_{i \in L} precision(c_i)}{|L|} \tag{A.10}$$

$$macro\_recall = \frac{\sum_{i \in L} recall(c_i)}{|L|} \tag{A.11}$$

Finally, *micro precision/recall* represents the percentage of objects in the dataset that are correctly recognized with independence of their belonging class, that is:

$$micro\_precision(c_i) = \frac{\sum_{i \in L} |recognized(c_i) \bigcap gt(c_i)|}{\sum_{i \in L} |recognized(c_i)|} \tag{A.12}$$

$$micro\_recall(c_i) = \frac{\sum_{i \in L} |recognized(c_i) \bigcap gt(c_i)|}{\sum_{i \in L} |gt(c_i)|} \tag{A.13}$$

Since we assume that objects belong to a unique class, then $\sum_{i \in L} |gt(c_i)| = \sum_{i \in L} |recognized(c_i)|$, and consequently the computation of both micro precision/ recall metrics gives the same value.

In our experiments we have trained five CRFs using the same synthetic dataset that comprises 1000 training samples including a total of 7170 objects and 16700 relations among them. CRFs differ in the combination of the selected pairwise features (configurations), aiming to analyze their suitability to the given environment.

The trained CRFs with synthetic data have been used to recognize the objects from the UMA-offices dataset. The results of the recognition process using the above metrics are shown in table A.3. Observe that the achieved micro precision/recall is above 81%, with a best value of 90.91% for the configuration #2. Figure A.8 shows some scene objects recognized with this configuration, while figure A.7-left illustrates its confusion matrix. Note that in this case, the most challenging class to recognize is `table_side`, since it may not be clearly differentiated from other object classes like `chair_back`. Next, we highlight some meaningful comparisons and results of our approach.

Table A.3: Results of the recognition process with different sets of pairwise features (configurations) and methods for the UMA-offices dataset. For the convenience of the reader, these features, previously listed in table A.1, are: i_1–Perpendicularity, i_2–on/under relation, i_3–Vertical distance of centroids, i_4–Ratio between areas, and i_5–Ratio between elongations. The features employed in each configuration are: #1=$\{None\}$, #2=$\{i_1, i_2, i_3\}$, #3=$\{i_1, i_2, i_3, i_4\}$, and #4=$\{i_1, i_2, i_3, i_4, i_5\}$.

Method	Metric	Configurations			
		#1	#2	#3	#4
CRF trained with synthetic data	micro p./r.	81.82	**90.91**	86.06	84.85
	macro p.	80.17	**89.25**	84.91	81.82
	macro r.	83.78	**89.99**	86.69	83.95
CRF trained with real data [16]	micro p./r.	83.19	**87.50**	86.65	84.47
	macro p.	81.93	**85.84**	85.19	81.90
	macro r.	82.76	**86.36**	85.72	82.46

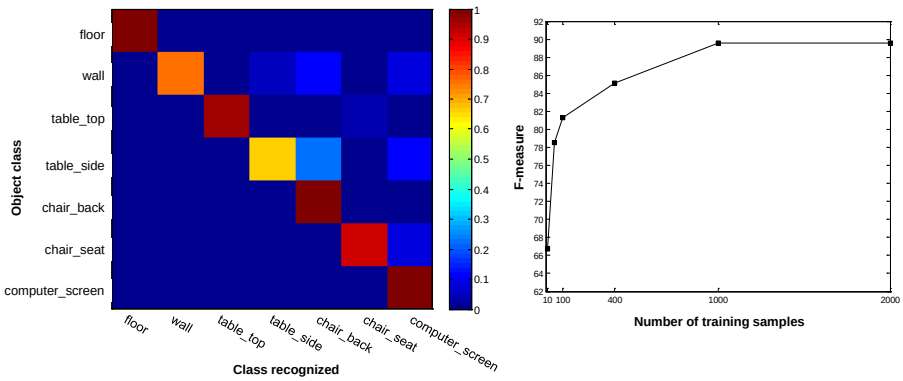


Figure A.7: Left, confusion matrix that relates the ground truth to the recognition results in the second configuration. Right, influence of the number of training samples on the recognition success as it is measured by the F-measure.

Comparison with state-of-the-art methods. We have compared the results of our method with two state-of-the-art alternatives: i) a standard formulation of a CRF trained and tested with real data [16], and ii) the CRF presented in [34]. The results for both recognition systems were obtained through a 5-fold cross-validation and average process using the UMA-offices dataset. Such a process firstly splits the 25 offices into 5 groups. Then, four of these groups are used for training, and the remaining one for testing. This process is repeated five times shifting the group used for testing, and finally the results are averaged. Table A.3 shows the results for the evaluation with the CRFs in [16], while the CRF with the configuration presented in [34] achieved a micro p./r. of 82.46%. These figures reveal that CRFs trained with the proposed

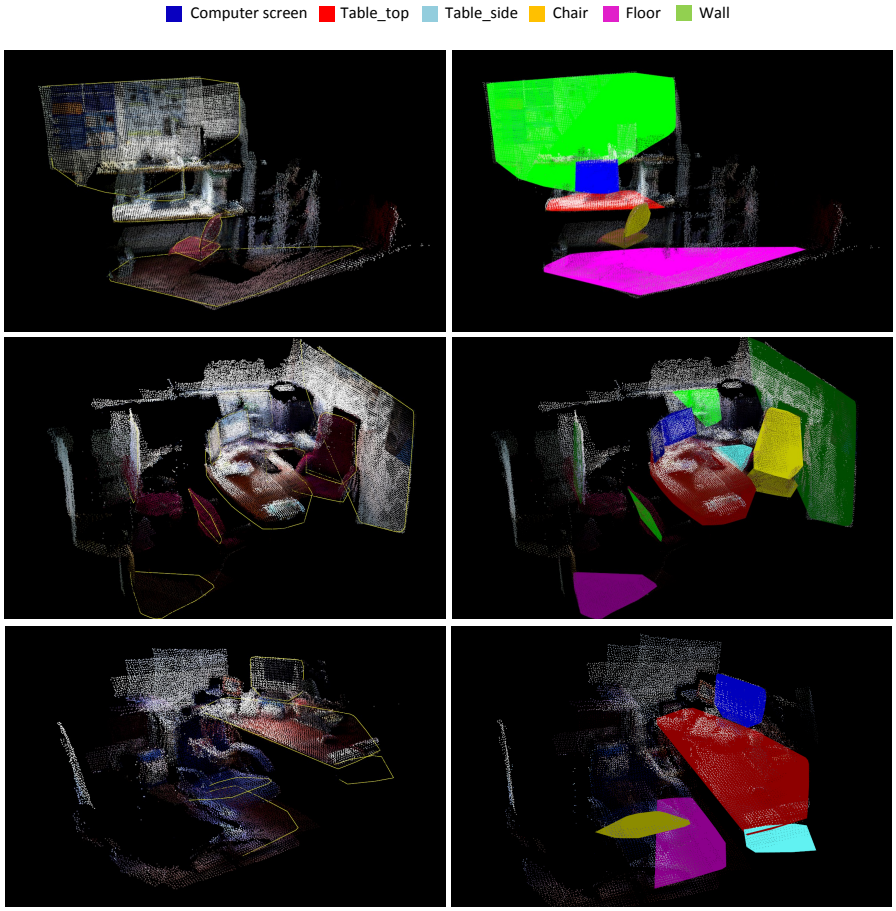


Figure A.8: Examples of scene object recognitions performed by our method. Left column, observed scenes with the detected planar patches delimited by yellow lines. Right column, recognition results of such scenes.

method can compete with, an even outperform the results of the other two state-of-the-art alternatives.

How much does the context relations contribute to the recognition performance? We have trained a CRF that does not consider pairwise factors, i.e., only taking into account the geometric properties of the planar patches (unary factors). The recognition results of using this CRF correspond to the first configuration in table A.3, which shows a significantly lower success than the other configurations exploiting contextual relations.

What pairwise features are more discriminative? Notice that, in the results shown in table A.3, the best ones are obtained when using perpendicularity, on/under

and centroid height difference relations (configuration #2), whereas the inclusion of the area and elongation ratios (configurations #3 and #4) deteriorates the method performance. This indicates that both features have a low discriminant capability, influencing negatively to the recognition process. It is important to underscore that this conclusion only holds for systems employing the set of object classes L, so these contextual features could be useful in other applications or domains relying on a different set of object classes.

How much does the size of the training dataset affect the recognition performance? Given that our method can generate an arbitrary number of samples, we have trained several CRFs with datasets of different sizes. To facilitate the comparison of their outcomes, the previous macro precision/recall metrics has been combined through the computation of their harmonic mean, also known as the $F-measure$. The harmonic mean, that mitigates the impact of large measures and increments the influence of small values, is defined as follows:

$$F = 2 * \frac{macro\_precision * macro\_recall}{macro\_precision + macro\_recall} \tag{A.14}$$

Figure A.7 shows the results of such outcomes, where the F value increases from the 66.68 obtained with 10 training samples up to 89.61 with 1000 samples. Notice that in this experiment the improvement reaches an upper limit for 1000 samples. This result remarks the importance of using large datasets to properly capture the variability of the domain as well as the convenience of techniques to reduce the burden of data gathering.

Do the generated synthetic data capture actual object properties and relations? In order to test the validity of the synthetic data generated for training CRFs, that is, how well the elicited ontology and the proposed method capture the real world, we have employed a CRF trained with our approach for recognizing objects from both real and synthetic datasets. Concretely, we have considered the CRF with configuration #2, the 25 offices from the UMA-offices dataset, and 25 synthetic scenarios generated with the approach described in section 4.2. The performance testing with the synthetic dataset yielded a micro precision/recall of 91.85%, a macro precision of 90.30%, and a macro recall of 90.39%. Note that these figures are similar to those obtained for the real dataset (see table A.3, configuration #2), which reveals the suitability of both the ontology defined by the person and our approach for the generation of synthetic scenarios through the exploitation of semantic knowledge.

Computational performance. The training process, including the generation of synthetic samples, takes from 0.21 seconds when using 10 samples, up to 39.62 seconds for 1500 in a PC with an Intel®Core™*i*5 3330 microprocessor at 3GHz and 8 GB DDR3 RAM memory at 1.6 GHz. Notice that the training process is performed only once, and does not take place during the robot operation. On the other hand, the inference process takes, on average, less than 0.2 milliseconds, which enables its integration in object recognition frameworks aiming to operate on-line.

Time saving using human elicitation plus synthetic samples generation. The results obtained in our experiments justify our claim that the proposed method can

Table A.4: Results of the recognition process with different sets of pairwise features (configurations) and methods for the NYU2 dataset. No pairwise features are used within configuration #1. #2 resort to i_1–Perpendicularity, i_2–on/under, relation, and i_3–Vertical distance of centroids.

Method	Metric	Configurations #1	#2
CRF trained with synthetic data	micro p./r.	76.23	**81.37**
	macro p.	73.72	**79.21**
	macro r.	76.32	**80.35**
CRF trained with real data [16]	micro p./r.	74.21	**76.03**
	macro p.	65.57	**67.65**
	macro r.	66.70	**69.57**

successfully replace the time-consuming and arduous tasks of gathering and processing real datasets. In order to also support its advantage for saving time/cost in the process, we have measured the time consumed by the human elicitation and samples generation processes.

In our experiments, the human elicitation process for the office domain took 20 minutes, including the collection of the knowledge from the person and its codification into an ontology.

On the other hand, the time employed in the synthetic samples generation is negligible, since our method is capable of generating hundreds of samples in a less than a second (e.g., 1500 samples in $0.11sec.$). Thus, summing up the time spent for human elicitation, synthetic samples generation, and CRF training, our object recognition system can be ready to work in less than 21 minutes. Thereby, the presented methodology reduces dramatically the time required for training with real data, which involves the navigation of the robot through a number of locations (large enough to capture the variability of the domain), collecting the data, and its posterior processing. In our case, the gathering and processing of the 25 offices within the UMA-offices dataset took more than 7 hours, that is, 20 times higher than the time needed by our method.

5.2 Results with the NYU2 dataset

Our approach has been also evaluated considering 61 scenes from *office-environments*, and 200 *home-environment* scenes, all of them from the NYU2 dataset [28].

Office-environments. For the tests within the office domain, two of the five CRFs trained during the evaluation with the UMA-offices have been reused, concretely the ones with configurations #1 and #2. Notice that the same set of objects classes L has been considered.

Table A.4 depicts the results of these tests. We can see how the integration of contextual information increments the micro p./r. value in a $\sim 5\%$. This is lower than the $\sim 9\%$ achieved with UMA-offices, which can be explained by the limited

contextual information obtained from one-shot observations in NYU2 w.r.t. the multi-shot registered scenarios gathered in the UMA-offices dataset.

The performance of our approach has been also contrasted with: i) the results yielded by a standard CRF [16] trained and tested with office data from NYU2, and ii) the CRF configuration from [34], following again a 5-fold cross-validation and average methodology. The second row of table A.4 shows the outcome of CRFs from [16], while the configuration in [34] reached a micro p./r. of 73.10% relying on unary features, and of 75.42% also integrating the pairwise ones. Both systems improve their results a $\sim 2\%$ when contextual information is introduced, however, they are still under the performance reached by the proposed methodology.

Home-environments. The aim of the testing with home scenes is to validate the applicability of the proposed approach to a different domain. For that, human elicitation has been used to define a new *home ontology*, publicly available at `http://goo.gl/mz51ho`, which contains 20 object classes typically found in a home environment, e.g. bottle, cabinet, faucet, sink, toilet, sofa, pillow, bed, clothes, etc. These objects exhibit arbitrary shapes, so the recognition framework shown in figure A.2 has been modified to work with object bounding boxes as geometric primitives, instead of the planar patches used in offices. In this case, the following properties replace those in table A.2 for defining objects' concepts: `hasBiggestArea`, `hasColorVariation`, `hasElongation`, `hasHeight`, `hasOrientation`, `hasSize` and `isPlanar`. The contextual relations were codified in the same way as with the office ontology (recall section 4.1).

The resultant ontology was exploited to generate synthetic training data, and two CRF were tuned. The first CRF considers the following unary features to characterize an object: orientation, planarity, and size of its bounding box, area of its two principal directions, height from the floor, and color hue variation, and the second CRF also includes contextual relations characterized by: difference between principal directions, vertical distance of centroids, volume ratio, connectivity and object-object compatibility. These configurations yielded a micro p./r. of 64% and a 69.44% respectively.

Additionally, a CRF following the standard formulation [16] has been trained and tested through the above described 5-fold cross-validation and average process using the 200 home-environment scenes. In this case, the system achieved a 61.67% of micro p./r. relying only on unary features, and a 65.42% also considering contextual relations. A comparison with the CRF from [34], as conducted in the previous sections, does not make sense here since it relies on planar patches. These figures support our claim that the proposed training approach can be applied to different environments compound of objects showing arbitrary shapes.

6 Conclusions and future work

Collecting real data for training object recognition systems is a highly time-consuming and cumbersome task, since the gathered data must be representative enough of the given domain. The approach presented in this paper overcomes this issue by replacing the data gathering task with the generation of synthetic samples. These samples

implicitly capture the semantics of the scene by exploiting the knowledge codified in an ontology by a human. Our proposal has also the advantage of avoiding the processing of the collected sensorial information, which usually involves: segmentation, feature extraction, creation of contextual relations (if the recognition method leverages them), and finally regions' labeling by a human. In order to support our claim, we have trained and evaluated a number of Conditional Random Fields, with different sets of pairwise features and two datasets.

The results obtained in the conducted evaluations achieve a recognition success of $\sim 90\%$ within the UMA-offices dataset, and of $\sim 81\%$ and $\sim 69.5\%$ using office and home scenes from the NYU2 dataset respectively, revealing that the use of semantic knowledge can be exploited for the suitable training of recognition systems. Our approach has been also compared with other state-of-the-art approaches based on CRFs yielding a substantial improvement. A number of additional, related issues have been also addressed. Firstly, the discriminant capability of different sets of contextual features has been studied, showing their positive effect on the system performance. Also, the relation between the size of the training datasets and the system performance has been analyzed, obtaining the expected conclusions: the larger the dataset is, the better the system outcomes are. It has been also reckoned the computational efficiency, evidencing the suitability of the proposed system for real time robot applications. Finally, we have studied the time saving gained with the use of human elicitation plus synthetic samples generation processes, resulting 20 times lower than the time spent in collecting real data from the UMA-offices dataset.

In the future we plan to exploit the symbolic representation of the recognized objects to perform higher-level robot tasks, such as efficient task planning or knowledge inference. We also plan to include temporal relations in the ontology as well as enabling crowdsourcing for the human elicitation process.

Acknowledgements

We are very grateful to our colleague E. Fernandez-Moral for providing us the implementation of the plane-based mapping algorithm, as well as for his support during the collection of the office dataset used to evaluate our method. This work has been funded by the Spanish grant program FPU-MICINN 2010 and the Spanish project "TAROTH: New developments toward a robot at home".

References

[1] Anand, A., Koppula, H. S., Joachims, T., Saxena, A., Jan. 2013. Contextually guided semantic labeling and search for three-dimensional point clouds. In the International Journal of Robotics Research 32 (1), 19–34.

[2] Baroni, M., Lenci, A., 2010. Distributional memory: A general framework for corpus-based semantics. Computational Linguistics 36 (4), 673–721.

[3] Besag, J., 1986. On the statistical analysis of dirty pictures. Journal of the Royal Statistical Society. Series B (Methodological) 48 (3), pp. 259–302.

[4] Coradeschi, S., Saffiotti, A., 2003. An introduction to the anchoring problem. Robotics and Autonomous Systems 43 (2-3), 85–96.

[5] Fergus, R., Fei-Fei, L., Perona, P., Zisserman, A., 2005. Learning object categories from google's image search. In: IEEE International Conference on Computer Vision (ICCV 2005). Vol. 2. pp. 1816–1823 Vol. 2.

[6] Fernandez-Moral, E., Mayol-Cuevas, W., Arevalo, V., Gonzalez-Jimenez, J., 2013. Fast place recognition with plane-based maps. In: IEEE International Conference on Robotics and Automation (ICRA 2013). pp. 2719–2724.

[7] Floros, G., Leibe, B., 2012. Joint 2d-3d temporally consistent semantic segmentation of street scenes. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2012). pp. 2823–2830.

[8] Galindo, C., Fernandez-Madrigal, J., Gonzalez, J., Saffiotti, A., 2007. Using semantic information for improving efficiency of robot task planning. In: IEEE International Conference on Robotics and Automation (ICRA), Workshop on Semantic Information in Robotics. Rome, Italy.

[9] Galindo, C., Fernandez-Madrigal, J., Gonzalez, J., Saffiotti, A., 2008. Robot task planning using semantic maps. Robotics and Autonomous Systems 56 (11), 955–966.

[10] Galindo, C., Saffiotti, A., 2013. Inferring robot goals from violations of semantic knowledge. Robotics and Autonomous Systems 61 (10), 1131–1143.

[11] Galleguillos, C., Belongie, S., Jun. 2010. Context based object categorization: A critical survey. Computer Vision and Image Understanding 114 (6), 712–722.

[12] Günther, M., Wiemann, T., Albrecht, S., Hertzberg, J., 2013. Building semantic object maps from sparse and noisy 3d data. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2013). pp. 2228–2233.

[13] Horrocks, I., Patel-Schneider, P. F., Boley, H., Tabet, S., Grosof, B., Dean, M., 2004. SWRL: A semantic web rule language combining OWL and RuleML. W3C Member Submission, World Wide Web Consortium.

[14] Jancsary, J., Nowozin, S., Sharp, T., Rother, C., 2012. Regression tree fields - an efficient, non-parametric approach to image labeling problems. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2012). pp. 2376–2383.

[15] Kahler, O., Reid, I., Dec 2013. Efficient 3d scene labeling using fields of trees. In: IEEE International Conference on Computer Vision (ICCV 2013). pp. 3064–3071.

[16] Koller, D., Friedman, N., 2009. Probabilistic Graphical Models: Principles and Techniques. MIT Press.

[17] Kunze, L., Kumar, K., Hawes, N., 2014. Indirect object search based on qualitative spatial relations. In: IEEE International Conference on Robotics and Automation (ICRA 2014). Hong Kong, China.

[18] Le, D.-T., Uijlings, J., Bernardi, R., 2013. Exploiting language models for visual recognition. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Seattle, Washington, USA, pp. 769–779.

[19] Lowe, D. G., Nov. 2004. Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60 (2), 91–110.

[20] Mottaghi, R., Ranganathan, A., Yuille, A. L., 2011. A compositional approach to learning part-based models of objects. In: IEEE International Conference on Computer Vision Workshops (ICCV 2011 Workshops). pp. 561–568.

[21] Nister, D., Stewenius, H., 2006. Scalable recognition with a vocabulary tree. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Vol. 2. pp. 2161–2168.

[22] Nocedal, J., 1980. Updating quasi-newton matrices with limited storage. In: Mathematics of Computation. Vol. 35. pp. 2376–2383.

[23] Nowozin, S., Rother, C., Bagon, S., Sharp, T., Yao, B., Kohli, P., 2011. Decision tree fields. In: IEEE International Conference on Computer Vision (ICCV 2011). pp. 1668–1675.

[24] Nüchter, A., Hertzberg, J., 2008. Towards semantic maps for mobile robots. Robots and Autonomous Systems 56 (11), 915–926.

[25] Quattoni, A., Collins, M., Darrell, T., 2004. Conditional random fields for object recognition. In: Advances in Neural Information Processing Systems. MIT Press, pp. 1097–1104.

[26] Ren, X., Bo, L., Fox, D., 2012. Rgb-(d) scene labeling: Features and algorithms. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2012). pp. 2759–2766.

[27] Schmidt, M., 2015. UGM: Matlab Code for Undirected Graphical Models. `http://www.cs.ubc.ca/~schmidtm/Software/UGM.html`, [Online; accessed 28-April-2015].

[28] Silberman, N., Hoiem, D., Kohli, P., Fergus, R., 2012. Indoor Segmentation and Support Inference from RGBD Images. In: Proc. of the 12th European Conference on Computer Vision (ECCV 2012). Springer-Verlag, Berlin, Heidelberg, pp. 746–760.

[29] Speer, R., Havasi, C., 2013. Conceptnet 5: a large semantic network for relational knowledge. In: The People's Web Meets NLP. Theory and Applications of Natural Language. Springer, pp. 161—176.

[30] Uschold, M., Gruninger, M., 1996. Ontologies: principles, methods and applications. The Knowledge Engineering Review 11, 93–136.

[31] Valentin, J., Sengupta, S., Warrell, J., Shahrokni, A., Torr, P., 2013. Mesh based semantic modelling for indoor and outdoor scenes. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2013). pp. 2067–2074.

[32] Viola, P., Jones, M., 2001. Rapid object detection using a boosted cascade of simple features. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001). Vol. 1. pp. 511–518.

[33] Xiang, Y., Zhou, X., Liu, Z., Chua, T.-S., Ngo, C.-W., 2010. Semantic context modeling with maximal margin conditional random fields for automatic image annotation. In: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. pp. 3368–3375.

[34] Xiong, X., Huber, D., 2010. Using context to create semantic 3d models of indoor environments. In: In Proceedings of the British Machine Vision Conference (BMVC 2010). pp. 45.1–11.

[35] Zhou, K., Zillich, M., Zender, H., Vincze, M., 2012. Web mining driven object locality knowledge acquisition for efficient robot behavior. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2012). pp. 3962–3969.

Joint Categorization of Objects and Rooms for Mobile Robots

Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, Javier Gonzalez-Jimenez

Published in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2015.

Joint Categorization of Objects and Rooms for Mobile Robots[1]

J.R. Ruiz-Sarmiento, C. Galindo and J. Gonzalez-Jimenez

Machine Perception and Intelligent Robotics Group, System Engineering and Auto. Dept., University of Málaga, Campus de Teatinos, 29071, Málaga, Spain.

In general, the problems of objects' and rooms' categorizations for robotic applications have been addressed separately. The current trend is, however, towards a joint modelling of both issues in order to leverage their mutual contextual relations: *object* $\rightarrow$ *room* (e.g. the detection of a microwave indicates that the room is likely to be a kitchen), and *room* $\rightarrow$ *object* (e.g. if the robot is in a bathroom, it is probable to find a toilet). *Probabilistic Graphical Models* (PGMs) are typically employed to conveniently cope with such relations, relying on inference processes to hypothesize about objects' and rooms' categories. In this work we present a *Conditional Random Field* (CRF) model, a particular type of PGM, to jointly categorize objects and rooms from RGBD images exploiting *object-object* and *object-room* relations. The learning phase of the proposed CRF uses *Human Knowledge* (HK) to eliminate the necessity of gathering real training data. Concretely, HK is acquired through elicitation and codified into an ontology, which is exploited to effortless generate an arbitrary number of representative synthetic samples for training. The performance of the proposed CRF model has been assessed using the NYU2 dataset, achieving a success of $\sim 70\%$ categorizing both, objects and rooms.

1 Introduction

A robot performing in human environments has to manage a rich representation of its surroundings for the execution of tasks like navigation, fetch-and-carry, surveillance, etc. Such a world representation has to support the semantics of the human concepts and their relations. That is, the robot must be able to *understand* human knowledge, e.g. "A kitchen is a room where you can find an oven", permitting the human to express his/her orders using natural, and probably incomplete, sentences, e.g. "Please check the oven". The spatial awareness needed by the robot to accomplish this task must account for the existing close relations between objects and their typical locations. Thus, in this context, the robot should solve i) the so-called *room categorization* problem, i.e. to infer the type of space where it is, and ii) the *object categorization* problem, i.e. to classify the perceived objects.

Recent publications (e.g. [1, 2]) have shown that the joint modelling of the object and room categorization problems can outperform other methods that address them

[1]Work funded by the Spanish grant program *FPU-MICINN 2010* and the Spanish projects *TAROTH* (DPI2011-25483) and *PROMOVE* (DPI2014-55826-R), both co-founded by *Fondo Europeo de Desarrollo Regional*.

separately [3, 4, 5, 6]. Holistic approaches exploit the fact that objects are located in rooms according to their functionality, so the presence of an object of a certain type is a hint for the room categorization [7, 8, 9]. Likewise, the category of a room is a good indicator of the object categories that can be found inside [10]. Besides, objects are not placed randomly, but following configurations that make sense from a human perspective [11, 12]. Thereby, the exploitation of these object-object and object-room contextual clues provides categorization methods with useful information.

A recurrently resorted framework to model contextual information is the so-called *Probabilistic Graphical Models* (PGMs) [13]. PGMs permit a categorization system to conveniently model a room, the objects inside, and their contextual relations. Such a representation handles the uncertainty latent in the robot sensing system, and supports the execution of probabilistic inference algorithms (e.g. ICM [14] or LBP[15]). However, a significant drawback of these models is that they require a learning phase where the training dataset must be large and comprehensive enough to properly capture the variability of the domain at hand.

In this work we present a *Conditional Random Field* model (CRF) [13], a particular type of PGM, which enables the joint categorization of objects and rooms by exploiting their contextual relations. A distinctive feature of our approach is the utilization of *Human Knowledge* (HK) during the training phase, removing, thus, the arduous task of gathering real datasets. Concretely, we rely on the acquisition of HK about objects' and rooms' categories through elicitation and its codification into an *ontology* [16]. The advantage of using HK for training CRFs has been proven in [17].

Our approach has been tested with home RGBD scenes from the NYU2 dataset [18] (see figure B.1-left). This dataset is employed as a testbed by state-of-the-art methods given its size and challenging features. For example, it is utilized in [1], also employing a CRF, and achieving a success of $\sim 60.5\%$ and $\sim 58.7\%$ recognizing objects and rooms respectively. Although a fair comparison is not possible since the authors consider a different set of object categories and room types, it permits us to qualitatively confirm the promising performance of our approach, which yields a success of $\sim 70\%$ for categorizing both objects and rooms.

2 Conditional Random Fields. Application to the joint categorization of objects and rooms

The joint room and object categorization problem can be stated as the assignation of classes to both a given area of the robot workspace and the objects within, taking into account their observed geometric/appearance features and contextual relations. The following definitions are required in order to set the problem from a probabilistic stance:

- Let $o = [o_1, .., o_n]$ be a vector of n observed objects, each one characterized through a number of features: size, height, orientation, etc.
- Let r be the observed room described by a set of features: size, color, etc.

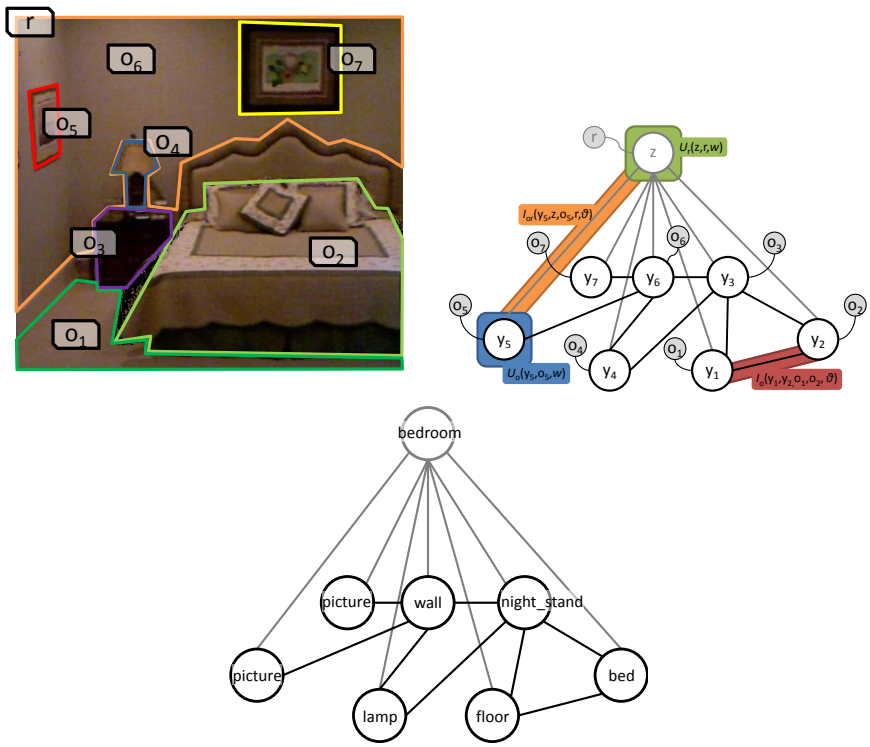


Figure B.1: Left, a coloured point cloud of a room (r) with a number of segmented objects (from o_1 to o_7), extracted from the NYU2 dataset. Right, the graph structure of a CRF modelling the objects in that room, the room itself, and their contextual relations. Each random variable y_i is associated to an observed object o_i, while z is related to r. The coloured parts indicate the scope of: an object unary factor – blue, a room unary factor – green, an object-object pairwise factor – red, and an object-room pairwise factor – orange. Bottom, result of a probabilistic inference process over the CRF.

- Define $L_o = \{l_{o_1}, .., l_{o_k}\}$ as the set of the k considered object categories (e.g. bed, oven, towel, etc.)
- Define $L_r = \{l_{r_1}, .., l_{r_j}\}$ to be the set of the j considered room categories (e.g. kitchen, bedroom, etc.).
- Define $y = [y_1, .., y_n]$ to be a vector of discrete random variables assigning a category from L_o to each object in o.
- Let $z \mid z \in L_r$ be a discrete random variable assigning a room category from L_r to r.

Thereby, the joint categorization process, modelled through a Conditional Random Field (CRF), consists of maximizing the probability distribution $P(y,z \mid o,r)$, i.e, to find the most probable room's and objects' categories given their characterized observations. CRFs exploit the concept of independence to break this distribution down into smaller pieces, since its high dimensionality prevents an exhaustive definition. A CRF is represented as a graph $H = (V,E)$, where V is a set of nodes representing random variables, and E a set of edges linking dependant/related nodes. In the addressed problem, a node represents a random variable, i.e. y_i or z, while an edge can set two types of dependencies: (a) between two close objects in the room, or (b) between an object and the room containing it. In figure B.1, an example of a relation of type (a) is the one between the *night stand* (o_3) and the *lamp* (o_4), while all the relations between the objects (from o_1 to o_7) and the room (r) are examples of relations of type (b). Thus, the categorization of an object affects the categorization of nearby objects, but not those placed far away, while the categorization of a room and its constituent objects has a mutual influence.

According to the Hammersley-Clifford theorem [13], the distribution $P(y,z|o,r)$ can be factorized over H as a product of factors, being a factor a function that represents a probability distribution over a part of H. In this work we have considered four factor types: two unary factors applicable to nodes (object and room unary factors), and two pairwise factors associated to edges (object-object and object-room pairwise factors).

For convenience, the factorization of $P(y,z|o,r)$ over the graph H is expressed by means of log-linear models as:

$$P(y,z|o,r,\omega,\theta) = \frac{1}{Z(o,r,\omega,\theta)}\, e^{-\varepsilon(y,z,o,r,\omega,\theta)} \tag{B.1}$$

where $Z(\cdot)$ is the normalizing partition function so $\sum_{\xi(y,z)} p(y,z|o,r,\omega,\theta) = 1$, being $\xi(y,z)$ an assignation to the variables in y and z, and $\varepsilon(\cdot)$ the so-called energy function, which in this work is defined as:

$$\varepsilon(y,z,o,r,\omega,\theta) = \sum_{i \in V_o} U_o(y_i,o_i,\omega) + U_r(z,r,\omega) + \sum_{(i,j)\in E_o} I_o(y_i,y_j,o_i,o_j,\theta) + \sum_{(i,j)\in E_{or}} I_{or}(y_i,z,o_i,r,\theta) \tag{B.2}$$

being V_o the subset of V containing the nodes associated to variables from y, E_o the subset of E entailing the edges that link nodes in V_o, and $E_{or} = E - E_o$, i.e. the edges connecting nodes representing objects with a room node. $U_o(\cdot)$, $U_r(\cdot)$, $I_o(\cdot)$ and $I_{or}(\cdot)$ define the employed factors (see figure B.1).

Object unary factor ($U_o(\cdot)$). This factor encodes the likelihood of assigning objects categories from L_o to the random variable y_i, given the features extracted from the object o_i, e.g. height, size, elongation, etc. It is defined as a linear classification model as follows:

$$U_o(y_i, o_i, \boldsymbol{\omega}) = \sum_{l \in L_o} \delta(y_i = l)\boldsymbol{\omega}_l f_o(o_i) \tag{B.3}$$

where $f_o(o_i)$ is a function that computes the features' vector f_{o_i}, $\boldsymbol{\omega}_l = [\boldsymbol{\omega}_{1,l}, .., \boldsymbol{\omega}_{|f_{o_i}|,l}]$ is a vector of weights for each class $l \in L_o$ obtained during the training phase, and $\delta(y_i = l)$ is the Kronecker delta function that takes value 1 when $y_i = l$ and 0 otherwise. The features used to characterize an object are: orientation, planarity, and size of its bounding box, area of its two principal directions, height from the floor, and color hue variation.

Room unary factor ($U_r(\cdot)$). The factor represented by the following linear model:

$$U_r(z, r, \boldsymbol{\omega}) = \sum_{l \in L_r} \delta(z = l)\boldsymbol{\omega}_l f_r(r) \tag{B.4}$$

encodes the likelihood of the random variable z to belong to the different room types from L_r given the features extracted from the observation r, e.g. size, number of objects, color hue variation, etc. In this case, $f_r(r)$ is the function that computes such a vector of features f_r, being the vector of weights $\boldsymbol{\omega}_l = [\boldsymbol{\omega}_{1,l}, .., \boldsymbol{\omega}_{|f_r|,l}]$ associated to the classes in L_r. The features used are: size of the room bounding box, number of objects within the room, and variation of color hue.

Object-object pairwise factor ($I_o(\cdot)$). Nodes related with objects that appear close in the scene are linked by an edge in the CRF. Thus, the object-object pairwise factor is in charge of stating the compatibility of a pair of categories assigned to these nodes. Again, a linear classification model is employed:

$$I_o(y_i, y_j, o_i, o_j, \boldsymbol{\theta}) = \sum_{l_1 \in L_o} \sum_{l_2 \in L_o} \delta(y_i = l_1)\delta(y_j = l_2)\boldsymbol{\theta}_{l_1,l_2} g_o(o_i, o_j) \tag{B.5}$$

where $g_o(o_i, o_j)$ computes a vector of features $f_{o_i o_j}$ to characterize the relation between objects o_i and o_j, and $\boldsymbol{\theta}_{l_1,l_2} = [\boldsymbol{\theta}_{1,l_1,l_2}, .., \boldsymbol{\theta}_{|f_{o_i o_j}|,l_1,l_2}]$ is a vector of weights, learnt during the training phase, for each pair of classes in L_o. The features characterizing object-object relations are: difference between principal directions, vertical distance of centroids, volume ratio, connectivity and object-object compatibility.

Object-room pairwise factor ($I_{or}(\cdot)$). This encodes the compatibility of finding an object of a certain category into a room of type l_{r_i}, as well as the compatibility of being in a room of a certain category having perceived an object of type l_{o_i}. Its linear classification model is defined as:

$$I_{or}(y_i, z, o_i, r, \boldsymbol{\theta}) = \sum_{l_1 \in L_o} \sum_{l_2 \in L_r} \delta(y_i = l_1)\delta(z = l_2)\boldsymbol{\theta}_{l_1,l_2} g_{or}(o_i, r) \tag{B.6}$$

being $g_{or}(o_i, o_j)$ a function that yields a fixed value $f_{o_i r}$. Therefore, the learnt vector of weights $\boldsymbol{\theta}_{l_1,l_2}$ for each pair of classes in $(l_1, l_2) \mid (l_1 \in L_o, l_2 \in L_r)$ states the object-room compatibility.

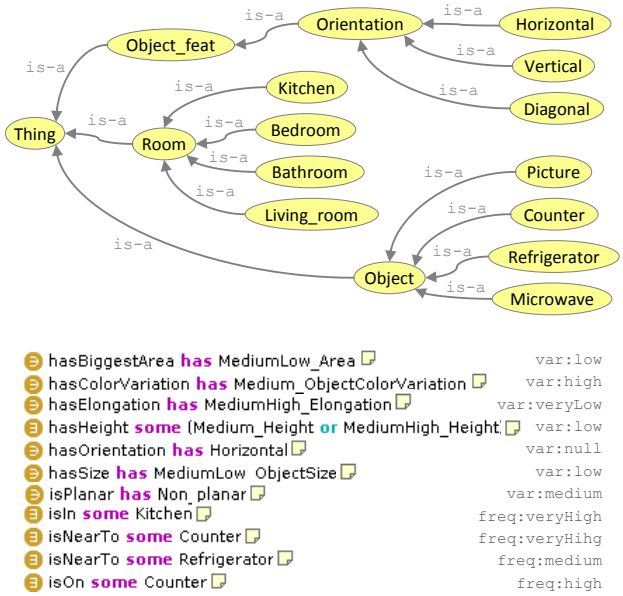


Figure B.2: Top, excerpt of the used ontology. Bottom, definition of the concept `Microwave`.

Training and Inference over the CRF. The training of the CRF model, i.e. the learning of the vectors of weights ω and θ, is performed by means of the optimization of the so-called pseudo-likelihood function, a tractable, alternative objective function to the computationally high-demanding likelihood one [13]. To fed this learning process we employ representative synthetic samples of the domain, which are generated as explained in the next section.

Once trained, the CRF is used to categorize rooms and objects through probabilistic inference. We resort to the Iterated Conditional Modes (ICM) algorithm [14], an efficient, approximated inference method that performs by maximizing local conditional probabilities. Figure B.1-bottom shows the results yielded by this method over the CRF of the figure B.1-right.

3 From Human Knowledge to training data

The proposed CRF model for the categorization of rooms and objects is tuned following a top-down design. First, knowledge of the domain at hand is collected through human elicitation. This information is codified into an ontology by means of the definition of concepts, e.g. `Kitchen`, and relations, e.g. `Microwave isIn Kitchen` (see section 3.1), and then exploited for the generation of an arbitrary number of representative, synthetic training samples (see section 3.2). The generated data feed an optimization process that iteratively tunes the CRF parameters defined in section 2.

On the other hand, the categorization process performs in a bottom-up fashion. Given a RGBD observation of a room, its constituent objects are segmented and characterized through a set of features (e.g. their size, height, etc.). The RGBD observation itself is also processed in order to characterize the room according to its geometry and appearance. Then, a number of object-object and object-room relations are computed according to the objects' features and locations. Finally, a probabilistic inference process over the trained CRF yields their most probable categories employing: i) objects' features, ii) room's features, and iii) contextual relations.

3.1 Codification of Human Knowledge

In this work we rely on human knowledge (HK) encoded in an ontology. An ontology is an explicit specification of a conceptualization related to a domain, which entails *concepts*, *relations*, and *individuals*. In the case of a home domain, examples of concepts are `Kitchen`, or `Microwave`, a relation can be stated as `Microwave isIn Kitchen`, and `kitchen-1` or `microwave-3` identify individuals, i.e. instantiations of concepts. The use of HK encoded in ontologies for mobile robotics exhibits significant advantages for a variety of applications, as reported in [19, 20].

Figure B.2-top depicts an excerpt of the ontology used during the conducted experiments, showing some concepts and relations[2]. Figure B.2-bottom shows the definition of the `Microwave` concept setting their usual features (geometry and appearance), as well as their contextual relations. It states, for example, that microwaves usually share a medium size, and are placed near counters, within kitchens. This information is collected from humans through an elicitation process, and it is straightforwardly codified into the ontology given its capability to naturally encode notions from natural language. Nevertheless, some human concepts need to be transformed into crispy values as required in our system. For that, the *hasValue* property is added in the ontology to quantify human concepts, like for instance `Vertical`, `Horizontal`, or `Diagonal`. These concepts allow an easy codification of object properties such as `Floor hasOrientation Horizontal` or `Picture hasOrientation Vertical`. The *hasValue* property assigns a crispy value to these concepts (in degrees) that is also gathered through elicitation, e.g. `Vertical hasValue 90`, `Horizontal hasValue 0`, and `Diagonal hasValue 45`.

In order to cope with the inherent variability of the considered domain, our approach annotates properties and relations with an element from the set R_A = *{null, veryLow, low, medium, high, veryHigh}*. For example in the definition of the microwave concept (see figure B.2-bottom) the size feature has been annotated with a `veryLow` variability indicating that most of microwaves exhibit similar dimensions. Similarly, these annotations are also used to express the frequency of the object-object and object-room relations. For example, the annotation `Microwave isNear Counter freq:high` sets that microwaves are usually found close to a counter,

[2]This ontology and other resources are available online at: http://mapir.isa.uma.es/work/objects-rooms-categorization.

Table B.1: Top, example of objects included in a room of type kitchen. Bottom, objects related with an included microwave.

Concept	frequency	$P(appearing)$	sample
`Bottle`	*medium*	0.5	not appearing
`Cabinet`	*veryHigh*	0.95	appearing
`Chair`	*medium*	0.5	not appearing
`Counter`	*veryHigh*	0.95	appearing
`Dishwasher`	*high*	0.8	not appearing
`Floor`	*always*	1	appearing
`Microwave`	*high*	0.8	appearing
`Picture`	*low*	0.2	not appearing
`Refrigerator`	*veryHigh*	0.95	appearing
`Stove`	*veryHigh*	0.95	appearing
`Table`	*medium*	0.5	not appearing
Concept	frequency	$P(related)$	sample
`Cabinet`	*high*	0.8	near
`Counter`	*veryHigh*	0.95	near
`Floor`	*veryLow*	0.05	not near
`Refrigerator`	*medium*	0.5	not near
`Stove`	*medium*	0.5	near

while the definition `Microwave isIn Kitchen freq:veryHigh` expresses that it is highly probable to find a microwave in a kitchen.

3.2 Generation of training data

Once the HK about the home domain has been encoded, we use it for the generation of synthetic training data. The presented process can be repeated to generate an arbitrary number of samples, and no human participation is longer required [17]. For clarity sake, it is explained the process for the generation of a synthetic sample reifying a kitchen, but the methodology is the same for any room category:

1. **Room characterization**. The first step is the computation of the room features which, in the used ontology, includes its size (m^3) and color hue variation. For that, a *Gaussian distribution* $\mathcal{N}(\mu,\sigma)$ is considered for each feature, where the mean μ corresponds to the crispy value of the property, while the standard deviation σ symbolizes the annotated variability. For example, given a definition of kitchens where they show a `Medium` size, being `Medium_RoomSize hasValue` 25, and an annotation of *medium* variability[3], the Gaussian distribution results $\mathcal{N}(25,5)$ (see figure B.3-left). The function $f_{sr}(l_r)$ draws a sample from this distribution to get the size of a particular room (see figure B.3-right), where l_r represents the kitchen category in this case, and repeats this process with the

[3]To get the standard deviation (σ) of a feature, the variabilities are considered to be a percentage of the crispy values of the properties that they are annotating within the ontology. In this case, being the crispy value 25, and corresponding *medium* to its 20%, the standard deviation is 5.

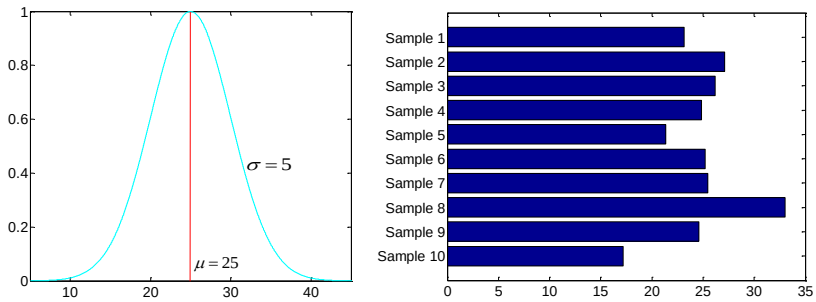


Figure B.3: *Left, unnormalized Gaussian distribution* for the size of a kitchen (in m^3) built according to its definition into the ontology. Right, samples drawn from that distribution to characterize the size of 10 kitchens.

remaining room features. This function replaces $f_r(r)$ in equation B.4 during the CRF training.

2. **Inclusion of objects in the room**. The inclusion of objects in the synthetic room is decided according to the `isIn` property. Only objects that contains the property `isIn value Kitchen` in their definitions are possible candidates. The inclusion of candidates depends on a probability distribution based on their frequency annotations. For example, the `Microwave` category is defined as `isIn value Kitchen freq:high`, which is translated to $P(Microwave_{appearing}) = 0.8$ and $P(Microwave_{notAppearing}) = 0.2$. Samples drawn from these distributions yield the final set of included objects, as it is illustrated in table B.1-top.

3. **Object characterization**. This step is similar to 1), but considering the properties defined over the objects included in the second step. A number of Gaussian distributions $N(\mu,\sigma)$ are built according to the different objects' geometric/appearance properties and their annotations, while the function $f_{so_i}(l_{o_i})$ draws samples from them to characterize each included object o_i. This function is used instead of $f_o(o_i)$ for learning the model (recall equation B.3).

4. **Object-object context creation**. The contextual relations between objects are established by the `isNear` properties and their annotations. In a similar way to the inclusion of objects, the likelihood of these relations is modelled by a probability distribution according to how frequently two objects appear close to each other in a Kitchen. For example, following the definition of the concept `Microwave`, they are often found near a counter, though it is more uncommon to find them near a table. As an illustrative example, table B.1-bottom shows the relations established for a microwave and the rest of objects included in a kitchen (in table B.1-top).

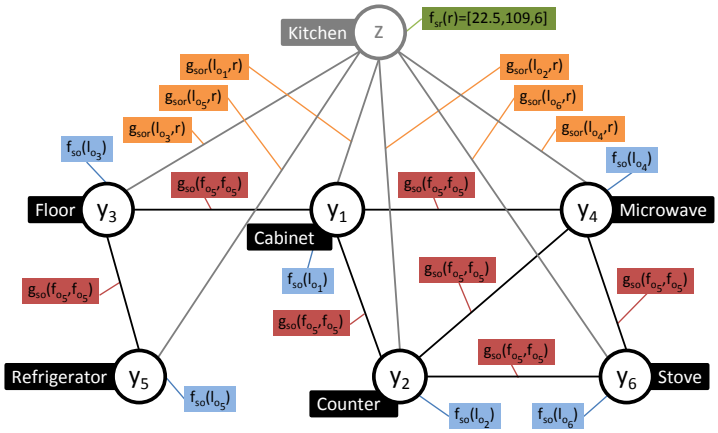


Figure B.4: Example of a CRF resultant from the generation of a synthetic room. The room type is a kitchen, with a total of 6 objects included (see table B.1). The resultant room's features are $f_{sr}(r) = [22.5, 109, 6]$, which correspond to its size, color hue variation, and number of objects.

5. **Object-object context characterization**. Different features can be computed to add valuable contextual information to the relations between two objects, e.g. difference of size, difference of height, perpendicularity, etc. These features can be easily computed from the object features extracted in the third step.

   In addition to these context features, two boolean properties are added: `isOn` and `isUnder`, which state if an object is placed on/under another one.

   The function in charge of compiling and yielding this information is $g_{so}(f_{o_i}, f_{o_j})$, being $f_{o_i} = [f_{so_i}(l_{o_i}), l_{o_i}]$, which replaces $g_o(o_i, o_j)$ in equation B.5.

6. **Object-room context characterization**. The relation between the room and its objects is characterized by a fixed value, as it is the training process of the CRF which learns automatically the likelihood of finding an object of a certain type into a kitchen. The function $g_{sor}(l_{o_i}, r)$ provides this value, and plays the role of $g_{or}(o_i, r)$ during training (recall equation B.6).

In summary, the above six steps yield the objects, room and contextual features needed to feed the unary and pairwise factors during the training of the CRF (equations B.3-B.6). Figure B.4 shows an example of a synthetic room represented in the form of a graphical model. It depicts the objects' and room's types, the functions in charge of characterizing them, and their contextual relations.

Table B.2: Method evaluation results.

	Our approach		Trained with real data	
Configuration	**Object**	**Room**	**Object**	**Room**
Appearance	17.86	27.88	17.79	27.66
Geometry	62.50	46.63	43.85	41.91
App.+geo.	63.87	50.96	47.70	47.22
App.+geo.+obj-obj	66.29	50.96	48.88	47.22
App.+geo.+obj-room	67.48	61.22	49.61	58.09
All combined	**69.61**	**69.71**	**56.08**	**62.65**

4 Evaluation results

In order to evaluate our approach, a number of CRFs have been tuned using synthetic samples (see section 3.2). These CRFs differ in the type of features and factors employed, aiming to contrast the performance achieved by different configurations.

We have resorted to the NYU2 dataset as a testbed, which is widely employed in the literature given its number of scenes and their diverse nature. Concretely, we have extracted 208 RGBD scenes resembling rooms perceived by a robot visiting a home environment, equally divided into four categories: *bathroom*, *bedroom*, *kitchen* and *living-room*. These rooms are compound of a total of 1692 objects belonging to 26 different categories provided by the dataset, including *bottle*, *sink*, *toilet*, *towel*, *sofa*, *bed*, *microwave*, etc.

In our experiments, the CRFs were trained with a dataset compound of 400 synthetic rooms, and their performance were measured by categorizing objects and rooms from the 208 NYU2 scenes. The implementation uses the Undirected Probabilistic Graphical Models library (UPGMpp) [21].

Table B.2 (left part) shows the results obtained for the different CRF configurations employing our model. Note how the integration of additional features and contextual relations progressively increases the performance. The first group of configurations only considers unary factors, the second one includes object-object or object-room pairwise factors, while the last integrates all of them. A closer look at the data reveals how the integration of object-object contextual relations boosts the performance in categorizing objects a 2.5% w.r.t. a configuration relying only on object local features (appearance and geometry), while the categorization of rooms increases a 10.2% if the object-room relations are considered. The combination of both contextual relations augments these figures to 5.7% and 18.7% respectively, which highlights the benefits of a joint categorization of objects and rooms. Examples of rooms and objects categorized by this last configuration are depicted in figure B.5-top. Figure B.5-bottom-right reports the rooms' confusion matrix for the last configuration, where rows represent the ground truth information and columns the categorization results.

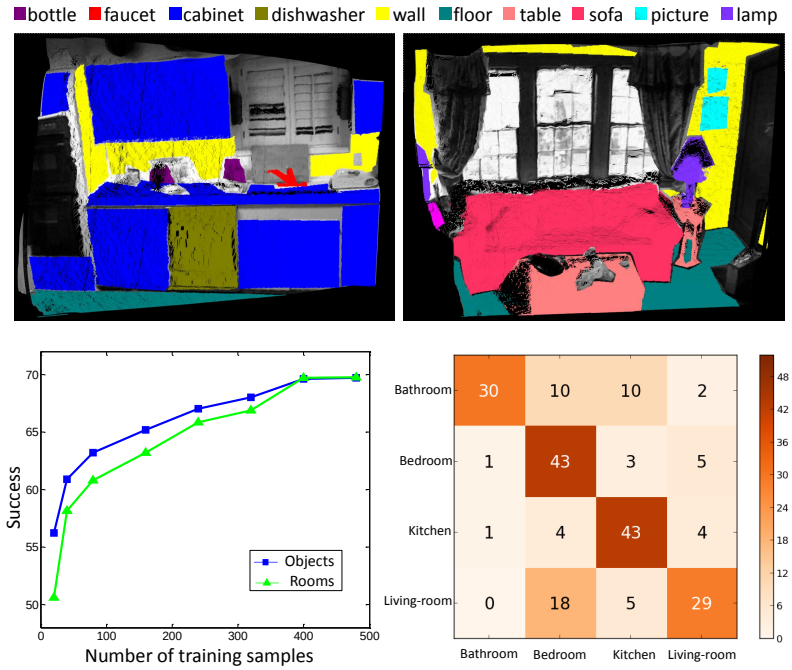


Figure B.5: Top, examples of a kitchen and a living-room correctly categorized as yielded by the method. Bottom-left, categorization success w.r.t. the number of samples used for training. Bottom-right, rooms' confusion matrix.

In order to validate the use of synthetic samples for training, the CRFs have been also trained and tested with the 208 NYU2 scenes following a 5-fold cross validation methodology. The results shown in the right part of table B.2 reveal that despite the positive effect of using contextual relations, these CRFs exhibit a lower performance.

Notice that the proposed training methodology based on HK permits a robot to effortless generate the training dataset, which size largely influences on the results. Figure B.5-bottom-left shows the categorization success yielded by a CRF trained with synthetic datasets of different sizes. It can be observed how the addition of more, representative training data boost the performance, from a 60.55% and 51.50% of success for object and room categorization respectively – 40 samples, up to 69.75% and 66.4% – 480 samples. This increment attenuates when the number of training samples approaches 500, which suggests that a success upper-limit can be reached despite the utilization of more samples. Notice that each training sample is compound of a room and its constituent objects so, for example, in the case of training with 480 rooms the number of objects is $\sim$4,900.

5 Conclusions and future work

This work has presented a *Conditional Random Field* (CRF) model to jointly categorize objects and rooms into the workspace of a robot. A key feature of this model is that we rely on *Human Knowledge* to replace real training data with prototypal, synthetic samples of the domain codified in an ontology, which removes the tedious and time-consuming task of gathering a real dataset. Additionally, the utilization of an ontology enables the execution of high-level robotic tasks. The approach has been validated against home scenes from the NYU2 dataset, reaching a categorization success of $\sim 70\%$ for both objects and rooms. It is worth to mention that the applicability of the approach is not limited to robots working at home environments, but it is suitable to perform in other domains which properties and semantics can be defined by human elicitation, e.g. office facilities or hospitals.

From here, we plan to endow the system with the capability to identify new categories of rooms and objects. A first step towards this could be the utilization of a logical reasoner over the yielded categorization results in order to check their coherence w.r.t. the set of defined objects and rooms within the ontology.

References

[1] Lin, D., Fidler, S., and Urtasun, R. (2013). Holistic Scene Understanding for 3D Object Detection with RGBD cameras. In *International Conference on Computer Vision (ICCV)*.

[2] Rogers, J.G., and Christensen, H.I. (2012). A Conditional Random Field Model for Place and Object Classification. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1766-1772.

[3] Felzenszwalb, P., Girshick, R., McAllester, D., and Ramanan, D. (2010). Object Detection with Discriminatively Trained Part-Based Models. *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627-1645.

[4] Oliva, A., Torralba, A. (2001). Modeling the shape of the scene: a holistic representation of the spatial envelope. In *Int. J. of Computer Vision*, vol. 42, pp. 145-175.

[5] Quattoni, A., and Torralba, A. (2009). Recognizing Indoor Scenes. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

[6] Pronobis, A., Martínez Mozos, O., Caputo, B., and Jensfelt, P. (2010). Multimodal Semantic Place Classification. In *International Journal of Robotics Research*, vol. 29, n. 2-3, pp. 298-320.

[7] Viswanathan, P., Southey, T., Little, J., and Mackworth, A. (2011). Place Classification Using Visual Object Categorization and Global Information. In *Canadian Conf. on Computer Robot Vision*, pp.1 -7.

[8] Pronobis, A., and Jensfelt, P. (2011). Hierarchical Multi-Modal Place Categorization. In *Proceedings of the 5th European Conference on Mobile Robots (ECMR'11)*.

[9] Espinace, P., Kollar, T., Soto, A., and Roy, N. (2010). Indoor scene recognition through object detection. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1406-1413.

[10] Torralba, A., Murphy, K. P., Freeman, W. T., and Rubin, M. A. (2003). Context-based vision system for place and object recognition. In *IEEE International Conference on Computer Vision (ICCV)*, pp. 273-280.

[11] Ruiz-Sarmiento, J.R., Galindo, C., and Gonzalez-Jimenez, J. (2014). Mobile Robot Object Recognition through the Synergy of Probabilistic Graphical Models and Semantic Knowledge. In *European Conf. of Artificial Intelligence*, CogRob workshop.

[12] Anand, A., Koppula, H.S., Joachims, T., and Saxena, A. (2013). Contextually guided semantic labeling and search for three-dimensional point clouds. In *Int. J. Robotic Res.*, vol. 32, n. 1, pp. 19-34, 2013.

[13] Koller, D., and Friedman, N. (2009). Probabilistic Graphical Models: Principles and Techniques. In *MIT Press*.

[14] Besag, J. (1986). On the statistical analysis of dirty pictures. In *Journal of Royal Statistical Society*, Series B (Methodological), pp. 259-302.

[15] Greig, D., Porteous, B., and Seheult, A. (1989). Exact maximum a posteriori estimation for binary images. In *Journal of the Royal Statistical Society*, Series B.

[16] Uschold, M., and Gruninger, M. (1996). Ontologies: principles, methods and applications. In *The Knowledge Engineering Review*, 11.

[17] Ruiz-Sarmiento, J.R., Galindo, C., Gonzalez-Jimenez, J. (2015). Exploiting Semantic Knowledge for Robot Object Recognition. In *Knowledge-Based Systems*.

[18] Silberman, N., Hoiem, D., Kohli, P., & Fergus, R. (2012). Indoor segmentation and support inference from RGBD images. In *Proc. of the 12th European conference on Computer Vision (ECCV)*, Vol. V.

[19] Galindo, C., Fernández-Madrigal, J-A, and González, J. (2008). Multihierarchical interactive task planning: application to mobile robotics. In *IEEE Trans. on Systems, Man, and Cybernetics*, Part B: Cybernetics.

[20] Galindo, C., Fernández-Madrigal, J-A, González, J., and Saffiotti, A. (2007). Using semantic information for improving efficiency of robot task planning. In *IEEE Int. Conf. on Robotics and Automation (ICRA)*.

[21] Ruiz-Sarmiento, J.R., Galindo, C., Gonzalez-Jimenez, J. (2015). UPGMpp: a Software Library for Contextual Object Recognition. In *3rd. Workshop on Recognition and Action for Scene Understanding*.

C

Scene Object Recognition for Mobile Robots Through Semantic Knowledge and Probabilistic Graphical Models

Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, Javier Gonzalez-Jimenez

Published in Expert Systems with Applications, 2015.

Scene Object Recognition for Mobile Robots Through Semantic Knowledge and Probabilistic Graphical Models

J.R. Ruiz-Sarmiento, C. Galindo and J. Gonzalez-Jimenez

Machine Perception and Intelligent Robotics Group, System Engineering and Auto. Dept., University of Málaga, Campus de Teatinos, 29071, Málaga, Spain.

Scene object recognition is an essential requirement for intelligent mobile robots. In addition to geometric or appearance features, modern recognition systems strive to incorporate *contextual information*, normally modelled through Probabilistic Graphical Models (PGMs) or Semantic Knowledge (SK). However, these approaches, separately, show some weaknesses that limit their application, e.g., the exponential complexity of the probabilistic inference over PGMs or the inability of SK to handle uncertainty. This paper presents *a hybrid PGM-SK system* for object recognition that integrates both techniques reducing their individual limitations and gaining in probabilistic inference efficiency, performance robustness, uncertainty handling, and providing coherent results according to domain knowledge codified by a human expert. We support this claim with an extensive experimental evaluation according to both recognition success and time requirements in real scenarios from two datasets (NYU2 and UMA-offices). The yielded figures support the suitability of the hybrid PGM-SK recognition system, and its applicability to mobile robotic agents.

Keywords: object recognition, semantic knowledge, probabilistic graphical models, mobile robotics, expert systems, autonomous agents.

1 Introduction

Mobile robots aiming to perform in human environments need both to conveniently represent information about its surroundings (i.e. managing a knowledge base) and to reason about it. In this paper we focus on the robot ability for scene object recognition which becomes crucial for the intelligent performance of the robot. In order to cope with such capability, a robotic agent must account for a knowledge base and an inference system able to manage the inherent *contextual relations* found in human environments, emulating thus, the decision-making capability of human beings. For instance, a black, thin and elongated object with buttons could be identified as a remote control or as a calculator. The exploitation of additional, contextual information helps to disambiguate the recognition [92, 19]: if one finds the object near to a notebook and a pen, the calculator option is the most plausible, while if it is on a sofa or close to a TV-set, the remote control option should be the expected result.

Being a valuable source of information, modern recognition systems strive to incorporate contextual information through different methodologies in order to boost their success.

A well-known framework to model contextual relations is *Probabilistic Graphical Models* (PGMs) [23], which provide a mathematical grounded mechanism to manage the inherent uncertainty of the robot's perception. In short, PGMs encode a knowledge base of the domain by learning a set of weights associated to the appearance/geometry of object classes, as well as to their usual relations with other object classes. Once this knowledge base is created, the learnt weights feed a *probabilistic inference process*, aiming at finding the most likely class-labelling for the perceived objects. A major drawback arises when the knowledge inherent in the domain at hand is complex, i.e., when the system has to deal with a high number of: i) possible object classes, ii) objects in the scene, and/or iii) features used to describe the objects and their context. In these cases, the inference process is computationally expensive, even intractable, due to its exponential nature. As an illustrative example, let us consider a scene with 8 objects to be recognized, which can belong to 10 different object classes. The application of *exact probabilistic inference* requires the computation of the likelihood of the 10^8 possible objects' labellings in order to find the most probable one. Approximate methods can be applied to cut down such complexity, like Iterated Conditional Modes (ICM) [5], Graph Cuts [7], or Loopy Belief Propagation (LBP) [27], but at the expense of compromising the system performance.

An alternative to PGMs is the use of *Semantic Knowledge* (SK) in the form of *ontologies* [39], which codifies the domain knowledge of a human expert through definitions of: concepts, like for instance `Calculator`, `Notebook`, etc., class attributes like `Calculator hasVolume small`, and contextual relations, like `Calculator isNear Notebook`. These definitions can be used by an expert system for object recognition employing logical reasoners, for instance Pellet [38], and their results can be directly exploited for further high-level robotic modules, like a task planner [14, 15]. An advantage of this approach is that semantic knowledge is common-sense, compact, and human-readable, facilitating in this way the information exchange between humans and robots. This methodology, though, entails some drawbacks: it is difficult to fill the gap between the low level sensor information and the SK base without introducing additional ad-hoc processes, and it is not suited to handle uncertainty.

In this work we present a *hybrid PGM-SK system* for object recognition that exploits the advantages of both techniques, while mitigating their individual drawbacks. Concretely, we exploit the synergy between Conditional Random Fields (CRFs) (a particular type of PGM), and SK encoded in an ontology. In this combination, the CRF provides the recognition system with the ability to manage uncertainty and fully exploit contextual relations through probabilistic inference, while SK offers a comprehensive representation of expert knowledge that contributes:

1. *A reduction of the CRF inference complexity*. The SK is used to generate hypotheses about the most probable belonging classes of the objects according to

their features. For example, an horizontal surface with a medium height from the floor could be hypothesized as belonging to the `Chair_seat`, `Table` or `Counter` classes, but not to `Wall` or `Computer_screen`. These hypotheses are then taken by the CRF as the only possible candidates. This leads to a considerable reduction in the number of combinations, decreasing the inference complexity, even enabling, in some cases, exact inference.

2. *Prior information about the frequency of occurrence of the different object classes*. Ontologies may encode different types of information, as for example, the occurrence frequency of object classes in a given domain. This information reveals that, for example, it is more likely to encounter a computer than a couch in an office environment, while it is quite unlikely to find an ironing table. A modification to the usual CRF formulation is proposed in this paper to exploit this prior information from the ontology.

3. *A ready-to-use representation for high-level reasoning tasks*. Interestingly from the AI point of view is the capability of the presented system to enable the direct exploitation of the gathered semantic information by high-level reasoning procedures, as reported in [14, 15, 16].

In this paper we extend our previous work [32], where this idea was initially explored as a proof-of-concepts, by including a thorough validation aimed to: i) demonstrate the suitability of our approach within a wider set of realistic environments and ii) conduct a deeper analysis of its performance under different situations. Concretely, we contribute:

1. *Tests with an additional state-of-the-art dataset, NYU2* [37], in addition to the initially resorted UMA-offices one. Such a dataset consists of 1,449 densely labelled RGB-D images gathered from a wide range of commercial and residential buildings employing a Kinect sensor. Contrarily to the UMA-offices dataset, scenes from NYU2 are limited to one-shot Kinect observations. This supposes an additional challenge since contextual information is confined to the narrow field-of-view of such cameras (57^o horizontal by 43^o vertical). Additionally, not only the planar patches employed in UMA-offices are considered, but regions with arbitrary geometries.

2. *An analysis of the feasibility of the exact inference*. In our previous work, preliminary results were obtained with the UMA-offices dataset, where the inference complexity reduction always enabled exact inference. This is not the case for the NYU2 dataset, where a higher number of classes and objects per scene are normally present. Thus, we have analysed when the complexity reduction is enough to enable exact inference conditioned to time consumption requirements.

3. *A study of two approximate probabilistic inference methods, namely ICM and Graph Cuts*, which can be applied whenever exact inference is not possible.

Concretely, we have studied the benefits of using SK for generating hypotheses in the performance of these state-of-the-art approximate approaches.

It is worth to mention that although the datasets considered in this work consist of colored point clouds, our system can be also adapted to other scene representations like RGB images.

Next section relates our approach with previous works in the field. Section 3 describes the application of PGMs for object recognition, while section 4 presents the proposed recognition system detailing how SK and PGMs are combined. Section 5 presents a thorough evaluation of such a system and its suitability for mobile robotic applications. Finally, section 6 highlights some conclusions and future work.

2 Related work

Scene object recognition is a widely studied topic in computer vision and robotics. *Local object recognition systems*, i.e. those only relying on the features of the objects like their geometry or appearance, have traditionally focused the research efforts due to their acceptable performance. Objects' characterization widely differs in the literature, resulting in a broad range of available approaches. For example, the work in [41] uses an *integral image* representation to encode the objects' appearance. Other popular options are SIFT features [25], employed in works like [8], which show a relative invariance to translation, scale, rotation, illumination, or partial occlusion of objects, and SURF features, which are faster to retrieve and, as claimed in [22], even more robust against those image transformations than SIFT features. A different, type of feature is such of Local Binary Pattern (LBP), which is fast to compute and describe the texture of a given portion of an image [10]. Some recognition approaches have been built based on these primary features, like Mixture Models [6]. Another example of these approaches is The Bag of Words (BoW) one [28], which works with sparse vectors of occurrence counts of codewords/features [20]. There also exist works that study the automatic learning of low level features, e.g. using neuronal networks, as is the case of [3].

Despite the success of local recognition systems for certain applications, their integration into mobile robots can lead to ambiguous recognitions, i.e. they are prone to fail in identifying classes with similar features, as analysed in [30, 9, 17, 35]. This is mainly due to only rely on features of the objects themselves, disregarding valuable contextual information that is also available. Therefore, a significant, growing body of current research aiming to overcome this issue is considering contextual information of the scene objects in addition to their usually employed individual features, and a number of applications dealing with this source of information have came out, e.g. [43] or [34]. Some works have attempted to exploit this information by providing ad-hoc or preliminary solutions, like in [26], where the co-occurrence of objects appearing in distinct types of rooms are implicitly modelled. However, these works lack a consistent theoretical background, compromising among others their comparison, generalization, reusability, or scalability. Moreover, their output consists of a set of

objects' labels, which do not carry any semantic information profitable by high-level AI robotic components. Well grounded alternatives for modelling/exploiting contextual relations are *Probabilistic Graphical Models* and *Semantic Knowledge*, which are combined in the recognition system presented here with the goal of mitigating the aforementioned drawbacks and boosting their virtues.

2.1 Recognition systems based on Probabilistic Graphical Models

Probabilistic Graphical Models (PGMs) are suitable mathematical tools to model and exploit contextual relations between scene objects. The first works introducing PGMs relied on intensity RGB images, like [31] that extends the Conditional Random Fields (CRFs) framework for the recognition of parts of objects by incorporating hidden variables, or [44], which presents a novel discriminative CRF for tackling the Automatic Image Annotation problem. Both methods do not require any segmentation step since they respectively work with i) local scale-invariant features of intensity images, and ii) the information in the images as a whole. However, their potential for exploiting the geometric properties of objects and relations is limited by the use of intensity images.

The arrival of RGB-D sensors, like Kinect, opened a new horizon for a more suitable modelling of contextual information. For example, 3D point clouds are used in [45] to extract and classify planar patches into four-coarse classes: *clutter*, *wall*, *floor* and *ceiling*, through a CRF formulation and the approximate inference algorithm ICM [5]. The work presented in [2] relies on a model isomorphic to a Markov Random Field (MRF) to recognize 17 object classes in both office and home environments. In this case, the inference problem is tackled by a Graph Cuts method [7]. The same approximate algorithm is chosen in [40], where a mesh representation of RGB-D depth measurements is built, and whose mesh faces are classified by a CRF, and in [36], which resorts to a *octree* representation of the scene. In [42] a densely connected CRF is defined over a set of voxels extracted from RGB-D information, and the inference process is driven by a *mean field approximation* approach. These works have in common the utilization of approximate methods to reduce the PGMs' inference complexity, hence compromising the overall performance of the recognition system. Particularly related to us is the work in [1], where the authors derive an object-to-object contextual MRF model based on Flickr labels co-occurrence. The authors cut down the exponential search space by considering the results from a previous classifier, conditioning the robustness of the whole system to its performance.

In comparison with those methods, the CRF employed in this work is part of a more sophisticated recognition system, completed by a ontology encoding expert knowledge. This combination reduces the burden of the probabilistic inference, increasing the situations/scenes where the *desired* exact inference is applicable, as well as to increase the success of approximate methods when it is not. Moreover, the works found in the literature bet on a certain approximate inference approach, without further comparisons with other methods. Such study is performed in this paper, involving

two widely used approaches: ICM and Graph Cuts. Finally, involving expert knowledge in the system ensures the coherence of the recognition results according to the knowledge base encoded into the ontology, and makes them profitable by high-level reasoning tasks [14, 15, 16].

2.2 Semantic Knowledge for contextual object recognition

A different trend in the literature resorts to Semantic Knowledge (SK) for both recognizing objects and exploiting their contextual information. For example, the work described in [19] codifies contextual information in an ontology, combined with a set of rules defined with the Semantic Web Rule Language, to generate objects' candidate classes. These hypotheses are subsequently validated through a matching process with CAD models. Another example is [29], which defines a constraint network in Prolog to classify the main structural surfaces of buildings, i.e. walls, floors, ceiling and doors, using contextual relations like orthogonal, parallel, above, etc. In [14], data codified into an ontology about scene objects and their relations are used to infer new high-level information. The work introduced in [11] recognizes segmented regions that have been previously characterized through a set of features in RGB images. These features are defined in an ontology, and their usual values for the different object types are learnt by symbolic supervised machine learning tools. In this case, a specific procedure matches characterized regions with semantically defined concepts, but although the authors propose the use of contextual relations, they are neither defined nor exploited. An ontology is also used in [12] for the recognition of isolated objects and their subparts, which manually establishes the association between geometric features and numeric values. This ontology is populated through machine learning techniques like Perceptrons and Support Vector Machines.

A common characteristic of these approaches based on SK is that they show limitations in quantifying the uncertainty of their results, and in exploiting the encoded contextual relations. The proposed approach faces these issues through collaboration with a CRF, which provides the robotic agent with a recognition system endowed with a probabilistic inference mechanism, able to manage uncertainty and adequately exploit contextual relations.

3 Scene object recognition through Conditional Random Fields

From a probabilistic stance, the object recognition process can be formulated as follows. Let us have a scene with $x = [x_1, .., x_n]$ observed objects (see figure C.1-a), each one characterized by a vector of features $f_{x_i u} = [f_{x_i u_1}, .., f_{x_i u_m}]$ (e.g. height, area, etc.), and $L = \{l_1, .., l_k\}$ the set of classes to which the objects can belong to.. Let $y = [y_1, y_2, .., y_n] \mid y_i : L^k \rightarrow \{0,1\}^k$ be a vector of discrete random variables corresponding to the class assignment to $\mathbf{x}$. Thus, the recognition process consists of maximizing the joint probability distribution $P(y, \mathbf{x})$, i.e., to find the most probable

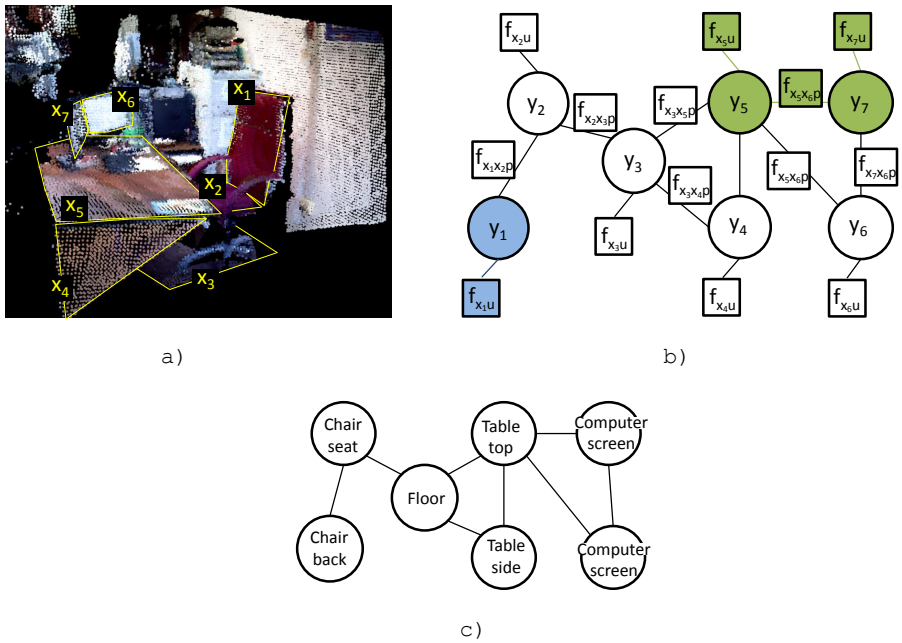


Figure C.1: a) Planar patches x extracted from a scene within the UMA-offices dataset, delimited by yellow lines. b) The CRF built for that scene, where each y_i is associated with its respective x_i, and conditioned by its extracted features $f_{x_i u}$ ($f_{x_i x_j p}$ stands for the pairwise or contextual features). Blue shapes represent an example of the scope of an unary factor, while green ones identify the scope for a pairwise factor. c) Recognition result obtained through probabilistic reasoning over the CRF.

classes assignation to y, also maximizing a number of probability distributions over the features extracted from x. Such a joint distribution has a high dimensionality, so its exhaustive definition is prohibitive. Probabilistic Graphical Models (PGMs) permits to break down such a definition into smaller pieces exploiting the concept of independence. To further simplify the problem, we employ a particular type of PGM called Conditional Random Field (CRF) [23], which factorizes the distribution $P(y|x)$, instead of encoding the probability distribution $P(y,x)$. This avoids the definition of the probability distributions over the object features extracted from x, which usually exhibits complex dependencies.

In general, a CRF is represented through a graph $H = \{V,E\}$, built upon two elements: a set of nodes V, and a set of edges E. Nodes V represent random variables, and edges E link nodes that keep some kind of relation, i.e., they are dependent. Typically, for modelling contextual information in visual object recognition, the nodes correspond to the random variables y, and two nodes y_i and y_j are connected if their associated objects x_i and x_j are close in the scene (see figure C.1-b). The rationale of

this is that the recognition of an object condition the recognition of nearby objects, but not those far away.

According to the Hammersley-Clifford theorem [23], the distribution $P(y|x)$ can be factorized over H as a product of factors, being a factor a function that represents a probability distribution over a part of H. In this work we use two kinds of factors: local and pairwise. *Local factors* refer to nodes, and express how probable is that an observed object x_i belongs to a certain class from L according to its extracted features. On the other hand, *pairwise factors* are associated to pairs of nodes, and codify the compatibility of the classes assigned to a given pair.

Concretely, we define an unary factor, denoted by $U(\cdot)$, as a linear classification model:

$$U(y_i, x_i, \omega) = \sum_{l \in L} \delta(y_i = l)\omega_l f(x_i) \tag{C.1}$$

where $f(x_i)$ is the function in charge of computing the features $f_{x_i u}$ for the object x_i, $\omega_l = [\omega_{1,l}, .., \omega_{f_m,l}]$ is a vector of weights for each class $l \in L$ obtained during the training phase, and $\delta(y_i = l)$ is the Kronecker delta function, which takes value 1 when $y_i = l$ and 0 otherwise. Table C.1-top shows the unary features used in this work. Note that we consider different features according to the characteristics of the segmented regions in each dataset, i.e., planar patches in the UMA-offices, and arbitrary-shaped regions in the NYU2 dataset.

On the other hand, a pairwise factor $I(\cdot)$ is defined as:

$$I(y_i, y_j, x_i, x_j, \theta) = \sum_{l_1 \in L} \sum_{l_2 \in L} \delta(y_i = l_1)\delta(y_j = l_2)\theta_{l_1,l_2} g(x_i, x_j) \tag{C.2}$$

where the function $g(x_i, x_j)$ computes a set of pairwise features $f_{x_i x_j p} = [f_{x_i x_j p_1}, .., f_{x_i x_j p_q}]$ capturing the relation between objects x_i and x_j (see the considered relations in table C.1-bottom), and $\theta_{l_1,l_2} = [\theta_{1,l_1,l_2}, ...\theta_{q,l_1,l_2}]$ is a vector of weights for each pair of classes in L.

The CRF training consist of the estimation of the vectors of weights ω and θ, which in this work is performed through the optimization of the so-called *pseudo-likelihood function* [23].

For convenience, the factorization of $P(y|x)$ over the graph H is expressed by means of log-linear models as:

$$P(y|x, \omega, \theta) = \frac{1}{Z(x, \omega, \theta)} e^{-\varepsilon(y, x, \omega, \theta)} \tag{C.3}$$

where $Z(\cdot)$ is the normalizing partition function so $\sum_{\xi(y)} p(y|x, \omega, \theta) = 1$, being $\xi(y)$ an assignation to the variables in y, and $\varepsilon(\cdot)$ the so-called energy function defined as:

$$\varepsilon(y, x, \omega, \theta) = \sum_{i \in V} U(y_i, x_i, \omega) + \sum_{(i,j) \in E} I(y_i, y_j, x_i, x_j, \theta) \tag{C.4}$$

Table C.1: Unary and pairwise features used to characterize regions and their relations for the two datasets used: UMA-offices and NYU2.

id	**Unary features** for UMA-office
$f_{x_i u_1}$	Centroid height from the floor.
$f_{x_i u_2}$	Orientation w.r.t. the horizontal.
$f_{x_i u_3}$	Area of its bounding box (b.b.).
$f_{x_i u_4}$	Elongation.

id	**Unary features** for NYU2
$f_{x_i u_1}$	Centroid height from the floor.
$f_{x_i u_2}$	Orientation w.r.t. the horizontal.
$f_{x_i u_3}$	Area of its b.b. biggest face.
$f_{x_i u_4}$	Minimum height of its b.b.
$f_{x_i u_5}$	Maximum height of its b.b.
$f_{x_i u_6}$	Volume of its b.b.
$f_{x_i u_7}$	Planarity.
$f_{x_i u_8}$	Linearity.
$f_{x_i u_9}$	Hue variation.

id	**Pairwise features for both datasets**
$f_{x_i x_j p_1}$	Perpendicularity.
$f_{x_i x_j p_2}$	on/under relation.
$f_{x_i x_j p_3}$	Vertical distance of centroids.
$f_{x_i x_j p_4}$	Ratio between areas.
$f_{x_i x_j p_5}$	Ratio between elongations.

Given an observation of the scene, the CRF graph $H = \{V, E\}$ is built according to the observed objects x and their proximity (objects at a distance below a given threshold are linked together), which sets the conditional dependencies between the random variables in y. Thus, the object recognition problem is that of finding the assignation to y that maximizes the posterior, that is:

$$\hat{y} = arg\max_{y} P(\mathbf{y}|\mathbf{x}, \omega, \theta) = arg\max_{y} \frac{1}{Z(\mathbf{x}, \omega, \theta)} e^{-\varepsilon(\mathbf{y}, \mathbf{x}, \omega, \theta)} \tag{C.5}$$

Given that the partition function does not depend on the assignments to y, such expression can be simplified by:

$$\hat{y} = arg\max_{y} e^{-\varepsilon(\mathbf{y}, \mathbf{x}, \omega, \theta)} \tag{C.6}$$

This equation is known as the Maximum a Posteriori (MAP) or Most Probable Explanation (MPE) problem, and it can be solved by exact or approximate inference methods. The next section describes how the exact inference method works and its limitations. Next, two widely used approaches for approximate inference are described as an alternative to the former.

3.1 Exact inference

Exact inference refers to a *brute force* technique where all the possible assignations to the variables in y are checked in order to find the labelling $\hat{y}$ that maximizes equation C.6. This method obviously ensures that the maximum to such an equation is

always found, as opposite to approximate methods that may yield a local maximum, or maximize a simplified version of the equation. Unfortunately, in real, complex scenarios this approach is unfeasible, because the number of assignations to be checked grows exponentially with the number of nodes in V (i.e., the number of objects to be recognized). For example, a scene with 10 objects that can belong to 14 different classes sum up a total of 14^{10} possible assignations. Thereby, the use of exact inference has been traditionally limited to simple, toy-problems or situations with undemanding time constraints, which is not the usual case in mobile robotic applications.

The presented hybrid PGM-SK approach is able to reduce the burden of the exact inference by exploiting Semantic Knowledge to generate hypothesis (see section 4.2). Additionally, in section 5, we have conducted an analysis of the conditions in which such a complexity reduction comes up with a feasible exact inference execution.

3.2 Approximate inference

When the problem at hand is complex, approximate probabilistic inference approaches can be employed instead of exact inference, which becomes intractable. The next subsections introduce two approximate methods intensively resorted by state-of-the-art recognition systems: Iterated Conditional Modes and Graph Cuts.

Iterated Conditional Modes

A widely employed method is the Iterated Conditional Modes (ICM, [5]), which maximizes local conditional probabilities instead of the whole $P(y|x)$. Briefly, ICM initializes the assignations to the variables in y to some initial object classes (usually to those that maximize the unary factors), and iterate over the variables following a pre-established order, changing such an initial labelling for the one that maximizes the following local conditional probability:

$$\hat{y_i} = arg\max_{y_i} P(y_i|y_{N_H(y_i)}, x_i, x_{N_H(y_i)}) \tag{C.7}$$

where $y_{N_H(y_i)}$ and $x_{N_H(y_i)}$ are sub-vectors of the original y and x vectors, and contain the random variables and observations of the neighbour nodes of y_i in the graph H. This algorithm ends when convergence is achieved, i.e., an iteration is completed without changing the state of any node, or when a given limit of iterations is reached. Thereby, termination is guaranteed by such an iterations upper limit, although the method usually needs only a few of them to converge. Figure C.1-c) shows the most probable classes assignation computed by ICM for the scene objects in figure C.1-a).

Graph Cuts

Techniques based on Graph Cuts [7], as for instance the α-expansion method, are well-known options for approximate inference. In a nutshell, this method simplifies the MAP task to instances of the minimum cut problem, which outcomes are used

to expand[1] each of the labels α in L until no expansion exists that produces a higher likelihood value.

Let V_α be the set of nodes assigned to the class α, and $V_{\bar{\alpha}}$ the nodes assigned to other classes. Thereby, this method relies on graph cuts in each iteration to compute the minimum cut of the graph $H_c = (V_c, E_c)$, where $V_c = \{V_{\bar{\alpha}}, s, t\}$, and $E_c = \{e_{ij} \mid (\mathbf{y}_i \neq \alpha) \cap (\mathbf{y}_j \neq \alpha)\} \cup \{e_{sk}, e_{kt}, \forall k \in V_{\bar{\alpha}}\}$[2]. Notice that for this computation two new nodes are added to V_c, which are usually called source (s) and sink (t). On the other hand, the set of edges E_α in the graph is compound of two groups: the edges in H between nodes that have not been assigned to class α, and the edges linking each of these nodes to both the source and sink nodes. The nodes connected to the source in the minimum cut produce an α-expansion to label α, while those linked to the sink keep their previous label. This process is repeated until no α-expansion can increment the likelihood, i.e. convergence is achieved, or until a maximum number of iterations is reached. As in the case of ICM, the execution of this method usually stops after a few iterations, and most of the class changes are done in the early steps.

4 Exploitation of Semantic Knowledge

The recognition system proposed in this work follows a bottom-up methodology (see figure C.2). During the robot operation, 3D observations gathered from a Kinect-like sensor are segmented[3], and their constituent regions are characterized through the set of features shown in table C.1. This information is exploited by the ontology to hypothesize the most probable class assignments for each region by means of logical inference[4]. These hypotheses dramatically reduce the number of potential classes to be considered by the CRF. Additionally, a modification to the usual CRF formulation has been carried out in order to also take advantage of prior information about the frequency of occurrence of the different object classes. Finally, the object recognition results are provided by probabilistic reasoning over a CRF, managing (i) a number of characterized regions from the scene, (ii) hypotheses about the most probable classes of each region, and (iii) prior information about the occurrence of classes.

The next section describes the codification of Semantic Knowledge through ontologies, while section 4.2 presents the use of a given ontology to provide hypotheses, and section 4.3 introduces our approach to integrate prior information in the traditional CRF formulation.

[1] An expansion is defined as a change in an object label from $\bar{\alpha} \in L_{-\alpha}$ to α.

[2] As previously mentioned nodes correspond to random variables. In this equation $\mathbf{y}_i$ and $\mathbf{y}_j$ are the random variables associated to the nodes v_i and v_j, respectively.

[3] In the case of the UMA-offices dataset, previously to their segmentation, such observations are registered together using the method presented in [13].

[4] In this work we use Pellet [38] as logical reasoner.

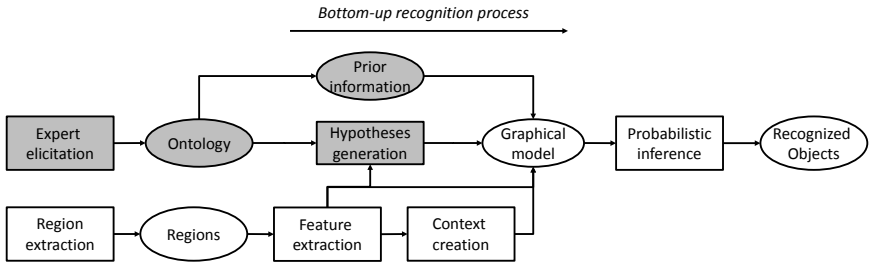


Figure C.2: Overview of the proposed system for object recognition. Boxes are processes, while ovals represent generated/consumed data by the processes. Gray shapes identifies the components that directly make use of semantic knowledge.

4.1 Ontology definition through expert elicitation

In this work we have opted for ontologies as a suitable and widely used semantic knowledge representation. An ontology is commonly defined as a representation of a conceptualization related to a knowledge domain, which accounts for a number of *classes* arranged hierarchically, *relations* among them, and *instances* of such classes, also called *individuals* [39]. One way to define ontologies is through *expert elicitation*, where experts in a certain knowledge domain codify their elements and relations. For example, with the appropriate tool any person having an acceptable background could model an office environment by defining the type of objects that usually appears in it (classes), e.g. `Table`, `Chair`, `Computer_screen`, etc., and establishing their contextual properties (relations), e.g. `Table hasOrientation Horizontal`. Relations can also set associations between classes, e.g. `Chair isNear Table`, which expresses that chairs are normally placed near tables. Knowledge about the objects from a particular scenario and their properties can be stated in the ontology through instances, e.g. `table-1`, `chair-1`, and instantiations of relations, "`table-1 isNear chair-1`". Figure C.3-bottom shows part of the ontology used to validate our work within the UMA-offices dataset, while figure C.3-top depicts, as an illustrative example, the definition of the class `Table_top` through a number of relations using the Protégè software [18]. This software codifies the resultant ontology into the OWL language [4].

The relations that characterize a class can be seen as properties, which are useful to describe the typical shape, size or relative position of its instances. For example, the relation "`Object has_area MetricMeasurement`" is used to codify the instances of the class `Object` that have an area of $MetricMeasurement$. The subclasses of `MetricMeasurement` discretizes real values into intervals, and have the form `MM_AroudXX`, which means that the measure is in the interval of the value `XX`. However, not all the instances of a class have the same appearance in the real world. To quantify that variability, properties describing the geometry of a class are annotated into the ontology with a discrete value from the set $R_A = \{null, veryLow, low, medium,$

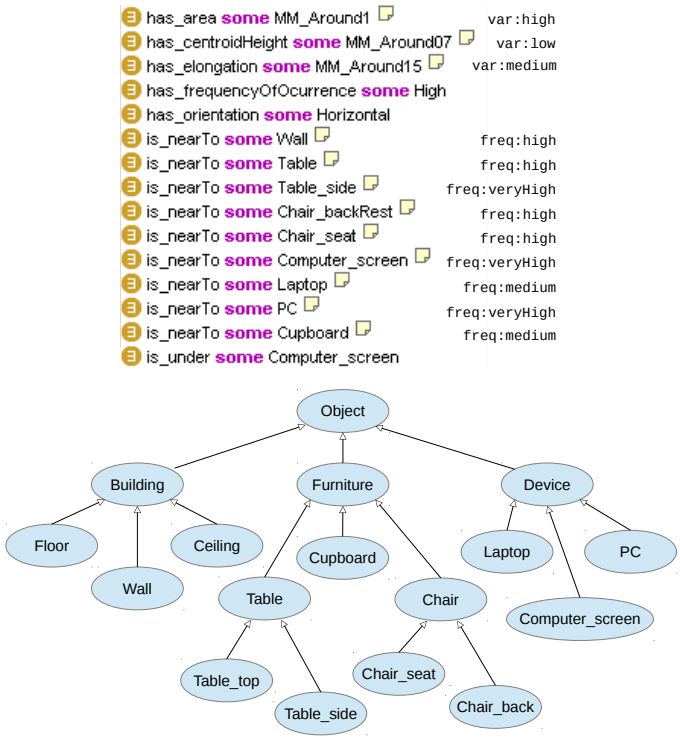


Figure C.3: Top, definition of the class `Table_top`. Bottom, part of the ontology defined by expert elicitation.

$high, veryHigh\}$. For example, the definition by an expert of the class `Table_top` in figure C.3-top, encodes that tables often share a common height around $0.70m$, although their area can largely vary around their averaged value, $1m^2$.

4.2 Hypotheses generation

As previously commented, one of the drawbacks of PGMs is the high computational complexity of the inference process even for a relative small number of objects in the scene and considered classes. The common solution is to rely on approximate inference methods, but jeopardizing the recognition results. Semantic information is used in this work to mitigate this effect by hypothesizing about the potential classes of the observed objects, reducing the complexity of the problem at hand. This hypotheses generation process is as follows.

Given the set $L = \{l_1, \ldots, l_k\}$ of the considered k classes of the domain, and a region x_i to be recognized, a new instance derived from the `Object` class is created

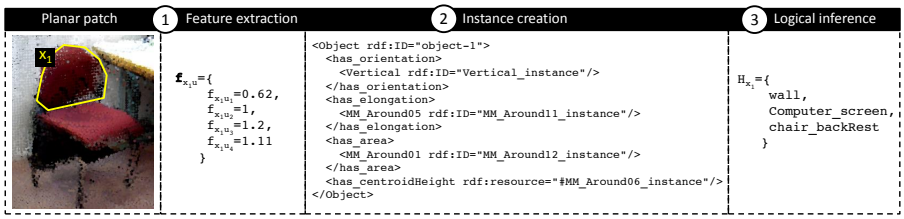


Figure C.4: Example of hypotheses generation for a given region. New instances are inserted into the ontology using the OWL language.

into the ontology, e.g. `object-1`, annotating its unary features $f(x_i) = f_{x_i u_i}$ based on the relations shown in figure C.3-top. For example, if a region has a centroid height of 0.73 meters from the floor, the relation "`object-1 hasCentroidHeight MM_Around07`" is added to the ontology. Once the instance is properly characterized, a logical reasoner, Pellet [38] in our implementation, infers a set of classes, $H_{x_i} \subseteq L$, which include such a relation in their definitions. This process is performed for all the n observed objects, obtaining a set of hypothetical classes $H = \{H_{x_1}, H_{x_2}, ..., H_{x_n}\}$ that contributes in reducing the complexity of the inference process, as shown in section 5. Figure C.4 shows an example of hypotheses generation for a `Chair_backRest`. It is important to underscore that the execution time of such a classification process is negligible in comparison to the time saved during the exact inference process (it takes just a few milliseconds in our tests). This is due to the size of the considered ontology, where the codified object properties are used to quickly accept or reject the hypotheses. An analysis of the cost of this process for larger and more complex ontologies is out of the scope of this paper[5], however, in each particular application this analysis should be conducted in order to measure the benefits of the utilization of the presented system.

According to the ontology definition shown in the previous section, the scene objects are hypothesized as belonging to concepts that strictly fulfil the defined geometric constrains. For example, a table with a centroid height of $0.6cm$ would not be hypothesized as a `Table_top` since the concept definition states `Table_top hasCentroidHeight MM_Around07` (see figure C.3-top). Thereby, in order to cope with the variability that objects may exhibit, after the expert elicitation process, the range of the geometric properties is modified according to their annotations[6]. As an illustrative example, let us consider the `Table_top` definition that encodes a "low" height variation from the average centroid height value (i.e. $70cm$). This semantic information is used to spread out the definition, widening the interval from $60cm$ to

[5]The computational cost of a classification process through logical inference heavily depends on the particularities of the ontology at hand, so it is difficult to quantify in general. See [21] for further information.

[6]Notice that these annotations could have been introduced as additional relations, e.g., `has_area_variability`. However, given that the logic reasoner is not going to take advantage of them, and aiming to have a representation as clear as possible, we opted for annotations.

80*cm*, codified as: "`has_centroidHeight some MM_Around06, MM_Around07 or MM_Around08`". Note that this process is automatically done by the system only once after the expert elicitation.

It is worth to mention that an additional advantage of using such hypotheses as class candidates is that the recognition results provided by the probabilistic reasoning over the CRF will be coherent with the information in the ontology, and consequently, with the semantic knowledge that the expert encoded about the domain.

4.3 Frequency of occurrence as prior

As commented in section 3, unary factors $U(\cdot)$ in a CRF give information about the compatibility of a certain object x_i w.r.t a set of classes H_i according to its appearance and geometry, which can be viewed as a way to model the probability distribution $P(y_i|x_i)$. On the other hand, pairwise factors codify their compatibility based on relational (contextual) features, and encode the distribution $P(y_i,y_j|x_i,x_j)$. In this section we propose the addition of information about the frequency of occurrence of objects to the CRF formulation as a prior probability, which helps in disambiguating the recognition results in certain situations. Thus, unary factors are given by the product of two probabilities, i.e:

$$U(y_i,x_i,\omega) \approx P(y_i|x_i,\omega)P(y_i) \tag{C.8}$$

Prior information is codified into the ontology through the relation `has_frequencyOfOccurrence`, which takes values from the set R_A. In order to adapt the probability distribution $P(y_i)$ to the linear classification model in equation C.1, it is replaced by the function $f_o(y_i) : R_A \rightarrow [0..1]$, which can be considered as a non-normalized version of the former probability. For example, if the class `Chair_back` is defined in the office domain with the relation "`Chair_back has_frequencyOfOccurrence veryHigh`", and `Computer_screen` with "`Computer_screen has_frequencyOfOccurrence medium`", the f_o function can be defined to produce $f_o(Chair\_back) = 0.9$ and $f_o(Computer\_screen) = 0.5$. Thus, we define an unary factor as follows:

$$U(y_i,x_i,\omega) = \sum_{l\in L} \delta(y_i = l)\omega_l f(x_i) f_o(y_i) \tag{C.9}$$

Conversely to the hypotheses generation case, here the function $f_o(\cdot)$ is independent of the scene, so it can be computed once and stored in a look-up table, hence speeding up the recognition process.

5 Evaluation of the proposed system

The proposed system has been thoroughly evaluated using two substantially different datasets: UMA-offices and NYU2. In both cases we have analysed the effects on the recognition success of: i) the inclusion of object contextual information together with

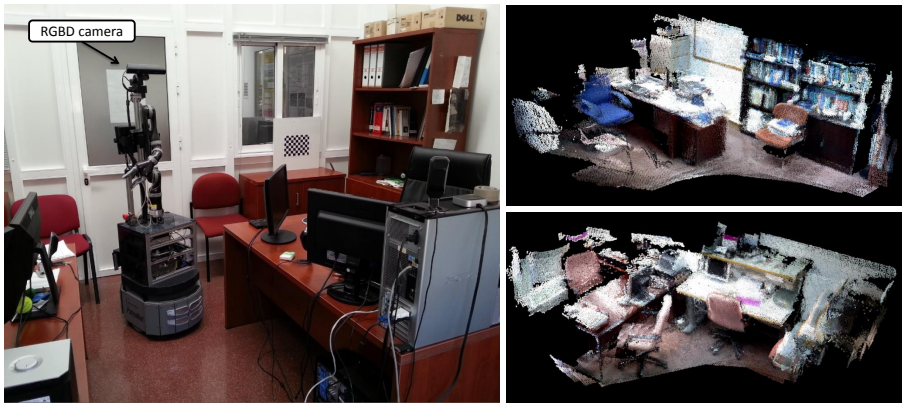


Figure C.5: Left, mobile robot Rhodon gathering 3D data from an office. Right, two sample office scenes from the UMA-Offices dataset.

the geometric and appearance features of objects, ii) the generation of hypotheses about the most promising object belonging classes, and iii) the addition of prior information about the frequency of occurrence of the different object classes. Additionally, for the second dataset, we have also studied the performance of the two approximate inference methods described above: Iterated Conditional Modes and Graph Cuts. The success of our approach has been measured using the micro precision/recall metrics.

5.1 Evaluation with UMA-offices

The UMA-offices dataset is compound of 25 office scenes, which where gathered using the mobile robot Rhodon. This robot is endowed with a Kinect-like sensor mounted on a Pan-Tilt unit, which permits the robot to perceive the world from a human-like point of view (see figure C.5). In this dataset, planar patches extracted from registered RGB-D images where manually labelled as belonging to 7 different object classes $L_{UMA-offices}$={`Floor`, `Wall`, `Table_top`, `Table_side`, `Chair_bakRest`, `Chair_seat`, `Computer_screen`}.

The planar patches extracted from the scene fed a CRF, which is trained following the approach in [33], and are also employed to generate hypotheses about their most probable classes, as described in section 4.2.

System performance

The second column of table C.2 shows the results obtained using 4 different recognition variants on the 25 considered scenarios. The first variant only uses appearance and geometric object features, which are modelled through unary factors (recalls equation C.1), achieving a micro p./r. of $\sim 79\%$. Contextual relations are integrated in the second variant by the addition of pairwise factors (recalls equation C.2), in-

Table C.2: Results of the tests conducted with the UMA-offices and NYU2 datasets.

Variant used	UMA-offices	NYU2
(1) No context	79.23	53.85
(2) Context	84.07	59.12
(3) Context + Hypotheses	93.45	61.25
(4) Context + Hypotheses + Prior	94.31	65.10

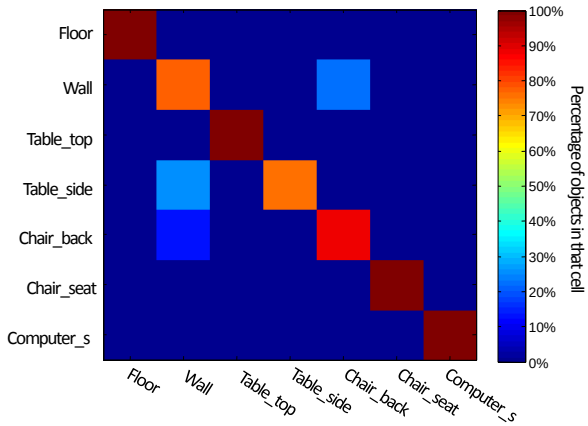


Figure C.6: Confusion matrix of the actual object classes and the recognition results.

creasing that percentage by $\sim 5\%$. The third variant incorporates the generation of hypotheses, reaching a micro p./r. of $\sim 93.5\%$, and the last one also uses prior information, obtaining $\sim 94.3\%$ of success. These results proves that contextual information improves the recognition of objects in a scene, and that the use of semantic information prevents the CRF from providing non-coherent results, increasing thus the recognition success. Prior information also adds a sense of coherence to the system operation by yielding the frequency of occurrence of the different object classes in an office environment, which is reflected as an improvement in the results.

Figure C.6 shows the confusion matrix obtained using the last variant, where the rows represent the actual class of the objects, and the columns the class to which they have been assigned. We can notice that erroneous recognitions correspond to the classes `Wall`, `Table_side` and `Chair_back`, given that, with the considered features, it is sometimes difficult to differentiate them.

It is worth to mention that the recognition system also yields the probability associated to the results as a measure of uncertainty. Thus, results with high uncertainty could motivate the execution of further actions by the robot, like gathering additional data from the scene, in order to reduce such uncertainty.

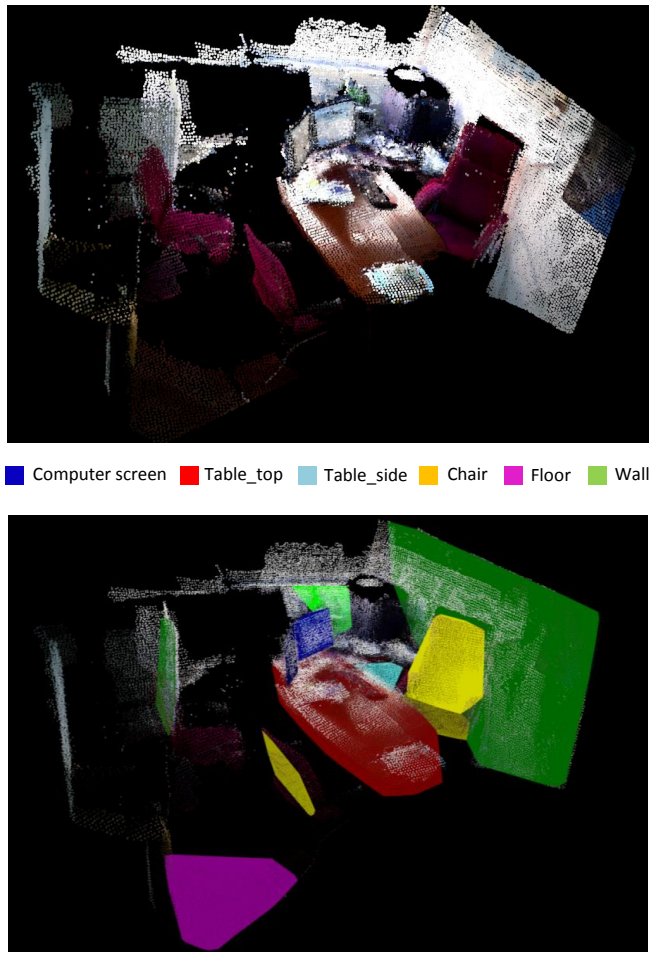


Figure C.7: Recognition result in one of the studied scenarios from UMA-offices. Top, 3D data from an office environment. Bottom, planar patches detected and recognized using our approach.

Complexity reduction

The above improvements are to a great extend due to the generation of hypotheses which allow the execution of *probabilistic exact inference*, i.e., checking all the possible class assignations for the scene objects. To illustrate this, let us consider the scenario shown in figure C.7-top, entailing 11 objects. Given that we have considered 7 object classes, probabilistic reasoning by exact inference consists of computing equation C.3 a total of 7^{11} times. Such a computation would take several hours, which is unfeasible for a mobile robot aimed to operate within real environments.

However, relying on the generated hypotheses as candidate classes, the number of combinations is reduced, in this example, to 1536, which can be computed in a few milliseconds. Figure C.7-bottom shows the objects from C.7-top recognized through an exact inference process.

5.2 Evaluation with NYU2

The NYUv2 dataset [37] is a large collection of RGB-D images from different indoor scenarios within commercial and residential buildings. From them, we have selected the office scenes, which sum up a total of 61, and have considered a total of 14 different object classes $L_{NYU2} = \{Cabinet, Ceiling, Chair, Computer, Desk, Floor, Keyboard, Light, Monitor, Mouse, Printer, Sofa, Table, Wall\}$. As in the tests with the UMA-offices dataset, the CRF weights were trained through synthetic samples following the methodology presented in [33].

System performance

The third column of table C.2 shows the results yielded according to the different variants. These results are given by an exact inference process when feasible, and by the ICM approximate method when it is not (see the following sections for further detail). Without exploiting contextual information, the CRF reached a success of $\sim$ 53.8%, while with its integration it augments to $\sim$ 59.1%. These numbers are quite similar to the ones provided in [24] (a 58.92% with the second configuration), which are outperformed by the following variants: a $\sim$ 2% of increment is obtained with the inclusion of the hypotheses generation step, and the final 65.1% is achieved by also considering prior information about the frequency of occurrence of the different object classes. Figure C.8 shows some examples of office scenes with the recognition results produced by the presented approach. These numbers support our claim that the inclusion of contextual information, hypotheses, and prior knowledge, significatively increment the system performance when compared to a method that only relies on features of the objects.

Note the drop in performance experimented with respect to the results obtained for the UMA-offices dataset which are mainly due to the following factors:

- *A limited contextual information*. The NYU2 dataset only includes isolated, one-shot RGB-D observations, and thus the contextual information is more limited than in the UMA-offices dataset, where several observations were registered together and a largest portion of the scene was processed.

- *A higher intraclass geometric variability*. The labelling provided by the NYU2 dataset contains a high number of spurious measures that, even after removing most of them through different filters, negatively affect the objects' geometric characterization. Thereby, objects of the same class exhibit quite different geometric features.

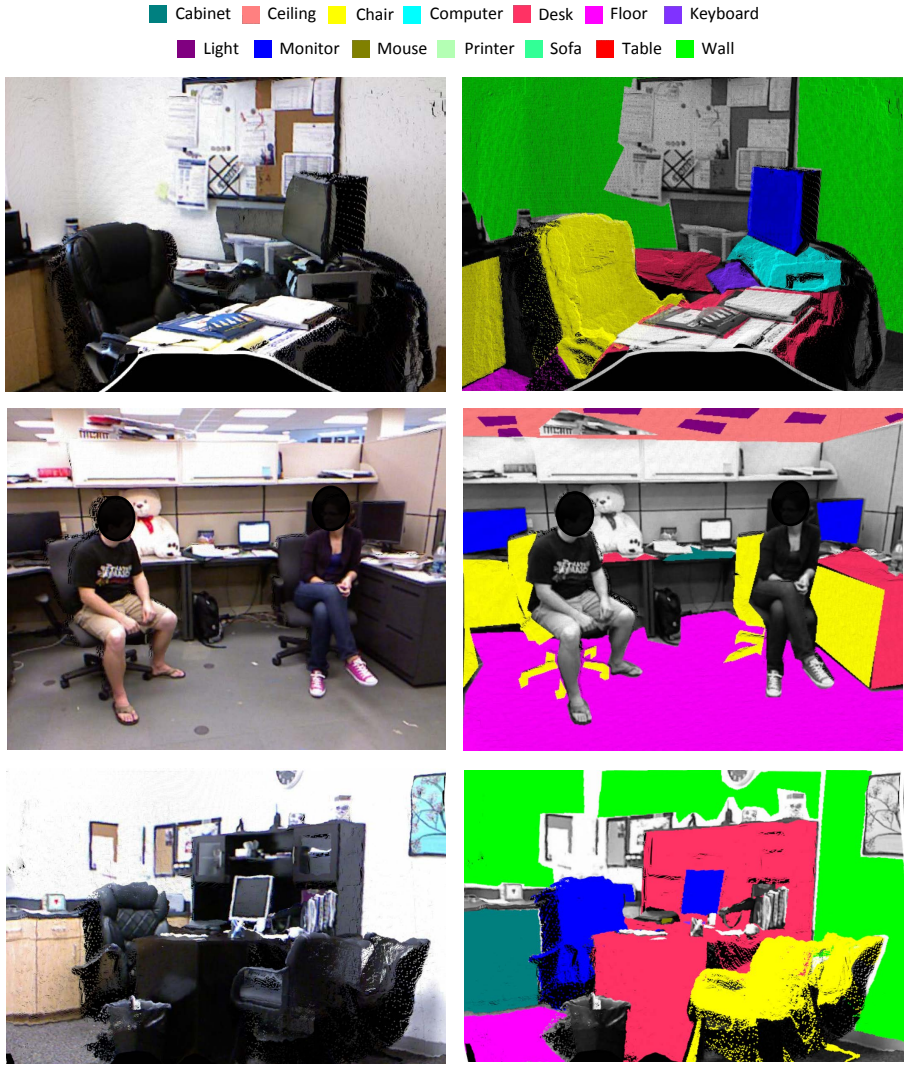


Figure C.8: Some scenes from the NYU2 dataset (left column) and the results yielded by our recognition system (right column).

- *The number of considered classes*. Tests with the NYU2 dataset consider twice the number of classes (from 7 to 14). Additionally, the average number of objects per scene is considerably higher (from ~ 6.8 to ~ 12.3). These factors compromise the application of the exact inference, resulting in a performance reduction.

Table C.3: Number of tests and time consumption (in seconds) necessary to check all the possible combinations of classes of the objects into a scene demanded by an exact inference process. Such numbers are conditioned by the amount of objects into the NYU2 scene, and were taken considering 14 object classes.

#objects	2	3	4	5	6	7	8	9
#tests	196	2,744	3,841e1	5,378e2	7,529e3	1,054e5	1,475e6	2,066e7
reduced #tests	9	68	509	3,774	2,798e1	2,074e2	1,538e3	1,140e4
time	3.5e-3	0.049	0.691	9.680	135.5	1,897	2,656e1	3,718e2
reduced time	1.6e-4	1.2e-3	9.1e-3	0.067	0.503	3.734	27.69	205.31

Complexity reduction

The reduction of the probabilistic inference complexity is critical for improving the system performance within the NUY2 dataset, due to the aforementioned increment in the number of both considered classes and average number of objects per scene with respect to the UMA-offices dataset. In the best case, the complexity reduction enables exact inference, although this highly depends on: i) the number of possible belonging classes, ii) the number of objects to be recognized, and iii) time constraints.

An analysis has been carried out considering 14 object classes ($\mid L_{NUY2} \mid$), and the exact inference feasibility is studied depending on the number of objects in the scene. Table C.3 shows the results of this analysis[7]. The difference between the first and the second row illustrates the achieved reduction in the number of possible labellings to be considered by exact inference. This clearly speeds up such a process, as shown by the difference between the third and fourth rows. An additional discussion could arise about the assumable time constraints for the object recognition system. In our experiments, an execution time of 3,7 seconds is permitted, but this figure could change depending on the tasks to be carried out by the robot. In conclusion, it is clear that without considering the hypotheses generation, exact inference quickly becomes unfeasible, but thanks to the proposed approach, its execution can be assumable, in our tests, for scenes containing up to 7 objects.

A remaining question is if the use of exact inference after the hypotheses generation step really increases the performance of approximate inference methods. To check this, we have performed both exact inference and approximate inference over a subset of the office scenes into the NYU2 dataset, employing the fourth variant in table C.2. Such a subset was compound by the scenes where exact inference was possible, concretely 27 out of the initial 61. The gathered results were clear: a $\sim 65.06\%$ of success was yield using exact inference, while the best number reported by the ICM and Graph Cuts methods was $\sim 61.57\%$.

[7]Execution times higher than one hour was extrapolated by the estimation of the number of tests to be performed and the time needed for one test, i.e., 1.8e-5 seconds in the computer onboard the robot.

Table C.4: Results yielded by the ICM and α-expansions approximate inference methods for the 61 office scenes from the NYU2 dataset. Two different configurations have been tested.

Variant used	ICM	α-expansion
(1) Context + Prior	63.22	62.03
(2) Context + Hypotheses + Prior	64.53	63.61

Approximate inference

Despite the possibility of carrying out exact inference under some circumstances, there are still cases where approximate probabilistic inference is needed. In section 3.2 we briefly described the ICM algorithm, which can be defined as an iterative process for maximizing local conditional probabilities. On the other hand, section 3.2 introduced the α-expansion method, an iterative methodology that executes Graph Cuts in each step. These methods[8] and have been widely used for object recognition purposes, so it is relevant to assess the influence of the generation of hypotheses in their performance.

To this end, we have collected the results of both methods over the 61 NYU2 office scenes, with and without the hypotheses generation feature. Table C.4 exposes the output of this study. It reveals that both methods benefit from the inclusion of hypotheses, increasing a $\sim 1.3\%$ the performance of the ICM method, and a $\sim 1.5\%$ the success of the α-expansions one. Both methods reach a good performance, but as shown in table C.2, even a higher success is achieved (65.10%) when executing exact inference on the 27 office scenes, and ICM over the 34 remaining ones.

6 Conclusions

The presented work has addressed the scene object recognition problem for mobile robotic agents by combining Probabilistic Graphical Models (PGMs) and Semantic Knowledge (SK) into a hybrid system. The proposed solution provides robustness against ambiguous scenarios, uncertainty handling, an improvement in the inference performance, and it produces coherent results according to the expert knowledge encoded in an ontology which can be exploited for other high-level Artificial Intelligent tasks. Robustness against objects showing similar features is achieved by the use of Conditional Random Fields (CRF), a particular type of PGM, that leverage contextual information between objects. Moreover, a modification to the usual CRF formulation has been presented in order to exploit prior information about the likelihood of finding an object of a certain class, which comes from the SK-base of the domain codified by an expert into an ontology. To reduce the probabilistic inference burden, the encoded ontology is used to hypothesize about the most promising belonging classes of

[8] We have relied on the ICM and α-expansion methonds' implementations within the *Undirected Probabilistic Graphical Models in C++* (UPGMpp) library [35]

the scene objects, cutting down the candidate options. This ensures, in addition, that the system outcome is consistent with such expert knowledge. Finally, the recognition results provided by the inference process can be instantiated within the ontology, which supposes a ready-to-use knowledge representation exploitable by other high-level tasks within the mobile robotic agent.

The claimed virtues of our recognition system have been thoroughly validated considering two substantially different datasets: *NYU2* and *UMA-offices*. The evaluation provides the performance of a local object recognition system as a baseline, revealing the progressive increment in the performance and robustness as long as additional information is exploited: contextual information, hypotheses of objects' classes, and prior information about object category occurrences. Moreover, an analysis of the complexity reduction of the probabilistic inference process has been carried out by considering the most promising object belonging classes, including the feasibility of exact inference for the considered datasets. The yielded results are promising, allowing the system to rely on exact inference in a wider variety of scenarios. It has been also studied the performance of two state-of-the-art approximate inference methods: Iterated Conditional Modes and Graph Cuts, which have shown to also benefit from the hypotheses generation. The obtained results reveal the suitability of our approach for highly-demanding applications, as is the case of mobile robot object recognition.

One source of weakness in the proposed system, which is common to other methods exploiting objects' relations, is the case of scenes where the profitable contextual information is reduced. In these situations the system performance can be compromised, although it would be higher than the expected one from a local recognition system thanks to the utilization of both, the hypothesis generation, and the prior information about the typical occurrence of object classes. One way to face this situation could be the integration of additional information by means of a more conscientious inspection of the scene by the robotic agent, e.g. active perception. Other challenge are scenarios where objects may not fit into their usual description, for instance recognizing a computer screen which is placed on the floor. In this case, any logical reasoner will not yield the class `Computer_screen` as result, given that its properties largely differs from the expected ones, i.e. it is found on the floor and not on a table, close to a keyboard. Our next goal is to increase the robustness of the presented system to deal with this issue. A solution could be to consider the result of the logical inference as a score in the CRF formulation, at the cost of compromising the *exact inference* option, or the periodic revision of the recognition results by an expert/human operator, which could recognize limitations in the encoded SK and fit them for a given domain.

An additional aspect to be explored is the inclusion of information about the room where the objects are found, aiming to build a holistic model that permits the system to recognize both objects and rooms' types taking into account their usual relationships.

Acknowledgements

This work has been funded by the Spanish grant program FPU-MICINN 2010 and the Spanish projects "TAROTH: New developments toward a robot at home" (Ref. DPI2011-25483) and "PROMOVE: Advances in mobile robotics for promoting independent life of elders" (Ref. DPI2014-55826-R). Both projects are co-founded by "Fondo Europeo de Desarrollo Regional - FEDER".

References

[1] Ali, H., Shafait, F., Giannakidou, E., Vakali, A., Figueroa, N., Varvadoukas, T., Mavridis, N., Feb. 2014. Contextual object category recognition for rgb-d scene labeling. Robot. Auton. Syst. 62 (2), 241–256.

[2] Anand, A., Koppula, H. S., Joachims, T., Saxena, A., Jan. 2013. Contextually guided semantic labeling and search for three-dimensional point clouds. In the International Journal of Robotics Research 32 (1), 19–34.

[3] Bai, J., Wu, Y., Zhang, J., Chen, F., 2015. Subset based deep learning for rgb-d object recognition. Neurocomputing 165 (0), 280 – 292.

[4] Bechhofer, S., van Harmelen, F., Hendler, J., Horrocks, I., McGuinness, D. L., Patel-Schneider, P. F., Stein, L. A., 2004. OWL Web Ontology Language reference. W3C Recommendation.

[5] Besag, J., 1986. On the statistical analysis of dirty pictures. Journal of the Royal Statistical Society. Series B (Methodological) 48 (3), 259–302.

[6] Bourouis, S., Mashrgy, M. A., Bouguila, N., 2014. Bayesian learning of finite generalized inverted dirichlet mixtures: Application to object classification and forgery detection. Expert Systems with Applications 41 (5), 2329 – 2336.

[7] Boykov, Y., Veksler, O., Zabih, R., Nov 2001. Fast approximate energy minimization via graph cuts. Pattern Analysis and Machine Intelligence, IEEE Transactions on 23 (11), 1222–1239.

[8] Chang, L., Duarte, M. M., Sucar, L., Morales, E. F., 2012. A bayesian approach for object classification based on clusters of SIFT local features. Expert Systems with Applications 39 (2), 1679 – 1686.

[9] Divvala, S., Hoiem, D., Hays, J., Efros, A., Hebert, M., June 2009. An empirical study of context in object detection. In: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. pp. 1271–1278.

[10] Dornaika, F., Bosaghzadeh, A., Salmane, H., Ruichek, Y., 2014. Graph-based semi-supervised learning with local binary patterns for holistic object categorization. Expert Systems with Applications 41 (17), 7744 – 7753.

[11] Durand, N., Derivaux, S., Forestier, G., Wemmert, C., Gancarski, P., Boussaid, O., Puissant, A., Oct 2007. Ontology-based object recognition for remote sensing image interpretation. In: Tools with Artificial Intelligence, 2007. ICTAI 2007. 19th IEEE International Conference on. Vol. 1. pp. 472–479.

[12] Eric Maillot, N., Thonnat, M., Jan. 2008. Ontology based complex object recognition. Image Vision Comput. 26 (1), 102–113.

[13] Fernandez-Moral, E., Mayol-Cuevas, W., Arevalo, V., Gonzalez-Jimenez, J., 2013. Fast place recognition with plane-based maps. In: IEEE International Conference on Robotics and Automation (ICRA 2013). pp. 2719–2724.

[14] Galindo, C., Fernandez-Madrigal, J., Gonzalez, J., Saffiotti, A., 2008. Robot task planning using semantic maps. Robotics and Autonomous Systems 56 (11), 955–966.

[15] Galindo, C., Fernández-Madrigal, J.-A., González-Jiménez, J., 2008. Multihierarchical interactive task planning. application to mobile robotics. IEEE Transactions on Systems, Man, and Cybernetics, part B 38 (3), 785–798.

[16] Galindo, C., Saffiotti, A., 2013. Inferring robot goals from violations of semantic knowledge. Robotics and Autonomous Systems 61 (10), 1131–1143.

[17] Galleguillos, C., Belongie, S., Jun. 2010. Context based object categorization: A critical survey. Computer Vision and Image Understanding 114 (6), 712–722.

[18] Gonçalves, R., Horridge, M., Musen, M., Nyulas, C., Tu, S., Tudorache, T., 2015. Protégè home page. http://protege.stanford.edu/, [Online; accessed 26-June-2015].

[19] Günther, M., Wiemann, T., Albrecht, S., Hertzberg, J., 2013. Building semantic object maps from sparse and noisy 3d data. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2013). pp. 2228–2233.

[20] Hoo, W. L., Lim, C. H., Chan, C. S., 2015. Keybook: Unbias object recognition using keywords. Expert Systems with Applications 42 (8), 3991 – 3999.

[21] Kang, Y.-B., Li, Y.-F., Krishnaswamy, S., 2012. Predicting reasoning performance using ontology metrics. In: Proceedings of the 11th International Conference on The Semantic Web - Volume Part I. ISWC'12. Springer-Verlag, Berlin, Heidelberg, pp. 198–214.

[22] Knopp, J., Prasad, M., Willems, G., Timofte, R., Van Gool, L., 2010. Hough transform and 3d surf for robust three dimensional classification. In: Proceedings of the 11th European Conference on Computer Vision: Part VI. ECCV'10. Springer-Verlag, Berlin, Heidelberg, pp. 589–602.

[23] Koller, D., Friedman, N., 2009. Probabilistic Graphical Models: Principles and Techniques. MIT Press.

[24] Lin, D., Fidler, S., Urtasun, R., 2013. Holistic scene understanding for 3d object detection with rgbd cameras. IEEE International Conference on Computer Vision 0, 1417–1424.

[25] Lowe, D. G., Nov. 2004. Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60 (2), 91–110.

[26] Mekhalfi, M. L., Melgani, F., Bazi, Y., Alajlan, N., 2015. Toward an assisted indoor scene perception for blind people with image multilabeling strategies. Expert Systems with Applications 42 (6), 2907 – 2918.

[27] Murphy, K. P., Weiss, Y., Jordan, M. I., 1999. Loopy belief propagation for approximate inference: An empirical study. In: Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence. UAI'99. pp. 467–475.

[28] Nister, D., Stewenius, H., 2006. Scalable recognition with a vocabulary tree. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Vol. 2. pp. 2161–2168.

[29] Nüchter, A., Hertzberg, J., 2008. Towards semantic maps for mobile robots. Robots and Autonomous Systems 56 (11), 915–926.

[30] Oliva, A., Torralba, A., Dec. 2007. The role of context in object recognition. Trends in Cognitive Sciences 11 (12), 520–527.

[31] Quattoni, A., Collins, M., Darrell, T., 2004. Conditional random fields for object recognition. In: Advances in Neural Information Processing Systems. MIT Press, pp. 1097–1104.

[32] Ruiz-Sarmiento, J.R., Galindo, C., González-Jiménez, J., 2014. Mobile robot object recognition through the synergy of probabilistic graphical models and semantic knowledge. In: European Conf. on Artificial Intelligence. Workshop on Cognitive Robotics.

[33] Ruiz-Sarmiento, J.R., Galindo, C., González-Jiménez, J., 2015. Exploiting semantic knowledge for robot object recognition. In: Knowledge-Based Systems.

[34] Ruiz-Sarmiento, J.R., Galindo, C., González-Jiménez, J., 2015. OLT: A Toolkit for Object Labeling Applied to Robotic RGB-D Datasets. In: European Conference on Mobile Robots.

[35] Ruiz-Sarmiento, J.R., Galindo, C., González-Jiménez, J., 2015. UPGMpp: a Software Library for Contextual Object Recognition. In: 3rd. Workshop on Recognition and Action for Scene Understanding.

[36] Sengupta, S., Sturgess, P., 2015. Semantic octree: Unifying recognition, reconstruction and representation via an octree constrained higher order mrf. In: Proceedings of IEEE International Conference on Robotics and Automation (ICRA). Seattle, WA, USA.

[37] Silberman, N., Hoiem, D., Kohli, P., Fergus, R., 2012. Indoor Segmentation and Support Inference from RGBD Images. In: Proc. of the 12th European Conference on Computer Vision (ECCV 2012). pp. 746–760.

[38] Sirin, E., Parsia, B., Grau, B. C., Kalyanpur, A., Katz, Y., Jun. 2007. Pellet: A practical owl-dl reasoner. Web Semantics: Science, Services and Agents on the World Wide Web 5 (2), 51–53.

[39] Uschold, M., Gruninger, M., 1996. Ontologies: principles, methods and applications. The Knowledge Engineering Review 11, 93–136.

[40] Valentin, J., Sengupta, S., Warrell, J., Shahrokni, A., Torr, P., 2013. Mesh based semantic modelling for indoor and outdoor scenes. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2013). pp. 2067–2074.

[41] Viola, P., Jones, M., 2001. Rapid object detection using a boosted cascade of simple features. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001). Vol. 1. pp. 511–518.

[42] Wolf, D., Prankl, J., Vincze, M., 2015. Fast semantic segmentation of 3d point clouds using a dense crf with learned parameters. In: Proceedings of IEEE International Conference on Robotics and Automation (ICRA). Seattle, WA, USA.

[43] Wong, Y.-S., Chu, H.-K., Mitra, N. J., 2015. Smartannotator an interactive tool for annotating indoor rgbd images. Computer Graphics Forum 34 (2), 447–457.

[44] Xiang, Y., Zhou, X., Liu, Z., Chua, T.-S., Ngo, C.-W., 2010. Semantic context modeling with maximal margin conditional random fields for automatic image annotation. In: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. pp. 3368–3375.

[45] Xiong, X., Huber, D., 2010. Using context to create semantic 3d models of indoor environments. In: In Proceedings of the British Machine Vision Conference (BMVC 2010). pp. 45.1–11.

UPGMpp: a Software Library for Contextual Object Recognition

Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, Javier Gonzalez-Jimenez

Published in the 3rd Workshop on Recognition and Action for Scene Understanding (REACTS), 2015.

UPGMpp: a Software Library for Contextual Object Recognition

J.R. Ruiz-Sarmiento, C. Galindo and J. Gonzalez-Jimenez

Machine Perception and Intelligent Robotics Group, System Engineering and Auto. Dept., University of Málaga, Campus de Teatinos, 29071, Málaga, Spain.

Object recognition is a cornerstone task towards the *scene understanding* problem. Recent works in the field boost their performance by incorporating contextual information to the traditional use of the objects' geometry and/or appearance. These contextual cues are usually modeled through *Conditional Random Fields* (CRFs), a particular type of undirected *Probabilistic Graphical Model* (PGM), and are exploited by means of probabilistic inference methods. In this work we present the *Undirected Probabilistic Graphical Models in C++* library (UPGMpp), an open source solution for representing, training, and performing inference over undirected PGMs in general, and CRFs in particular. The UPGMpp library supposes a reliable and comprehensive workbench for recognition systems exploiting contextual information, including a variety of inference methods based on *local search*, *graph cuts*, and *message passing* approaches. This paper illustrates the virtues of the library, i.e. it is efficient, comprehensive, versatile, and easy to use, by presenting a use-case applied to the object recognition problem in home scenes from the challenging NYU2 dataset.

Keywords: contextual object recognition, probabilistic graphical models, probabilistic inference, scene understanding

1 Introduction

Scene understanding systems aim to provide a valid interpretation of the perceived imagery which can be leveraged by a large variety of innovative technologies, like robotics, assistance to visual impaired, autonomous driving, etc. *Object recognition* is a key component of these systems, whose results become crucial for a proper understanding of the scene. Modern approaches improve the object recognition performance by incorporating contextual information of the objects, in addition to their usually employed geometry and/or appearance properties [1, 12, 13, 16, 15, 14, 20, 23]. This enables the disambiguation of confusing classifications provided by methods *only* relying on properties of the objects themselves [5]. Let's suppose, for example, a scene with a brown, cylindrical object. A method relying on geometric/appearance properties could have problems to classify it as a pot or a flowerpot, however, if it is found on a stove, the pot option is more probable.

The *Probabilistic Graphical Models* (PGMs) framework [7] has been widely used to exploit contextual relations among objects. Concretely, a particular type of PGM,

namely *Conditional Random Field* (CRF), has focused the interest of researchers given its suitability to model this kind of problems. PGMs integrate a compact and powerful graph-based representation of complex probability distributions defined over high-dimensional spaces, and employ probabilistic inference algorithms to efficiently perform queries of interest over it. Of particular concern is the *Maximum a Posteriori* query (MAP), since it provides the recognition results by computing the most probable category assignations to the scene objects[1]. The simplest MAP inference method, called *exact inference*, exhaustively tests all the possible objects' category assignations, which is an unfeasible approach in many real-world problems. Instead, approximate methods are exploited, which can be roughly classified into three major groups: local search [2], graph cuts [4], and message passing algorithms [9].

Most contextual-based object recognition works rely on an ad-hoc implementations of both the PGMs framework and inference algorithms [1, 12, 20, 23]. This makes it difficult to conduct a fair comparison between state-of-the-art works, even when they report results resorting to the same dataset [15]. There are some publicly available software libraries implementing this framework [11, 18], but they are not suited for the contextual object recognition problem (e.g. they only handle *chain-structured* models), or their applicability to this issue is limited.

This paper presents the *Undirected Probabilistic Graphical Models in C++* (UPGMpp) library, a software package for working with undirected PGMs, as is the case of CRFs, and its application to scene object recognition. UPGMpp exhibits a number of features that make it suitable for facing this particular problem: i) it works with discrete random variables, like the ones needed to model the possible objects' categories (e.g. chair, table, book, etc.), ii) it handles unary and pairwise relations, needed for representing the objects' features and relationships, and iii) it enables the representation of arbitrary structures, i.e. it can codify any number of scene objects and relations among them. This library implements inference methods from the three major groups mentioned above, including for example Iterated Conditional Modes (graph search), α-β swaps (graph cuts), or Loopy Belief Propagation (message passing). Therefore, UPGMpp provides a good basis for their evaluation and integration into recognition systems exploiting context. From an algorithmic point of view, the library also includes mechanisms to train PGMs and to perform probability queries (carry out marginal inference), as well as functionality for storing/loading PGMs from files through serialization. UPGMpp is designed to be efficient, versatile, extensible, and easy to use through clear and intuitive APIs, and resorts to well known libraries for numerical optimization (libLBFGS [10]), matrix operations (Eigen [6]) and memory handling (Boost [17]). It is entirely open-source, and is publicly available under a GNU General Public License (`http://mapir.isa.uma.es/work/upgmpp-library`). The library is distributed along with a number of code tutorials, so the user can master and start using it quickly.

As an illustrative example of its suitability to the contextual object recognition problem, we describe a use-case of recognizing objects from home scenes within the

[1] Along this paper we employ the term inference to refer to MAP inference.

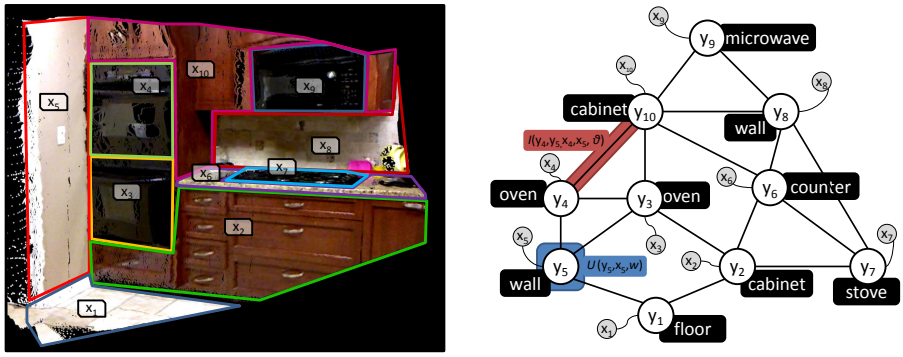


Figure D.1: Left, RGB-D image of a kitchen from the NYU2 dataset including the scene objects marked as $x = \{x_1, ..., x_{10}\}$. Right, CRF structure built from the scene. The blue shape represents the scope of an unary factor, while the red one states the scope of a pairwise factor. Random variables are labeled with the categories assigned by the execution of a probabilistic inference method over the CRF.

challenging NYU2 dataset [19]. Performance results regarding the execution time of inference and training methods within UPGMpp are also shown.

The next section describes the application of Conditional Random Fields to the scene object recognition issue, in order to provide a theoretical background for a better understanding of the library components. Then, section 3 presents the UPGMpp library, as well as the inference algorithms that it implements. Section 4 illustrates the UPGMpp application to the recognition of objects from scenes within the NYU2 dataset. Finally, section 5 outlines the conclusions and possible future work.

2 Contextual object recognition through Conditional Random Fields

The object recognition problem can be stated as the assignation of classes (e.g. table, chair, notebook, etc.) to a number of regions observed in imagery from a given scene. Let's consider the following definitions to address this problem from a probabilistic stance:

- Define $x = \{x_1, .., x_n\}$ as the set of n objects appearing in the scene, where each x_i is characterized through a vector of m features, $f_{x_i u} = [f_{x_i u_1}, .., f_{x_i u_m}]^T$, e.g. their size, color, orientation, etc.
- Let $L = \{l_1, .., l_k\}$ be the set of k possible object classes.
- Define $y = \{y_i, .., y_n\}$ as the set of discrete random variables over L, where each y_i assigns a class from L to its associated object x_i.

Thereby, the object recognition problem, modeled through a Conditional Random Field [7], is such of maximizing the probability distribution $P(y|x)$, i.e., to find the most probable classes' assignation from L to the random variables in y according to the characterized objects in x. The structure of a CRF is represented by a graph $H = (V,E)$, where V is a set of nodes associated to random variables, and E stands for a set of edges liking related variables/nodes. Regarding the problem at hand, a node represents a variable from y, and an edge connects two variables which associated objects are contextually related in the scene, i.e. they are placed close to each other. Figure D.1-left shows an scene with ten objects, which are represented as nodes in the CRF in figure D.1-right. We can see how, for example, the stove is related to the cabinet, the wall, and the counter, so their associated nodes are linked. Thereby, the probability distribution $P(y|x)$ can be factorized over this graph structure H, which is expressed for convenience by means of log-linear models [7]:

$$P(y|x,\omega,\theta) = \frac{1}{Z(x,\omega,\theta)} e^{-\varepsilon(y,x,\omega,\theta)} \tag{D.1}$$

where $Z(\cdot)$ is known as the partition function, so $\sum_{\xi(y)} P(y|x,\omega,\theta) = 1$, being $\xi(y)$ a possible assignation to the variables in y, ω and θ are vectors of weights learned during the CRF training, and $\varepsilon(\cdot)$ is the energy function, defined as:

$$\varepsilon(y,x,\omega,\theta) = \sum_{i \in V} U(y_i,x_i,\omega) + \sum_{(i,j) \in E} I(y_i,y_j,x_i,x_j,\theta) \tag{D.2}$$

being $U(\cdot)$ and $I(\cdot)$ the so-called unary and pairwise factors respectively. These factors can be seen as functions encoding small parts of the whole $P(y|x)$ over the nodes and edges of the graph H. Thus, an unary factor gives an intuition about how probable is for a node y_i to belong to a class from L according to the features of the object x_i. On the other hand, a pairwise factor speaks about an edge, and states the compatibility of two related variables being assigned a certain pair of classes from L. The scope of these factors is shown in figure D.1-right. They are defined by means of log-linear models as follows:

$$U(y_i,x_i,\omega) = \sum_{l \in L} \delta(y_i = l)\omega_l f_{x_i u} \tag{D.3}$$

$$I(y_i,y_j,x_i,x_j,\theta) = \sum_{l_1 \in L} \sum_{l_2 \in L} \delta(y_i = l_1)\delta(y_j = l_2)\theta_{l_1,l_2} f_{x_i x_j p} \tag{D.4}$$

where $\delta(y_i = l)$ is the Kronecker delta function, and $f_{x_i x_j p}$ is the vector of pairwise features characterizing the relationship between the objects x_i and x_j.

The training of a CRF consist of finding the vectors of weights ω and θ that maximize the likelihood function:

$$\max_{\omega,\theta} L_P(\omega,\theta : D) = \max_{\omega,\theta} \prod_{d \in D} P(y_d|x_d) \tag{D.5}$$

where D is the set of all the scenes used for training, compound each one of a set of characterized objects x_d, and their respective ground truth classes y_d. Solving Eq. D.5 requires the computation of the partition function, which is unfeasible in practise. Section 3.1 introduces the approaches implemented in the UPGMpp library to face this issue.

Once the CRF is trained, it can handle the execution of inference algorithms to contextually recognize objects. Thus, given a scene, its particular graph structure $H = (V,E)$ is built according to the relations shown by its constituent objects (see figure D.1). The (MAP) inference goal is to find the classes assignation $\hat{y}$ that maximizes the probability distribution $P(y|x)$ factorized over H, that is:

$$\hat{y} = arg\max_{y} P(\mathbf{y}|\mathbf{x}, \omega, \theta) \tag{D.6}$$

Again the computation of the partition function $Z(\cdot)$ is needed. However, since given a certain scene its value remains constant, this expression can be simplified by:

$$\hat{y} = arg\max_{y} e^{-\varepsilon(\mathbf{y}, \mathbf{x}, \omega, \theta)} \tag{D.7}$$

Despite this simplification, to compute an exact solution of such an equation is still unfeasible due to the huge number of possible assignations to be checked (k^n), which motivates the use of approximate inference methods. The algorithms implemented in the UPGMpp library for this are described in section 3.2.

3 UPGMpp library

The Undirected Probabilistic Graphical Models in C++ (UPGMpp) library is an open-source software for dealing with undirected PGMs, e.g. Markov Random Fields, or Conditional Random Fields. The library works with discrete random variables and handles local and pairwise relations, i.e. first and second order PGMs. UPGMpp provides tools for: i) defining graph representations, ii) completing a fast training of models, and iii) performing efficient inference queries (both probability and MAP queries). This section presents an overview of the most relevant features of the library and its components (section 3.1), as well as the available inference algorithms (section 3.2).

3.1 Overview

The UPGMpp library is divided into three packages (see figure D.2):

- ***base***. Implements the functionality for building and managing PGM graphs.
- ***training***. Permits the definition of training datasets to tune a PGM.
- ***inference***. Implements algorithms to perform probability and MAP inference queries over PGMs.

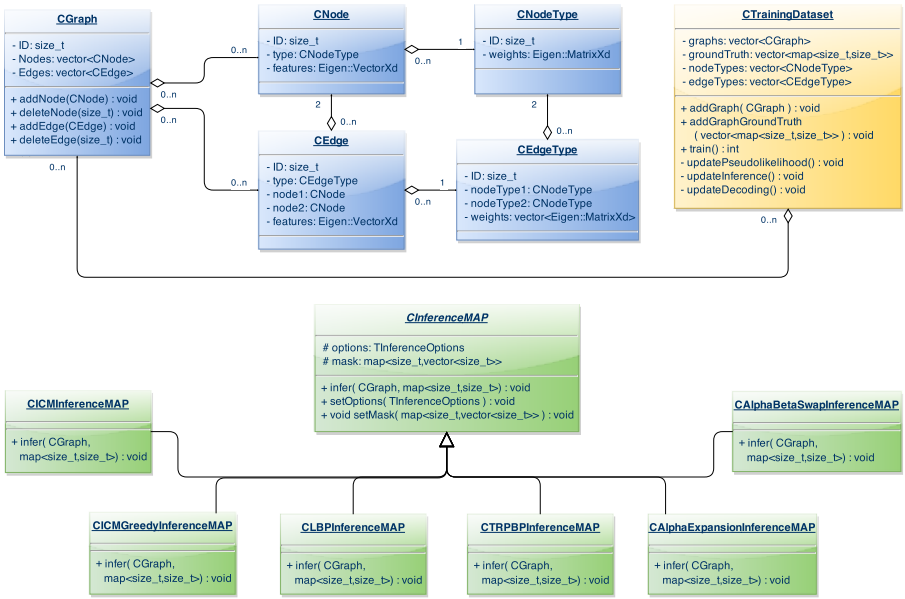


Figure D.2: Simplified UML class diagram of the main classes within the *base* (blue), *training* (yellow) and *inference* (green) packages within the UPMGpp library. For interpretation of references to color, the reader is referred to the web version of this work.

The *base package* provides an easy way to create and manage graphs representing PGM structures. Instances of nodes from V can be created employing the `CNode` class, as well as edges from E through the `CEdge` one. The `CNodeType` and `CEdgeType` classes permit us the definition of typed nodes and edges. Having the sets of nodes and edges, they can be inserted into an instance of the `CGraph` class, which represents the graph structure $H = (V, E)$. The factors within nodes (unary) and edges (pairwise) have been implemented through log-linear models (recall equations D.3 and D.4), although the user can easily define a different way to compute them through a prototype function.

The *training package* provides mechanisms for building datasets employing the `CTrainingDataset` class, i.e. sets of graphs along with their ground truth categories (see the *yellow* class and methods in figure D.2). Once created and populated, a dataset can be used to train an undirected PGM, i.e. to find the vectors of weights ω and θ in equation D.5. Recalling that the computation of such an equation is unfeasible in practice, two major approaches are considered in the literature: the definition of tractable alternative objective functions, like the *pseudolikelihood*, and the use of approximate inference processes (including MAP and marginal inference) [7]. Both approaches have been implemented and are available to the user in the *training package*.

Finally, the *inference package* implements a number of state-of-the-art inference algorithms for performing both, probability and MAP queries (recall equation D.7), although in this work we focus on MAP since it provides the scene object recognition results. To facilitate its use and future expansion, every MAP inference algorithm inherit the same functionality from a base class, `CInferenceMAP`, and implements the same abstract method for performing inference (see *green classes* in figure D.2). The implemented MAP inference methods are described in section 3.2.

UPGMpp resorts to the also open-source project libLBFGS [10] for performing numerical optimization, and the Eigen [6] library for performing fast matrix operations. The Boost library [17] is used to avoid unnecessary re-copy of data across the library methods by means of shared smart pointers. This library is also used for serialization purposes, which adds the possibility of storing/loading graphs from/to files, enabling the long-term life of PGMs beyond execution time.

3.2 MAP inference methods

This section briefly describes the theory behind the approximate MAP inference methods implemented in the UPGMpp library. The interested reader can refer to the provided citations for further information.

Local search methods.

Local search methods are the simplest approaches for approximated MAP inference, and they are widely used due to their easy implementation and acceptable results. In a nutshell, these methods operate over a set of candidate solutions called *search states*, which define a *search space*. In object recognition, a search state can be seen as a certain assignation $\xi(y)$ to the variables in y, which have an associated likelihood value, and the search space corresponds to the set of all possible assignations. Thus, starting at a certain state $\xi_c(y)$, a local search method checks if there is a state among the set of *similar states*, defined as $Sim(\xi_c(y))$, showing a higher likelihood value. If so, the algorithm *moves* to it as the current search state $\xi_c(y)$. Thereby, these methods perform small movements while exploring the search space, always increasing the expected likelihood, until a local maximum is reached, i.e. there is not a similar state to the current one with a higher likelihood. Algorithms within this group differ in how they define the similarity function $Sim(\xi_c(y))$ for a given state $\xi_c(y)$. Next, the *Iterated Conditional Modes* (ICM) local search method and its *Greedy* variant are described (see [2] for further detail).

Iterated Conditional Modes. ICM operates by giving an initial assignation to the variables in y, and iterating over those variables to maximize the local conditional probability:

$$\hat{y_i} = arg\max_{y_i} P(y_i | y_{N_H(y_i)}, x_i, x_{N_H(y_i)}) \tag{D.8}$$

where $y_{N_H(y_i)}$ and $x_{N_H(y_i)}$ are sub-vectors of the original y and x ones that contain the random variables and observations of the neighbor nodes of y_i in a certain graph H. Thus, being $\xi_c(\cdot)$ the current assignation to a set of random variables, the set of similar states is defined as $Sim(\xi_c(y)) = \{\xi(y) \mid \xi(y_{-i}) = \xi_c(y_{-i})\}$. This algorithm ends when convergence is achieved, i.e., an iteration over all the variables is completed without changing the search state, or when a given limit of iterations is reached.

Greedy ICM . The greedy variant takes the same initialization and ending criteria, but instead of performing a movement per random variable in y, it first iterates over all the variables, and then applies the movement that yields the maximum likelihood increment. In this case the set of similar states is defined as: $Sim(\xi_c(y)) = \{\xi(y) \mid diff(\xi(y), \xi_c(y)) = 1\}$, where $diff(\xi(y) - \xi_c(y))$ yields the number of random variables with different assigned classes, i.e. two states are similar if only one random variable in y shows a different assignation. This algorithm requires on average more iterations to converge than the original ICM, but it is more robust against getting stuck in a local maximum.

Graph cuts methods.

Graph cuts [4] have been extensively used to efficiently face early vision problems that can be formulated as a minimization of an energy function. This approach reduces the MAP inference task to instances of the minimum cut problem. Let's suppose a binary classification problem ($y_i = \{0,1\}$) with factors codified over a graph $H = (V,E)$. To apply graph cuts, the graph is modified in the following way: a pair of nodes, s (source) and t (sink), are added so $V_c = \{V,s,t\}$, and two edges linking each node with s and t are included, obtaining the set $E_c = \{E\} \cup \{e_{s \to i}, e_{i \to t}, \forall i \in V\}$. Then, the minimum cut of this new graph $H_c = \{V_c, E_c\}$ is computed, which divides the set of nodes into two sets: the one containing the nodes connected to the source s, called V_s, and the set of nodes V_t linked to the sink t. Finally, the nodes in V_s are classified as belonging to the class 0, and those in V_t to the class 1. This method can be extended to handle non-binary classification problems, as illustrate the α-β *swaps* and the α-*expansions* algorithms [3].

α-β swaps. This algorithm iterates over all the possible class pairs (α,β) in L, and checks if there is a swap among the variables assigned to that classes that increments the expected likelihood. Let $V_\alpha = \{V_i = \alpha, \forall i \in V\}$ be the set of nodes/variables assigned to the class α, and $V_\beta = \{V_i = \beta, \forall i \in V\}$ those assigned to β. Then, graph cuts compute the optimal classes assignation for the graph $H_c = (V_c, E_c)$, where $V_c = V_\alpha \cup V_\beta \cup \{s,t\}$ and $E_c = \{e_{ij} \in E \mid (i = \alpha) \cap (j = \beta)\} \cup \{e_{s \to k}, e_{k \to t}, \forall k \in (V_\alpha \cup V_\beta)\}$. In this case, a node connected to the source s in the minimum cut is classified as belonging to the class α, and to β otherwise. A change in the assignation of a node in the minimum cut with respect to its previous one produces an α-β swap move. The algorithm ends when no swap moves increasing the likelihood can be performed.

α-expansions. This method iterates over the classes α in L performing α-expansions, i.e. changing the class assigned to a node from $L_{\bar{\alpha}} \in L - \alpha$ to the class α. Thus, for each class, graph cuts are used to compute the minimum cut of the graph $H_c = (V_c, E_c)$, where in this case $V_c = \{V_{\bar{\alpha}}, s, t\}$ is the set of nodes not assigned to the class α plus the source s and the sink t, and $E_c = \{e_{ij} \in E \mid (y_i \neq \alpha) \cap (y_j \neq \alpha)\} \cup \{e_{s \to k}, e_{k \to t}, \forall k \in V_{\bar{\alpha}}\}$. The nodes connected to the source s in the minimum cut produce an α-expansion, i.e. they replace their assigned class by α, while those linked to the sink t keep their initial class. This process is repeated until no α-expansion can increase the current expected likelihood.

Message passing methods.

The message passing approach, also called Belief Propagation (BP) or max-product [22], is based on the exchange of statistical information among related nodes. This is performed by passing messages from node y_i to node y_j, denoted as $m_{ij}(y_j)$, indicating the belief of node y_i about the belonging class of node y_j. These messages are computed in the following way:

$$m_{ij}^t = U(y_i, x_i, \omega) I(y_i, y_j, x_i, x_j, \theta) \prod_{y_k \in N_H(y_i) \setminus y_j} m_{ki}(y_i) \tag{D.9}$$

where $N_H(y_i) \setminus y_j$ is the set of neighbors of y_i in the graph H less y_j, and t is an iteration counter. Thus, the BP algorithm keeps sending messages between nodes following a certain message scheduling until the graph is calibrated, i.e. the messages exchanged between nodes are the same in two consecutive algorithm iterations. Once calibrated, the belief of each node is computed as:

$$b(y_i) = \kappa U(y_i, x_i, \omega) \prod_{y_j \in N_H(y_i)} m_{ji} \tag{D.10}$$

being κ a normalization component so the beliefs for node y_i sum to 1. Then, each node y_i is assigned to the class with the highest belief value in $b(y_i)$.

In the case of tree-structured graphs, such a message updating rule yields the optimal maximum. On the other hand, when it is applied to graphs with loops it adopts the name of *Loopy Belief Propagation* (LBP), and it is able to approximate a solution with a reasonable success. Next, we briefly describe the *Tree-Based Reparametrization* message passing algorithm (TRP) [21].

Tree-Based Reparametrization. This method pursuits a more global exchange of statistical information, not only between related nodes, aiming to reach a faster calibration even in cases where traditional BP methods fail. For that, a set of trees $T = \{T_1, .., T_t\}$ are spanned over the original graph $H = \{V, E\}$ in such a way that every node in V belongs to (at least) one tree.

Once the set of trees T is obtained, the algorithm iteratively selects a tree and calibrates it, keeping fixed all the messages from the variables out of the tree. The

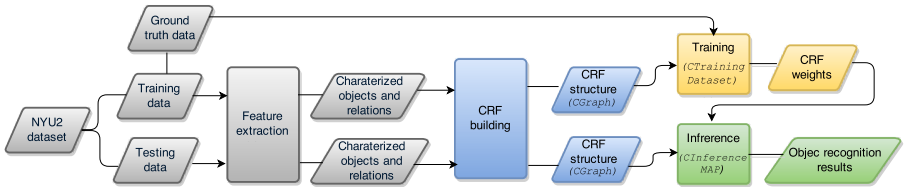


Figure D.3: Processing pipeline when training and exploiting a CRF model using the UPGMpp library. Colored shapes are processes and data handled by the library (see figure D.2), while gray shapes are problem specific and, therefore, defined by the user.

calibration of a tree can lead to the *miscalibration* of other trees, so the algorithm has to be repeated until global calibration is reached. Once calibrated, the inference results are obtained in the same way as the original LBP algorithm. The interested reader can refer to [21] for more detail.

4 Contextual object recognition using the UPGMpp library

This section shows the flexibility and usability of the UPGMpp library when applied to the scene object recognition problem exploiting contextual information. First, we introduce the NYU2 [19] dataset which has been used to test the library (section 4.1). Then, we describe the processes needed for training and testing (performing inference) Conditional Random Fields (CRFs) employing the UPGMpp library (section 4.2). Finally, the recognition results yielded by the different inference methods within UPGMpp are shown (section 4.3), including performance information, which support the suitability of the application of the presented library for the scene object recognition problem.

4.1 The NYU2 dataset

In this work we employ RGB-D images from the NYU2 dataset, which contains a total of 1,449 densely labeled pairs of intensity and depth data. The dataset has been widely used in the literature (e.g. [8]) due to its challenging cluttered scenes from commercial and residential buildings. In this work, we have used 208 scenes belonging to rooms typically found in houses, namely: bedrooms, bathrooms, kitchens and living rooms, containing 1,692 objects that belong to 22 objects classes, e.g. cabinet, counter, bottle, toilet, sofa, lamp, clothes, etc.

4.2 CRFs creation and operation with UPGMpp

Figure D.3 shows the general processing pipeline when training a CRF model and exploiting it to perform object recognition. The colored shapes in the image comprise the processes and data provided/managed by the UPGMpp library, and the involved package (as before, blue represents the *base* package, yellow the *training* one, and green the *inference* package). On the other hand, the gray shapes are problem-specific, so they have to be defined by the user. We have instantiated this pipeline for the case of the NYU2 dataset, although it can be replaced by any other.

The NYU2 dataset has been split into training and testing scenes which have to be processed in order to extract the features of the objects appearing in them and their relationships. These features are defined by the user, and in this work we have used the following object/node ones: orientation, planarity, linearity, minimum, maximum and centroid heights from the floor, volume, area of its biggest face, and hue variation, while the chosen contextual/edge features have been: difference of orientation, vertical distance, *is on* relation[2], and a bias value that states the compatibility of the related object classes. The extracted features from the training scenes are used to build their respective CRF representations (instances of `CGraph`), which together with ground truth information are inserted into an instance of the `CTrainingDataset`. Then, the selected training method computes the vectors of weights of the CRF model. In UPGMpp these weights are stored within the node and edge types (instances from `CNodeType` and `CEdgeType` respectively), so all the CRF graphs employing these node and edge types share the same vectors of weights.

On the other hand, for each scene into the testing data, its CRF structure is built according to the features shown by their constituent objects. Figure D.4 lines 1-22 shows a code snippet where two scene objects/nodes are created and characterized (concretely, x_6 and x_7 from figure D.1), as well as their contextual relation/edge, and then inserted into a CRF structure. Notice that both nodes share the same node type, `object`, and the used edge type is defined as `edgeBetweenObjects`. The same process is repeated for all the objects and relations appearing in the scene. Finally, the chosen inference process over the CRF structure gives the object recognition results, which are obtained for every scene within the testing data. As an illustrative example, figure D.4 lines 25-27 shows the definition of an ICM inference object, and its use to get the recognition results for a given CRF structure.

4.3 Contextual Object Recognition results

This section shows the results of applying the UPGMpp to the NYU2 dataset excerpt, as well as the computational time required for training and inference. This outcomes come from a 4-random-fold cross-validation, i.e. the 208 scenes were randomly divided into 4 folds with equal size, then three out of the four folds were used to train a CRF model, while the remaining fold was used to evaluate its performance. This process is repeated a total of 100 times, and the results are computed as the average

[2]This feature takes the value 1 if an object is placed on the other one, and 0 otherwise.

```
CGraph CRFgraph; // Conditional Random Field structure

Eigen::VectorXd node1Features(X); // Features of the object x6
node1Features << 4.11, 0.02, 0.22, 0.70, 0.89, 0.84, 0.17, 1.15, 78.59;

Eigen::VectorXd node2Features(X); // Features of the object x7
node2Features << 0.53, 0.02, 0.02, 0.83, 0.90, 0.87, 0.03, 0.40, 106.84;

CNodePtr y6 ( new CNode( object, node1Features ) ); // Create nodes
CNodePtr y7 ( new CNode( object, node2Features ) );

Eigen::VectorXd edgeFeatures(X); // Features of the contextual relation
edgeFeatures << 3.58, 0.13, 1, 1;

CEdgePtr edge_y6_y7 ( new CEdge( y6, y7, edgeBetweenObjects, edgeFeatures ) );

CRFgraph.addNode( y6 ); // Add nodes and edge to the graph
CRFgraph.addNode( y7 );
CRFgraph.addEdge( edge_y6_y7 );

// Keep on inserting nodes (objects) and edges (contextual relations)
// ...

// Perform ICM inference over the built CRF graph structure
CICMInferenceMAP ICM;
std::map<size_t,size_t> resultsMAP; // Map of results <ID_node,category>
ICM.infer( CRFGrpah, resultsMAP );  // Infer the recognition results
```

Figure D.4: A simple example of the use of the UPGMpp library.

of all the evaluations. The training of the CRF models has been done through the optimization of the pseudolikelihood function.

Table D.1 shows the results yielded by the different inference methods. Note that all the methods yielded the same outcome when considering *only* the features of the objects themselves. In this case, only nodes are added to the CRF graph structure, and all the methods chose the class assignation that maximizes the unary factor for each node (recall equation D.3). On the other hand, when contextual information is considered, the performance increases in all of the cases. It can be seen how the outcome of the Greedy method is slightly better than the ICM one, and similar to the α-expansions, being the α-β swaps the method with *worse* results. In contrast, LBP and TRP shows the better figures, improving the recognition results in more than a $\sim 5\%$ with respect to only using the objects' features (with no contextual information). Regarding the execution time consumed by these methods, the average ranges from the 0.46ms. of the ICM method up to the 37.39ms. of the α-β-swaps[3].

Despite of the results achieved by the inference methods in these tests, their general performance is affected by a number of factors, e.g. the features used to model the problem, the training method employed, or the domain at hand. Thus, for a differ-

[3]These figures were obtained using an Intel®Core™i5 3330 microprocessor at 3GHz and 8 GB DDR3 RAM memory at 1.6 GHz.

Table D.1: Scene object recognition results employing the different inference methods within the UPGMpp library with/without contextual information. It is also shown their mean execution time as well as the execution time in the worst cases (in ms.).

Method	ICM	Greedy	α-expansions	α-β swaps	LBP	TRP
Objects	65.71%	65.71%	65.71%	65.71%	65.71%	65.71%
Objects+context	68.37%	68.60%	68.99%	66.72%	**71.45%**	71.16%
Mean ex. time	**0.46**	2.92	7.78	37.39	2.16	11.05
Max ex. time	**4.85**	26.30	26.73	181.26	10.80	130.67

ent application, the performance of each individual method should be tested in order to employ the one giving the better results.

Regarding the time spent training the CRF models, its average over the 100 executions is 585.46 seconds. Notice that the training process has to be performed only once, and the resulting CRF model can then be used to recognize objects within any scene.

5 Conclusions and Future Work

This paper has presented the Undirected Probabilistic Graphical Models in C++ library (UPGMpp), a software library for dealing with the scene object recognition problem exploiting contextual information. A description of the main software packages of UPGMpp has been detailed, with especial emphasis on the implemented probabilistic inference algorithms, giving a practical idea about the library features and capabilities. This work also contributes with the application of the UPGMpp library to a use-case to both: train CRF models, and obtain object recognition results through the execution of a number of inference processes. The challenging NYU2 dataset is used to train and test the CRF models within the use-case, proving the virtues of the library, which is publicly available under a GNU General Public License at `http://mapir.isa.uma.es/work/upgmpp-library`.

Some additional features regarding the performance of the UPGMpp library are currently under work. For example, some parts could greatly reduce their execution time with the utilization of multi-core parallelization mechanisms, like OpenMP. Support for GPUs using CUDA and/or OpenCL could be also advantageous in that sense. We also plan to include visualization tools for PGM graphs, as well as sampling techniques to draw samples from the probability distribution defined by a PGM. We welcome any contribution to the UPGMpp library from the computer vision community.

Acknowledgements

This work has been funded by the Spanish grant program FPU-MICINN 2010 and the Spanish projects “TAROTH: New developments toward a robot at home” (Ref.

DPI2011-25483) and “PROMOVE: Advances in mobile robotics for promoting independent life of elders” (Ref. DPI2014-55826-R).

References

[1] Anand, A., Koppula, H.S., Joachims, T., Saxena, A.: Contextually guided semantic labeling and search for three-dimensional point clouds. Int. J. Rob. Res. 32(1), 19–34 (Jan 2013), http://dx.doi.org/10.1177/0278364912461538

[2] Besag, J.: On the statistical analysis of dirty pictures. Journal of the Royal Statistical Society. Series B (Methodological) 48(3), pp. 259–302

[3] Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. Pattern Analysis and Machine Intelligence, IEEE Transactions on 23(11), 1222–1239 (Nov 2001)

[4] D.M. Greig, B.P., Seheult, A.: Exact maximum a posteriori estimation for binary images. Journal of the Royal Statistical Society. Series B (Methodological) 51, pp. 271–279 (1989)

[5] Galleguillos, C., Belongie, S.: Context based object categorization: A critical survey. Comput. Vis. Image Underst. 114(6), 712–722 (Jun 2010)

[6] Guennebaud, G., Jacob, B., et al.: Eigen v3. http://eigen.tuxfamily.org (2010)

[7] Koller, D., Friedman, N.: Probabilistic Graphical Models: Principles and Techniques. MIT Press (2009)

[8] Lin, D., Fidler, S., Urtasun, R.: Holistic scene understanding for 3d object detection with rgbd cameras. Computer Vision, IEEE International Conference on 0, 1417–1424 (2013)

[9] Murphy, K.P., Weiss, Y., Jordan, M.I.: Loopy belief propagation for approximate inference: An empirical study. In: Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence. pp. 467–475. UAI’99, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1999)

[10] N. Okazaki, J.N.: libLBFGS: a library of Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS). http://www.chokkan.org/software/liblbfgs/ (2015), [Online; accessed 20-April-2015]

[11] Okazaki, N.: Crfsuite: a fast implementation of conditional random fields (crfs), http://www.chokkan.org/software/crfsuite/, [Online; accessed 28-April-2015]

[12] Ren, X., Bo, L., Fox, D.: Rgb-(d) scene labeling: Features and algorithms. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. pp. 2759–2766 (June 2012)

[13] Ruiz-Sarmiento, J.R., Galindo, C., González-Jiménez, J.: Mobile robot object recognition through the synergy of probabilistic graphical models and semantic knowledge. In: European Conf. on Artificial Intelligence. Workshop on Cognitive Robotics (2014)

[14] Ruiz-Sarmiento, J.R., Galindo, C., González-Jiménez, J.: Scene object recognition for mobile robots through semantic knowledge and probabilistic graphical models. In: Expert Systems with Applications, 42(22):8805–8816 (2015).

[15] Ruiz-Sarmiento, J.R., Galindo, C., González-Jiménez, J.: OLT: A Toolkit for Object Labeling Applied to Robotic RGB-D Datasets. In: European Conference on Mobile Robots (2015).

[16] Ruiz-Sarmiento, J.R., Galindo, C., González-Jiménez, J.: Exploiting semantic knowledge for robot object recognition. In: Submitted for publication (2015)

[17] Schling, B.: The Boost C++ Libraries. XML Press (2011)

[18] Schmidt, M.: UGM: Matlab Code for Undirected Graphical Models. `http://www.cs.ubc.ca/~schmidtm/Software/UGM.html` (2015), [Online; accessed 28-April-2015]

[19] Silberman, N., Hoiem, D., Kohli, P., Fergus, R.: Indoor Segmentation and Support Inference from RGBD Images. In: Proc. of the 12th European Conf. on Computer Vision. pp. 746–760. ECCV'12, Springer-Verlag, Berlin, Heidelberg (2012)

[20] Valentin, J., Sengupta, S., Warrell, J., Shahrokni, A., Torr, P.: Mesh based semantic modelling for indoor and outdoor scenes. In: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on. pp. 2067–2074 (June 2013)

[21] Wainwright, M., Jaakkola, T., Willsky, A.: Tree-based reparameterization framework for analysis of sum-product and related algorithms. Information Theory, IEEE Transactions on 49(5), 1120–1146 (May 2003)

[22] Weiss, Y., Freeman, W.T.: On the optimality of solutions of the max-product belief-propagation algorithm in arbitrary graphs. IEEE Trans. Inf. Theor. 47(2), 736–744 (Sep 2006)

[23] Xiong, X., Huber, D.: Using context to create semantic 3d models of indoor environments. In: in Proceedings of the British Machine Vision Conference (BMVC (2010)

E

OLT: A Toolkit for Object Labeling Applied to Robotic RGB-D Datasets

Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, Javier Gonzalez-Jimenez

Published in European Conference on Mobile Robots (ECMR), 2015.

OLT: A Toolkit for Object Labeling Applied to Robotic RGB-D Datasets

J.R. Ruiz-Sarmiento, C. Galindo and J. Gonzalez-Jimenez

Machine Perception and Intelligent Robotics Group, System Engineering and Auto. Dept., University of Málaga, Campus de Teatinos, 29071, Málaga, Spain.

In this work we present the *Object Labeling Toolkit* (OLT), a set of software components publicly available for helping in the management and labeling of sequential RGB-D observations collected by a mobile robot. Such a robot can be equipped with an arbitrary number of RGB-D devices, possibly integrating other sensors (e.g. odometry, 2D laser scanners, etc.). OLT first merges the robot observations to generate a 3D reconstruction of the scene from which object segmentation and labeling is conveniently accomplished. The annotated labels are automatically propagated by the toolkit to each RGB-D observation in the collected sequence, providing a dense labeling of both intensity and depth images. The resulting objects' labels can be exploited for many robotic oriented applications, including high-level decision making, semantic mapping, or contextual object recognition. Software components within OLT are highly customizable and expandable, facilitating the integration of already-developed algorithms. To illustrate the toolkit suitability, we describe its application to robotic RGB-D sequences taken in a home environment.

1 Introduction

A comprehensive dataset supposes a valuable benchmark tool for tuning, testing, and comparing robotic algorithms and systems in a convenient and fair way. Public datasets consisting of intensity images [1, 2, 3] have largely helped researchers to push ahead the state-of-the-art in object recognition or scene interpretation. Nowadays, given the increasing number of capabilities and applications that are demanded to a mobile robot, e.g. semantic mapping [4], high-level decision making [5], or contextual object recognition [6, 7, 8, 9], new particularly oriented datasets are required.

RGB-D cameras have become a key source of information for such *robotic* datasets. Although the sensory data of these datasets may be conveniently gathered by the mobile robot itself, human supervision is still needed to segment objects and to label them, i.e. to add annotations over portions of the observed data as belonging to a certain object class, e.g. floor, table, lamp, etc. This is the motivation for the software toolkit that we have developed and is described in this paper.

More specifically, we present the *Object Labeling Toolkit* (OLT) to provide the robotic community with a tool to efficiently label datasets compound of sequences of RGB-D observations, gathered from an arbitrary number of RGB-D sensors. For that, the toolkit builds a 3D reconstruction of each RGB-D sequence within a given dataset, and allows the user to graphically label objects within that reconstruction (see

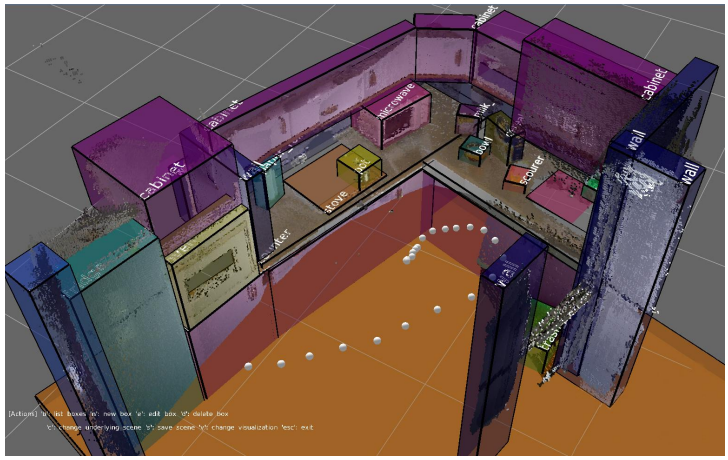


Figure E.1: Example of a kitchen reconstructed from a sequence of RGB-D observations within a robotic dataset. The appearing objects have been labeled (colored boxes). Gray spheres stand for RGB-D sensor poses.

Fig. E.1). This ground truth annotations are automatically propagated to all the RGB-D observations without requiring human supervision, resulting in a dense labeling of both intensity and depth data.

OLT comprises a number of software components covering the following functionality: i) dataset pre-processing, ii) localization of RGB-D observation poses, iii) 3D scene reconstruction, iv) labeling of the reconstructed scene, and v) automatic propagation of annotated labels (see Fig. E.2). Some of these functionalities can exploit additional information coming from sensors usually present in a robotic platform, e.g. the robot pose estimation computed from 2D laser scans. All the components are highly customizable in order to fit the particularities of robotic datasets, and can be easily expandable to integrate other algorithms of interest. OLT is publicly available under a GNU General Public License at (http://mapir.isa.uma.es/work/object-labeling-toolkit), and it resorts to the Mobile Robot Programing Toolkit (MRPT [10]) and the Point Cloud Library (PCL [11]) for point cloud registration and smoothing algorithms, and for data representation and visualization purposes. Aiming to illustrate the toolkit suitability, we show how it is employed for segmenting and labeling a robotic dataset from a home environment, and also describe its impact on the required processing time w.r.t. a typical manual solution.

2 Related work

In general, RGB-D datasets providing labeled objects information can be grouped into: *object-centric*, *single-view*, and *sequential-view* datasets. *Object-centric* datasets

[12, 13, 14, 15] provide labeled RGB-D observations of isolated objects, a poor source of information for many robotic applications like scene understanding or contextual object recognition. On the other hand, *single-view* datasets [16, 17, 18, 19, 20], are compounded of labeled RGB-D observations of particular scenarios (e.g. a room or an office). This information is richer from the point of view of those applications, but data of the whole robot environment is not available. Finally, *sequential-view* datasets [21, 22] provide a sequence of labeled observations covering the whole inspected workspace, which is the best suitable option for testing trending robotic algorithms or systems. Unfortunately, their number is quite limited mainly due to the arduous labor that entails the data processing.

RGB-D datasets carry out the tedious object labeling task in different ways. Some works resort to *Amazon Mechanical Turk* (AMT) to label their intensity images [16, 18, 19], usually through a labeling tool like LabelMe [2], but this merely divides the workload, and the annotated information still needs to be thoroughly checked to fix incoherent labels. Another approach is the manual labeling of *key intensity frames* from a sequence, propagating these labels to the remaining RGB-D observations [21, 22], but this is only suitable for sequences with simple sensor trajectories, and additionally shows the same limitations as the AMT option. There are also works that reconstruct a 3D representation of the inspected scene and annotate the objects appearing on it [17], but there is not a *labeling feedback* to the RGB-D observations' sequence(s). Similar works to our approach are [12] and [15], where the ground truth annotations over a reconstructed scene are also propagated to the individual RGB-D observations employing an ad-hoc software which, to the best of our knowledge, is not publicly available. We contribute in this paper with an open source solution conveniently divided into configurable components, which provides the robotic community with a number of functionalities towards an efficient labeling of arbitrarily large collections of RGB-D data.

3 Dataset management: OLT toolkit

The *Object Labeling Toolkit* (OLT) is a set of software components aimed to facilitate the management and processing of robotic *sequential-view* datasets. Concretely, it provides robotic researchers with the needed tools for achieving a dense labeling of the objects appearing in each RGB-D observation within a dataset sequence, aiming to drastically reduce the user participation in the process. It has been designed to be flexible: it handles datasets containing an arbitrary number of sensors providing RGB-D and (optionally) 2D scans information, and its components can be used independently according to the user needs, or even occasionally expanded with the integration of additional algorithms providing the same functionality.

Figure E.2 shows an overview of the software components within the toolkit and their interrelations. In a nutshell, the labeling process of RGB-D data within a certain dataset sequence starts with a pre-processing step, which sets the extrinsic and intrinsic parameters of the sensors employed during its gathering (Sec. 3.1). Then, the sensor poses in a global frame from where each RGB-D observation was taken are

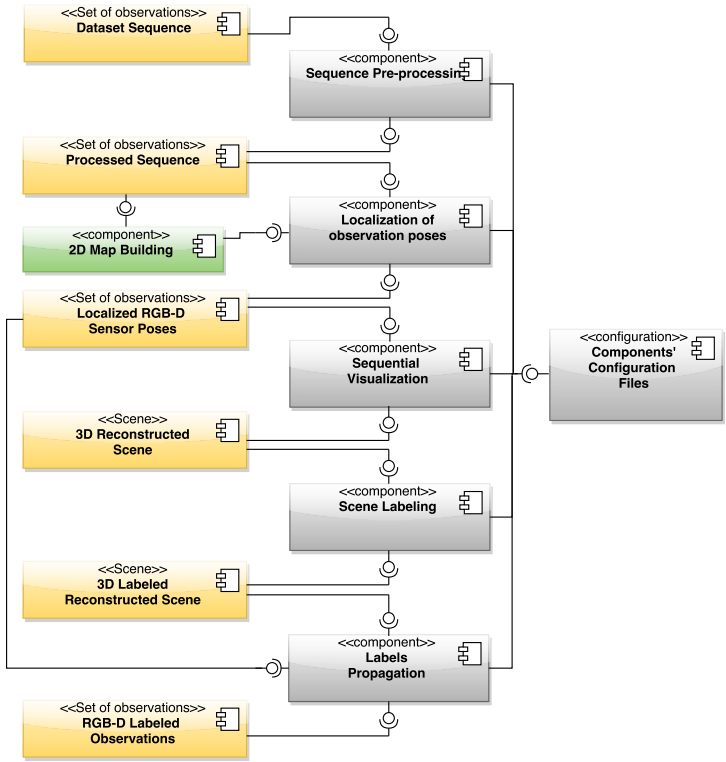


Figure E.2: Components diagram showing their collaboration within the OLT Toolkit. Gray boxes represent components provided by the toolbox, the green box is a component supplied by the MRPT library, and yellow boxes stand for persistent data. The reader is referred to the *online* version of this work for references to colors.

computed through the alignment of their depth information (Sec. 3.3). This component can optionally employ a geometric map built upon 2D laser observations from the same dataset sequence (Sec. 3.2). This permits the component to perform a rough robot localization within the explored area, hence giving a useful initial guess for such RGB-D sensor poses' computation. The resultant information is used to three-dimensionally reconstruct the scene, and the goodness of such a reconstruction can be visually inspected (Sec. 3.4). The reconstructed scene is then manually labeled by an human operator (Sec. 3.5), i.e. the objects appearing in the scene are annotated with their belonging classes, e.g. floor, table, book, etc. Finally, those annotated labels are propagated to subsequent RGB-D observations (both intensity and depth images) of the dataset making use of the computed sensor poses (Sec. 3.6). This labeling process can be repeated for an arbitrary number of RGB-D sequences within a given dataset.

It is worth to mention that each toolkit component resorts to a configuration file to easily fit their behavior to the requirements of a given dataset.

The toolkit components are built upon two widely used libraries: the *Mobile Robot Programming Toolkit* (MRPT [10]), and the *Point Cloud Library* (PCL [11]). We resort to MRPT to manage datasets into the *Rawlog common robotics dataset format*, which are capable of handling any variety of robotic sensor with precise timestamping[1]. This library also provides efficient visualization tools and implementations of point cloud registration algorithms. On the other hand, we rely on PCL to incorporate point cloud smoothing and registering techniques.

3.1 Dataset pre-processing

The first toolkit component sets the extrinsic and intrinsic parameters of the sensors used to gather the dataset sequence being processed. The extrinsic parameters refer to the position of the sensors with respect to the robot centroid, and can be retrieved in different ways [23, 24]. The intrinsic parameters describe geometric and distortion properties of the sensors. RGB-D devices show a different set of intrinsic parameters for their intensity and depth cameras, including: focal length, principal point coordinates, and radial and tangential distortions. Also needed are extrinsic parameters to relate the position of both cameras. The intrinsic parameters largely differ among RGB-D devices, so it is recommended to calibrate them through algorithms like [25]. Those extrinsic and intrinsic parameters can be conveniently introduced into a configuration file, and this component will set them throughout all the contained observations within the dataset sequence.

This pre-processing step permits the user to effortlessly change the sensor(s) calibration parameters within arbitrarily large dataset sequences, enabling in this way the comparison of the results yielded by the following toolkit components when employing different calibration techniques/parameters.

3.2 2D map building

The utilization of this component is optional, but it has shown to improve the results obtained during the computation of the RGB-D sensor poses (Sec. 3.3). To employ it, the dataset sequence must provide 2D laser observations from, at least, one laser range scanner. These observations are then processed by an ICP-based (Iterative Closest Point) technique [26] within the *icp-slam* MRPT application in order to generate a geometric map. Figure E.3 shows an example of a map from a bedroom built in our experiments.

[1]There exist a number of tools to convert datasets captured by other popular middlewares to *rawlogs*, e.g. ROS (http://wiki.ros.org/mrpt_rawlog).

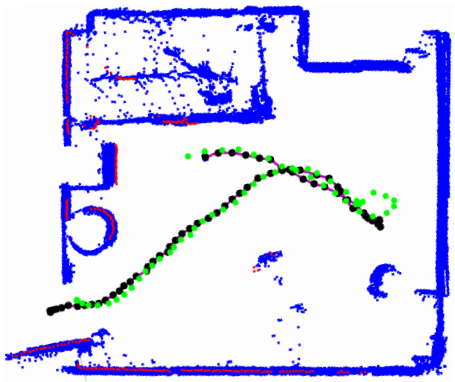


Figure E.3: In blue, geometric map of a bedroom built employing the 2D laser scans within a sequence. In red, example of 2D laser scan aligned within the geometric map. Black circles represent the robot localizations at the time instants when the 2D laser scans where gathered, and green ones the computed poses of the RGB-D sensor. The dataset sequence started at the room's door.

3.3 Observation poses

This component aims to find the sensor poses from where each RGB-D observation was taken within a 6D global frame (3D position: x,y, and z, plus three attitude angles: yaw, pitch, and roll). This sensors' localization can be performed following any of these two approaches:

i) For each RGB-D observation o_i gathered by a sensor d, it is carried out an alignment process with the observation o_{i-1} previously taken by the same device. For that, it is employed a registration algorithm that exploits their depth information in the form of point clouds. This registration yields the rigid transformation $T_{o_i,o_{i-1}}$ between the two sensor poses, from which we can compute the sensor location where the observation o_i was taken:

$$S_{o_i} = T_{o_i,o_{i-1}} \oplus L_d \tag{E.1}$$

where L_d stands for the pose of the sensor d on the robot frame (i.e. its extrinsic parameters). The first observation o_1 from such a sensor is considered to be taken with the robot in its initial position, i.e. at the origin of the global frame.

ii) The second approach employs the 2D geometric map from the previous component and the 2D laser observations to localize the robot within the global frame by means of ICP. Then, the sensor poses for each RGB-D observation o_i are computed through the interpolation of those robot localizations employing their timestamps:

Figure E.4: Left, point clouds representing a bed-set and a pair of shoes reconstructed employing the sensor poses yielded by the robot localization. Right, the same objects reconstructed with the sensor poses refined with GICP.

$$R_{o_i} = R_1 \oplus ((R_2 \ominus R_1) \cdot t_{elapsed}) \tag{E.2}$$

$$S_{o_i} = R_{o_i} \oplus L_d \tag{E.3}$$

where R_1 and R_2 are the robot locations with timestamps just before $(t-1)$ and after $(t+1)$ the o_i one (t), and $t_{elapsed} = (t-(t-1))/((t+1)-(t-1))$ is a scalar value. In this case, the global coordinate frame is specified by the geometric map. Optionally, these locations can be refined through the approach described in i) in a post-processing step (see Fig. E.4). Fig. E.3 shows an example of robot locations and RGB-D sensor poses from a *bedroom sequence*.

The toolkit user can choose between two different point clouds registration algorithms: the ICP-3D method within the MRPT library, and the implementation of the Generalized-ICP algorithm [27] from PCL. In addition to the sensor localization and point clouds registration algorithms to be used, a number of options can be selected from the component's configuration file:

- *Point clouds smoothing*. Depth observations from a RGB-D device are prone to provide noisy measurements over surfaces, effect that notoriously increases with distance. This option permits to apply a smoothing method to such depth information before operating with them. Concretely, we rely on the implementation of the *Fast Bilateral Filter* algorithm [28] within PCL.

- *Memory utilization*. This option enables an incremental registration through the use of a *memory of observations*, i.e. when a RGB-D observation taken at time t from a certain sensor is being registered, all the previous registered observations from all RGB-D sensors are considered. This increases the quality of the alignment results at the expense of a higher computational/time cost.

- *Key poses*. When enabled, only observations taken from *considerably* different robot poses are processed. This is useful in cases where the robot speed during the dataset gathering was too slow, so quite similar observations are collected. In the current implementation two poses are considered different according to two user defined parameters: minimum euclidean distance, and minimum rotation angle difference. This option relies on the robot locations yielded by the second localization approach.

The output of this component is a dataset sequence with the poses of the RGB-D observations set according to the their yielded localization into the global frame.

3.4 Sequential visualization

The goal of the *sequential visualization* component is twofold: first, it permits the user to visually inspect the results of the RGB-D sensor poses localization, and second, it creates a 3D reconstruction of the scene. Concretely, the colored point clouds from the RGB-D observations are projected from its local sensor frame to the global one. For that, given an observation o_i and its sensor pose S_{o_i}, each point P_j in its point cloud is projected as follows:

$$P_{j,G} = S_{o_i} \oplus P_{j,L} \tag{E.4}$$

being $P_{j,L}$ the point 3D coordinates in the sensor local frame. Once the point clouds have been projected, they are sequentially prompted to the user employing visualization tools from MRPT, which in turn resorts to *octrees* and *OpenGL*. The user can opt for a step by step visualization that adds a new registered point cloud when any *key* is pushed, if s/he needs to inspect the scene reconstruction in detail. Once the reconstruction has been shown, it is created a *scene* file containing the resultant colored point cloud map of the whole scene (see first column in Fig. E.6).

3.5 Label reconstructed scene

The labeling of the reconstructed scene is performed by manually fitting *boxes* to the objects appearing in it. We have chosen boxes as the geometric primitives given their easy operation and intuitive fit to objects showing different shapes. Thus, for each object to be labeled in the scene, the user creates and edits a box B_i by setting its position, scale and rotation so the object is fully contained in it. When such an editing is completed, each box can be annotated with its ground truth class, e.g. table, chair, wall, book, etc, conforming a *box-label* pair $(B_i, label_i)$. It is also possible to label

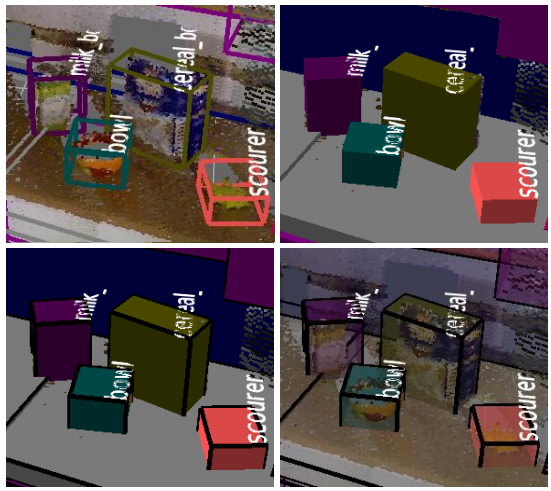


Figure E.5: Example of the four box visualization modes during the labeling of the reconstructed scene.

the scene for *instance object recognition* purposes, i.e. algorithms trying to recognize particular instances of objects instead of their general category, by adding an identifier to these annotations, e.g. *book_1* or *bed_red*. Complex objects can be labeled employing multiple boxes. The box editing operations, as well as the functionality described below, can be conveniently performed by means of keyboard shortcuts.

In order to facilitate the labeling process, the user has available a number of options: check at any moment a list of the already inserted boxes, add an arbitrary number of boxes, and edit/remove an existing box. Additionally, there are four different box visualization modes: *wireframe*, *solid*, *solid with borders*, and *transparent solid with borders*, which have resulted extremely useful during our tests to visually check the inner points for each box (see Fig. E.5). When the labeling is finished, the work done can be saved to a *scene* file containing the initial reconstructed scene and the set $B = ((B_0, label_0), ..., (B_N, label_N))$ of inserted boxes along with their labels, being N the number of objects appearing in the scene (see second column in Fig. E.6).

3.6 Labels propagation

The last component in the toolkit is in charge of propagating the labels into the reconstructed scene to each RGB-D observation within the dataset sequence. For that, given an observation o_i, for each point P_j in its point cloud representation it is checked in which boxes $B_0, ..., B_N$ the point lies inside. It is recalled that the point cloud of both the observation and the labeled scene are in the same coordinate frame thanks to the previous sensor pose localizations, so no additional transformations are needed.

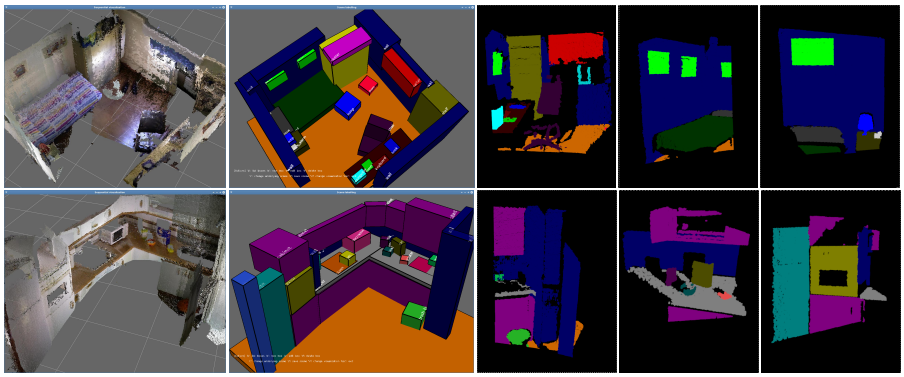


Figure E.6: First column, reconstructed scenes from the sequences within the dataset. Second column, labeled reconstructed scenes. Third-fifth columns, examples of individual point clouds from RGB-D observations labeled by the propagation of the annotations within the reconstructed scenes.

Then, if a certain point P_j lies inside a box B_k, for what we employ ray casting with the boxes' boundaries, it is annotated with the box associated label ($label_k$). Since the correspondence between such a point and the (x, y) coordinates of its associated pixel into the intensity and depth images from o_i are known, such images are also annotated. Thus, they are labeled at once the point cloud, the intensity image, and the depth image. The repetition of this process for each RGB-D observation within the dataset sequence completes the labeling pipeline.

4 Toolkit usage

This section aims to illustrate the usability of OLT, showing its virtues for an effortless labeling of RGB-D sequences. For that we have collected a home environment dataset employing a Giraff commercial robot [29] enhanced with an RGB-D device (Asus XTion Pro Live [30]), and a 2D laser scanner (Hokuyo model URG-04LX-UG01 [31]). The robot was teleoperated in two different sessions, fully inspecting a kitchen in the first session, and a bedroom in the second one. Each session produced a sequence of observations compound of data from the two added sensors, summing up a total of 77 RGB-D observations and 142 laser scans.

According to the functionality provided by the toolkit, the dataset sequences were preprocessed to set the sensors' calibration parameters (recall Sec. 3.1), and the 2D laser scans were used to build a geometric map for both, the kitchen and the bedroom (see Sec. 3.2). These maps were used to localize within them the sensor poses from where the RGB-D observations were taken. As it was explained in Sec. 3.3, those geometric maps are not an indispensable requirement for such a localization, but they have shown to provide useful cues for improving the registration of RGB-D

observations. Once localized, the RGB-D observations are registered, forming a 3D reconstruction of both scenes as it is shown in the first column of Fig. E.6 (recall Sec. 3.4). These reconstructions are then manually labeled by a human operator that disposes of an intuitive list of options to fit boxes to the scene objects, and annotates them with their respective belonging classes. Notice that this is the unique point in the toolkit where human intervention is needed. They were labeled in total 59 objects, belonging to 39 different classes. The second column in Fig. E.6 shows both labeled scenes. Finally, the annotated information is automatically propagated to all the RGB-D observations within the kitchen and bedroom sequences, resulting in an efficient labeling of their intensity and depth images[2]. Fig. E.6 shows a number of labeled point clouds.

Regarding the time spent in labeling, the human operator needed 2 hours to annotate both the kitchen and the bedroom scenes, spending on average 2 minutes per object. To compare this with the labeling of all the RGB-D observations individually, we followed the typical intensity image labeling approach and annotated 5 non-consecutive observations from each sequence, extrapolating the results to the whole dataset. This yields a total of $\sim$3 hours needed for the labeling of the kitchen sequence, and $\sim$7 hours for the bedroom, which clearly illustrates the benefits of the toolkit utilization. When following such a typical approach we found problems to accurately label the objects' boundaries, and with objects partially occluded and with an unclear belonging class, drawbacks that are mitigated with the utilization of the proposed toolkit.

It is worth to mention an advantage of the utilization of a geometric map to localize sensor poses when sequences to be labeled are gathered from the same places captured at different times. In this case, the labeling performed for a sequence can be loaded into the reconstructed scene of other sequence, so only the boxes associated to moved/appearing/disappearing objects have to be modified/added, resulting in an additional time saving.

5 Conclusion and future work

In this work we have presented the *Object Labeling Toolkit* (OLT), a publicly available software solution for the management of arbitrary large robotic datasets (`http://mapir.isa.uma.es/work/object-labeling-toolkit`). The major goal of OLT is to provide the robotic community with a tool to efficiently label objects appearing in a sequence of RGB-D observations. It has been also presented the flexible, highly customizable software components aiming to fit the needs of particular robotic datasets. The toolkit can handle different platform setups, i.e. datasets gathered by an arbitrary number of RGB-D sensors, and even can profit from 2D laser scanners, devices that are usually present in a mobile robot. We have illustrated how OLT is

[2]Recall that each point in the point cloud is associated with a pixel from the depth image, and given that this image and the intensity one are registered, the labeling of both images from the point cloud is straightforward.

applied to the labeling of a home environment dataset, and show that it considerably decreases the time needed by an human to complete such a task.

The toolkit is in constant development with the inclusion of new features and functionalities. For example, we are studying the incorporation of algorithms for a globally consistent alignment of the RGB-D observations used to reconstruct a scene. We also plan to integrate, in addition to boxes, different geometric primitives to be used during the labeling of the reconstructed scenes, e.g. spheres. OLT welcomes any contribution from the robotics community.

Acknowledgment

This work has been supported by the Spanish grant program *FPU-MICINN 2010* and the Spanish projects *TAROTH: New developments toward a Robot at Home* (DPI2011-25483) and *PROMOVE: Advances in mobile robotics for promoting independent life of elders* (DPI2014-55826-R).

References

[1] M. Everingham, L. van Gool, C. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.

[2] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, "Labelme: A database and web-based tool for image annotation," *Int. J. Comput. Vision*, vol. 77, no. 1-3, pp. 157–173, May 2008.

[3] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge," *CoRR*, vol. abs/1409.0575, 2014.

[4] A. Pronobis, P. Jensfelt, K. Sjöö, H. Zender, G.-J. M. Kruijff, O. M. Mozos, and W. Burgard, "Semantic modelling of space," in *Cognitive Systems*, ser. Cognitive Systems Monographs, H. I. Christensen, G.-J. M. Kruijff, and J. L. Wyatt, Eds., 2010, vol. 8, pp. 165–221.

[5] C. Galindo and A. Saffiotti, "Inferring robot goals from violations of semantic knowledge," *Robotics and Autonomous Systems*, vol. 61, no. 10, pp. 1131–1143, 2013.

[6] Ruiz-Sarmiento, J.R., C. Galindo, and J. González-Jiménez, "Exploiting semantic knowledge for robot object recognition," in *Knowledge-Based Systems*, 2015.

[7] Ruiz-Sarmiento, J. R., C. Galindo, and J. González-Jiménez, "Mobile robot object recognition through the synergy of probabilistic graphical models and semantic knowledge," in *European Conf. on Artificial Intelligence. Workshop on Cognitive Robotics*, 2014.

[8] Ruiz-Sarmiento, J.R., C. Galindo, and J. González-Jiménez, "UPGMpp: a Software Library for Contextual Object Recognition," in *3rd. Workshop on Recog. and Action for Scene Understanding*, 2015.

[9] Ruiz-Sarmiento, J. R., C. Galindo, and J. González-Jiménez, "Scene Object Recognition for Mobile Robots through Semantic Knowledge and Probabilistic Graphical Models," 2015, submitted.

[10] J.L. Blanco Claraco, "Mobile Robot Programming Toolkit (MRPT)," `http://www.mrpt.org`, 2015, [Online; accessed 28-April-2015].

[11] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, May 9-13 2011.

[12] 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*, May 2011, pp. 1817–1824.

[13] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab, "Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes," in *Proceedings of the 11th Asian Conference on Computer Vision - Volume Part I*, ser. ACCV'12. Berlin, Heidelberg: Springer-Verlag, 2013, pp. 548–562.

[14] A. Singh, J. Sha, K. Narayan, T. Achim, and P. Abbeel, "Bigbird: A large-scale 3d database of object instances," in *IEEE International Conference on Robotics and Automation*, May 2014.

[15] K. Lai, L. Bo, and D. Fox, "Unsupervised feature learning for 3d scene labeling," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, May 2014, pp. 3050–3057.

[16] A. Janoch, S. Karayev, Y. Jia, J. T. Barron, M. Fritz, K. Saenko, and T. Darrell, "A category-level 3-d object dataset: Putting the kinect to work," in *1st Workshop on Consumer Depth Cameras for Computer Vision (ICCV workshop)*, November 2011.

[17] A. Anand, H. S. Koppula, T. Joachims, and A. Saxena, "Contextually guided semantic labeling and search for three-dimensional point clouds," *In The International Journal of Robotics Research*, vol. 32, no. 1, pp. 19–34, Jan. 2013.

[18] N. Silberman and R. Fergus, "Indoor scene segmentation using a structured light sensor," in *Proceedings of the International Conf. on Computer Vision - Workshop on 3D Representation and Recog.*, 2011.

[19] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor Segmentation and Support Inference from RGBD Images," in *Proc. of the 12th European Conference on Computer Vision (ECCV 2012)*, 2012.

[20] A. Aldoma, T. Faulhammer, and M. Vincze, "Automation of "ground truth" annotation for multi-view rgb-d object instance recognition datasets," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, Sept 2014, pp. 5016–5023.

[21] D. Meger and J. J. Little, "The UBC visual robot survey: A benchmark for robot category recognition," in *Experimental Robotics - The 13th International Symposium on Experimental Robotics, ISER 2012, June 18-21, 2012, Québec City, Canada*, 2012, pp. 979–991.

[22] J. Xiao, A. Owens, and A. Torralba, "Sun3d: A database of big spaces reconstructed using sfm and object labels," in *Computer Vision (ICCV), 2013 IEEE International Conference on*, Dec 2013, pp. 1625–1632.

[23] E. Fernandez-Moral, J. González-Jiménez, P. Rives, and V. Arévalo, "Extrinsic calibration of a set of range cameras in 5 seconds without pattern," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2014)*, Chicago, USA, September 2014.

[24] R. Gómez-Ojeda, J. Briales, E. Fernández-Moral, and J. González-Jiménez, "Extrinsic calibration of a 2d laser-rangefinder and a camera based on scene corners," in *IEEE International Conference on Robotics and Automation (ICRA)*, Seattle, USA, 2015.

[25] A. Teichman, S. Miller, and S. Thrun, "Unsupervised intrinsic calibration of depth sensors via slam," in *Proceedings of Robotics: Science and Systems*, Berlin, Germany, June 2013.

[26] P. J. Besl and N. D. McKay, "A method for registration of 3-d shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, Feb. 1992.

[27] A. Segal, D. Haehnel, and S. Thrun, "Generalized-icp," in *Proceedings of Robotics: Science and Systems*, Seattle, USA, June 2009.

[28] S. Paris and F. Durand, "A fast approximation of the bilateral filter using a signal processing approach," *Int. J. Comput. Vision*, vol. 81, no. 1, pp. 24–52, Jan. 2009.

[29] Giraff Technologies AB, "Giraff robot," `http://www.giraff.org/`, 2015, [Online; accessed 06-April-2015].

[30] ASUS, "Xtion PRO LIVE," `http://www.asus.com/Multimedia/Xtion_PRO_LIVE/`, 2015, [Online; accessed 06-April-2015].

[31] Hokuyo Automatic Co., "Hokuyo URG-04LX-UG01," `http://www.hokuyo-aut.jp`, 2015, [Online; accessed 06-April-2015].

Building Multiversal Semantic Maps for Mobile Robot Operation

Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, Javier Gonzalez-Jimenez

Submitted to Knowledge-Based Systems, 2016.

Building Multiversal Semantic Maps for Mobile Robot Operation

J.R. Ruiz-Sarmiento, C. Galindo and J. Gonzalez-Jimenez

Machine Perception and Intelligent Robotics Group, System Engineering and Auto. Dept., University of Málaga, Campus de Teatinos, 29071, Málaga, Spain.

Semantic maps augment metric-topological maps with meta-information, *i.e. semantic knowledge* aimed at the planning and execution of high-level robotic tasks. Semantic knowledge typically encodes human-like concepts, like types of objects and rooms, which are connected to sensory data when symbolic representations of percepts from the robot workspace are grounded to those concepts. This *symbol grounding* is usually carried out by algorithms that individually categorize each symbol and provide a crispy outcome – a symbol is either a member of a category or not. Such approach is valid for a variety of tasks, but it fails at: (i) dealing with the uncertainty inherent to the grounding process, and (ii) jointly exploiting the contextual relations among concepts (*e.g.* microwaves are usually in kitchens). This work provides a solution for *probabilistic symbol grounding* that overcomes these limitations. Concretely, we rely on Conditional Random Fields (CRFs) to model and exploit contextual relations, and to provide measurements about the uncertainty coming from the possible groundings in the form of beliefs (*e.g.* an object can be categorized (grounded) as a microwave or as a nightstand with beliefs 0.6 and 0.4, respectively). Our solution is integrated into a novel semantic map representation called *Multiversal Semantic Map* (*MvSmap*), which keeps the different groundings, or universes, as instances of ontologies annotated with the obtained beliefs for their posterior exploitation. The suitability of our proposal has been proven with the Robot@Home dataset, a repository that contains challenging multi-modal sensory information gathered by a mobile robot in home environments.

Keywords: mobile robots, symbol grounding, semantic maps, conditional random fields, ontologies, probabilistic inference

1 Introduction

A mobile robot intended to operate within human environments needs to create and maintain an internal representation of its workspace, commonly referred to as a *map*. Robotic systems rely on different types of maps depending on their goals. For example, *metric maps* are purely geometric representations that permit robot self-localization with respect to a given reference frame [1, 2]. *Topological maps* consider a graph structure to model areas of the environment and their connectivity, hence straightforwardly supporting navigational planning tasks [3, 4]. In its turn, *Hybrid maps* come

up from the combination of the previous ones by maintaining local metric information and a graph structure to perform basic but core robotic skills as localization and global navigation [5, 6]. A pivotal requirement for the successful building of these types of maps is to deal with uncertainty coming, among other sources, from errors in the robot perception (limited field of view and range of sensors, noisy measurements, etc.), and inaccurate models and algorithms. This issue is addressed in state-of-the-art approaches through probabilistic techniques [7].

Despite the possibilities of these representations, planning and executing high-level robotic tasks within human-like environments demand more sophisticated maps to enable robots, for example, to deal with user commands like *"hey robot! I am leaving, take care of the oven while I am out, please"* or *'Guide the customer through the aisle with garden stuff and show him the watering cans"*. Humans share a commonsense knowledge about concepts like *oven*, or *garden stuff*, which must be transferred to robots in order to successfully face those tasks. *Semantic maps* emerged to cope with this need, providing the robot with the capability to *understand*, not only the spatial aspects of human environments, but also the meaning of their elements (objects, rooms, etc.) and how humans interact with them (*e.g.* functionalities, events, or relations). This feature is distinctive and traversal to semantic maps, being the key difference with respect to maps that simply augment metric/topological models with labels to state the category of recognized objects or rooms [8, 9, 10, 11, 12]. Contrary, semantic maps handle meta-information that models the properties and relations of relevant concepts therein the domain at hand, codified into a *Knowledge Base* (KB), stating that, for example, microwaves are box-shaped objects usually found in kitchens and useful for heating food. Building and maintaining semantic maps involve the symbol grounding problem [13, 14, 15], *i.e.* linking portions of the sensory data gathered by the robot (percepts), represented by symbols, to concepts in the KB by means of some categorization and tracking method.

Semantic maps generally support the execution of reasoning engines, providing the robot with inference capabilities for efficient navigation, object search [16], human-robot interaction [17] or pro-activeness [18] among others. Typically, such engines are based on logical reasoners that work with crispy[1] information (*e.g.* a percept is identified as a microwave or not). The information encoded in the KB, along with that inferred by logical reasoners, is then available for a task planning algorithm dealing with this type of knowledge and orchestrating the aforementioned tasks [19]. Although crispy knowledge-based semantic maps can be suitable in some setups, especially in small and controlled scenarios [20], they are also affected by uncertainty coming from both, the robot perception, and the inaccurate modeling of the elements within the robot workspace. Moreover, these systems usually reckon on off-the-shelf categorization methods to individually ground percepts to particular concepts, which disregard the contextual relations between the workspace elements: a rich source of

[1]For the purpose of this work, the term *crispy* takes the same meaning as in classical logic: it refers to information or processes dealing with facts that either are true or not.

information intrinsic to human-made environments (for example that nigh-stands are usually in bedrooms and close to beds).

In this work we propose a solution for addressing the symbol grounding problem from a probabilistic stance, which permits both exploiting contextual relations and modeling the aforementioned uncertainties. For that we employ a Conditional Random Field (CRF), a particular type of Probabilistic Graphical Model [21], to represent the symbols of percepts gathered from the workspace as nodes in a graph, and their geometric relations as edges. This representation allows us to jointly model the symbol grounding problem, hence exploiting the relations among the elements in the environment. CRFs support the execution of probabilistic inference techniques, which provide the beliefs about the grounding of those elements to different concepts (*e.g.* an object can be a bowl or a cereal box with beliefs 0.8 and 0.2 respectively). In other words, the uncertainty coming both from the robot perception, and from the own symbol grounding process, is propagated to the grounding results in the form of beliefs.

The utilization of CRFs also leads to a number of valuable advantages:

- *Fast inference*: probabilistic reasoning algorithms, resorting to approximate techniques, exhibit an efficient execution that permits the retrieval of inference results in a short time [22, 23].
- *Multi-modal information*: CRFs easily integrate percepts coming from different types of sensors, *e.g.* RGB-D images and 2D laser scans, related to the same elements in the workspace [21].
- *Spatio-temporal coherence*: they can be dynamically modified to mirror new information gathered by the robot, also considering previously included percepts. This is done in combination with an anchoring process [14].
- *Life-long learning*: CRFs can be re-trained in order to take into account new concepts not considered during the initial training, but that could appear in the current robot workspace [24].

In order to accommodate the probabilistic outcome of the proposed grounding process, a novel semantic map representation, called *Multiversal Semantic Map* (*MvSmap*), is presented. This map extends the previous work by Galindo *et al.* [25], and considers the different combinations of possible groundings, or *universes*, as instances of ontologies [26] with belief annotations on their grounded concepts and relations. According to these beliefs, it is also encoded the probability of each ontology instance being the right one. Thus, *MvSmaps* can be exploited by logical reasoners performing over such ontologies, as well as by probabilistic reasoners working with the CRF representation. This ability to manage different semantic interpretations of the robot workspace, which can be leveraged by probabilistic conditional planners (*e.g.* those in [27] or [28]), is crucial for a coherent robot operation.

To study the suitability of our approach, we have conducted an experimental evaluation focusing on the construction of *MvSmaps* from facilities in the novel

Robot@Home dataset [29]. This repository consists of 81 sequences containing 87,000+ timestamped observations (RGB-D images and 2D laser scans), collected by a mobile robot in different ready to move apartments. Such dataset permits us to intensively analyze the semantic map building process, demonstrating the claimed representation virtues. As an advance on this study, a success of $\sim 81.5\%$ and $\sim 91.5\%$ is achieved while grounding percepts to object and room concepts, respectively.

The next section puts our work in the context of the related literature. Section 3 introduces the proposed Multiversal Semantic Map, while Section 4 describes the processes involved in the building of the map for a given environment, including the probabilistic symbol grounding. The suitability of our approach is demonstrated in Section 5, and Section 6 discuses some of its potential applications. Finally, Section 7 concludes the paper.

2 Related work

This section reviews the most relevant related works addressing the symbol grounding problem (Section 2.1), aiming to put into context our probabilistic solution, as well as the most popular approaches for semantic mapping that can be found in the literature (Section 2.2).

2.1 Symbol grounding

As commented before, the symbol grounding problem consists of linking symbols that are meaningless by themselves to concepts in a Knowledge Base (KB), hence retrieving a notion of their meanings and functionalities in a given domain [13]. In the semantic mapping problem, symbols are typically abstract representations of percepts from the robot workspace, namely objects and rooms [15, 30]. Therefore, a common approach to ground those symbols is their processing by means of categorization systems, whose outcomes are used to link them to concepts in the KB. The remaining of this section provides a brief overview of categorization approaches for both objects and rooms, and concludes with our proposal for a probabilistic grounding.

In its beginnings, the vast literature around object categorization focused on the classification of isolated objects employing their geometric/appearance features. A popular example of this is the work by Viola and Jones [31], where an integral image representation is used to encode the appearance of a certain object category, and is exploited by a cascade classifier over a sliding window to detect occurrences of such object type in intensity images. A limiting drawback of this categorization method is the lack of an uncertainty measurement about its outcome. Another well known approach, which is able to provide such uncertainty, is the utilization of image descriptors like Scale-Invariant Feature Transform (SIFT) [32] or Speeded-Up Robust Features (SURF) [33] to capture the appearance of objects, and its posterior exploitation by classifiers like Supported Vector Machines (SVMs) [34] or Bag-of-Words based ones [35, 36]. The work by Zhang *et al.* [37] provides a comprehensive review

of methods following this approach. It is also considerable the number of works tackling the room categorization problem through the exploitation of their geometry or appearance, like the one by Mozos *et al.* [38] which employs range data to classify spaces according to a set of geometric features. Also popular are works resorting to global descriptors of intensity images, like the *gist* of the scene proposed by Oliva and Torralba [39], those resorting to local descriptors like the aforementioned SIFT and SURF [40, 41], or the works combining both types of cues, global and local, pursuing a more robust performance [42, 43]. Despite the acceptable success of these *traditional* approaches, they can produce ambiguous results when dealing with objects/rooms showing similar features to two or more categories [44]. For example, these methods could have difficulties to categorize a white, box-shaped object as a microwave or a nightstand.

For that reason, modern categorization systems also integrate contextual information of objects/rooms, which has proven to be a rich source of information for the disambiguation of uncertain results [45, 46, 47]. Following the previous example, if the object is located in a bedroom and close to a bed, this information can be used to determine that it will likely be a nightstand. Probabilistic Graphical Models (PGMs) in general, and Undirected Graphical Models (UGMs) in particular, have became popular frameworks to model such relations and exploit them in combination with probabilistic inference methods [21]. Contextual relations can be of different nature, and can involve objects and/or rooms.

On the one hand, objects are not placed randomly, but following configurations that make sense from a human point of view, *e.g.* faucets are on sinks, mouses can be found close to keyboards, and cushions are often placed on couches or chairs. These object–object relations have been exploited, for example, by Anand *et al.* [48], which reckon on a model isomorphic to a Markov Random Field (MRF) to leverage them in home and office environments, or by Valentin *et al.* [49], which employ a Conditional Random Field (CRF), the discriminant variant of MRFs, to classify the faces of mesh-based representations of scenes compounded of objects according to their relations. Other examples of works also resorting to CRFs are the one by Xiong and Huver [50], which employs them to categorize the main components of facilities: clutters, walls, floors and ceilings, and those by Ruiz-Sarmiento *et al.* [22, 51, 52], where CRFs and ontologies [26] work together for achieving a more efficient and coherent object categorization.

On the other hand, object–room relations also supposes a useful source of information: objects are located in rooms according to their functionality, so the presence of an object of a certain type is a hint for the categorization of the room and, likewise, the category of a room is a good indicator of the object categories that can be found therein. Thus, recent works have explored the joint categorization of objects and rooms leveraging both, object–object and object–room contextual relations. CRFs have proven to be a suitable choice for modelling this holistic approach, as it has been shown in the works by Rogers and Christensen [53], Lin *et al.* [54], or Ruiz-Sarmiento *et al.* [55].

In this work we propose the utilization of a CRF to jointly categorize the percepts of objects and rooms gathered during the robot exploration of an environment, as well as its integration into a symbol grounding system. This CRF is exploited by a probabilistic inference method, namely Loopy Belief Propagation (LBP) [56, 57], in order to provide uncertainty measurements in the form of beliefs about the grounding of the symbols of these percepts to categories. Such categories correspond to concepts codified within an ontology, stating the typical properties of objects and rooms, and giving a semantic meaning to those symbols. Additionally, to make the symbols and their groundings consistent over time, we rely on an anchoring process [14]. To accommodate the outcome of this probabilistic symbol grounding, a novel semantic map representation is proposed.

2.2 Semantic maps

In the last decade, a number of works have appeared in the literature contributing different semantic map representations. One of the earliest works in this regard is the one by Galindo *et al.* [25], where a multi-hierarchical representation models, on the one hand, the concepts of the domain of discourse through an ontology, and on the other hand, the elements from the current workspace in the form of a spatial hierarchy that ranges from sensory data to abstract symbols. NeoClassic is the chosen system for knowledge representation and reasoning through Description Logics (DL), while the employed categorization system is limited to the classification of simple shape primitives, like boxes or cylinders, as furniture, *e.g.* a red box represents a couch. The potential of this representation was further explored in posterior works, *e.g.* for improving the capabilities and efficiency of task planners [19], or for the autonomous generation of robot goals [18]. A similar approach is proposed in Zender *et al.* [20], where the multi-hierarchical representation is replaced by a single hierarchy ranging from sensor-based maps to a conceptual abstraction, which is encoded in a Web Ontology Language (OWL)–DL ontology defining an office domain. To categorize objects, they rely on a SIFT-based approach, while rooms are grounded according to the objects detected therein. In Nüchter and Hertzberg [58] a constraint network implemented in Prolog is used to both codify the properties and relations among the different planar surfaces in a building (wall, floor, ceiling, and door) and classify them, while two different approaches are considered for object categorization: a SVM-based classifier relying on contour-based features, and a Viola and Jones cascade of classifiers reckoning on range and reflectance data.

These works set out a clear road for the utilization of ontologies to codify semantic knowledge [59], which has been further explored in more recent research. An example of this is the work by Tenorth *et al.* [60], which presents a system for the acquisition, representation, and use of semantic maps called KnowRob-Map, where Bayesian Logic Networks are used to predict the location of objects according to their usual relations. The system is implemented in SWI-Prolog, and the robot's knowledge is represented in an OWL-DL ontology. In this case, the categorization algorithm classifies planar surfaces in kitchen environments as tables, cupboards, drawers, ovens

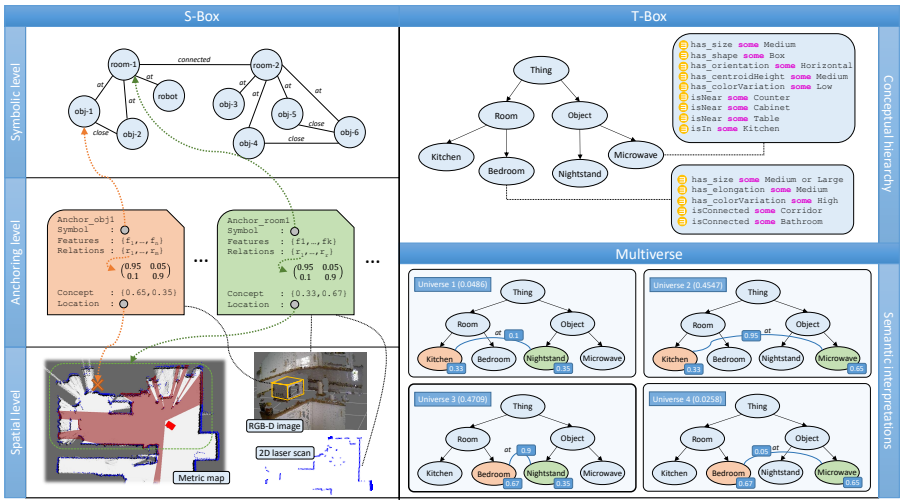


Figure F.1: Example of Multiversal Semantic Map representing a simple domestic environment.

or dishwashers [11]. The same map type and categorization method is employed in Pangercic *et al.* [61], where the authors focus on the codification of object features and functionalities relevant to the robot operation in such environments. The paper by Riazuelo *et al.* [62] describes the RoboEarth cloud semantic mapping which also uses an ontology for codifying concepts and relations, and rely on a Simultaneous Localization and Mapping (SLAM) algorithm for representing the scene geometry and object locations. The categorization method resorts to SURF features (like in Reinaldo *et al.*[63]), and performs by only considering the object types that are probable to appear in a given scene (the room type is known beforehand). In Günther *et al.* [64], the authors employ an OWL-DL ontology in combination with rules defined in the Semantic Web Rule Language (SWRL) to categorize planar surfaces.

It has been also explored the utilization of humans for assisting during the semantic map building process through a situated dialogue. Examples of works addressing this are those by Bastianelli *et al.* [65], Gemignani *et al.* [66], or the aforementioned one by Zender *et al.* [20]. The main motivation of these works is to avoid the utilization of categorization algorithms, given the numerous challenges that they must face. However, they themselves argue that the more critical improvement of their proposals would arise from a tighter interaction with cutting-edge categorization techniques. The interested reader can refer to the survey by Kostavelis and Gasteratos [67] for an additional, comprehensive review of semantic mapping approaches for robotic tasks.

The semantic mapping techniques discussed so far rely on crispy categorizations of the perceived spatial elements, *e.g.* an object is either a cereal box or not, a room is a kitchen or not, etc., which are typically exploited by (logical) reasoners and planners

for performing a variety of robotic tasks. As commented before, these approaches: (i) can lead to an incoherent robot operation due to ambiguous grounding results, and (ii) exhibit limitations to fully exploit the contextual relations among spatial elements. In this work we propose a solution for probabilistic symbol grounding to cope with both, the uncertainty inherent to the grounding process, and the contextual relations among spatial elements. Perhaps the closet work to ours is the one by Pronobis and Jensfelt [16], which employs a Chain Graph (a graphical model mixing directed and undirected relations) to model the grounding problem from a probabilistic stance, but that fails at fully exploiting contextual relations. We also present a novel representation called Multiversal Semantic Map (*MvSmap*), in order to accommodate and further exploit the outcome of the probabilistic symbol grounding.

3 The Multiversal Semantic Map

The proposed *Multiversal Semantic Map* (*MvSmap*) (see Figure F.1) is inspired by the popular, multi-hierarchical semantic map presented in Galindo *et al.* [25]. This map considers two separated but tightly related hierarchical representations containing: (i) the semantic, meta-information about the domain at hand, *e.g.* refrigerators keep food cold and are usually found in kitchens, and (ii) the factual, spatial knowledge acquired by the robot and its implemented algorithms from a certain workspace, *e.g.* `obj-1` is perceived and categorized as a refrigerator. These hierarchies are called terminological box (*T-Box*) and spatial box (*S-Box*), respectively, names borrowed from the common structure of hybrid knowledge representation systems [68].

MvSmaps enhance this representation by including uncertainty, in the form of *beliefs*, about the groundings (categorizations) of the spatial elements in the S-Box to concepts in the T-Box. For example, a perceived object, represented by the symbol `obj-1`, could be grounded by the robot as a microwave or a nightstand with beliefs 0.65 and 0.35, respectively, or it might think that a room (`room-1`) is a kitchen or a bedroom with beliefs 0.34 and 0.67. Moreover, in this representation the relations among the spatial elements play a pivotal role, and they have also associated compatibility values in the form of beliefs. To illustrate this, if `obj-1` was found in `room-1`, *MvSmaps* can state that the compatibility of `obj-1` and `room-1` being grounded to microwave and kitchen respectively is 0.95, while to microwave and bedroom is 0.05. These belief values are provided by the proposed probabilistic inference process (see Section 4.4).

Furthermore, *MvSmaps* assign a probability value to each possible set of groundings, creating a *multiverse*, *i.e.* a set of universes stating different explanations of the robot environment. A universe codifies the joint probability of the observed spatial elements being grounded to certain concepts, hence providing a global sense of certainty about the robot's understanding of the environment. Thus, following the previous example, a universe can represent that `obj-1` is a microwave and `room-1` is a kitchen, while a parallel universe states that `obj-1` is a nightstand and `room-1` is a bedroom, both explanations annotated with different probabilities. Thereby, the robot performance is not limited to the utilization of the most probable universe, like

traditional semantic maps do, but it can also consider other possible explanations with different semantic interpretations, resulting in a more coherent robot operation.

The next sections introduce the terminological box (Section 3.1), the spatial box (Section 3.2), and the multiverse (Section 3.3) in more detail, as well as the formal definition of *MvSmaps* (Section 3.4). In its turn, Section 4 describes how a *MvSmap* for a given robot workspace is built from scratch.

3.1 Representing semantic knowledge: the T-Box

The terminological box, or T-Box, represents the semantic knowledge of the domain where the robot is to operate, modeling relevant information about the type of elements that can be found there. Semantic knowledge has been traditionally codified as a hierarchy of concepts (*e.g.* `Microwave` *is-a* `Object` or `Kitchen` *is-a* `Room`), properties of that concepts (`Microwave` *hasShape* `Box`), and relations among them (`Microwave` *isIn* `Kitchen`). This hierarchy is often called *ontology* [26], and its structure is a direct consequence of its codification as a taxonomy. The T-Box gives meaning to the percepts in the S-Box through the grounding of their symbolic representations to particular concepts. For example, a segmented region of a RGB-D image, symbolized by `obj-1`, can be grounded to an instance of the concept `Microwave`.

The process of obtaining and codifying semantic knowledge can be tackled in different ways. For example, web mining knowledge acquisition systems can be used as mechanisms to obtain information about the domain of discourse [69]. Available common-sense Knowledge Bases, like ConceptNet [70] or Open Mind Indoor Common Sense [71], can be also analyzed to retrieve this information. Another valuable option is the utilization of internet search engines, like Google's image search [72], or image repositories like Flickr [73], for extracting knowledge from user-uploaded information. In this work we have codified the semantic knowledge through a human elicitation process, which supposes a truly and effortless encoding of a large number of concepts and relations between them. In contrast to online search or web mining-engine based methodologies, this source of semantic information (a person or a group of people) is trustworthy, so there is less uncertainty about the validity of the information being managed. Moreover, the time required by this approach is usually tractable, as reported in [52], although it strongly depends on the complexity of domain at hand. For highly complex domains the web mining approach – under human supervision – could be explored.

The left part of the T-Box in Figure F.1 depicts an excerpt of the ontology used in this work, defining rooms and objects usually found at homes. The top level sets the root, abstract concept `Thing`, with two children grouping the two types of elements that we will consider during the building of the map, namely `Rooms` and `Objects`. Rooms can belong to different concepts like `Kitchen`, `Bedroom`, etc., while examples of types of objects are `Microwave`, `Nightstand`, etc. The right part of the T-Box illustrates the simplified definitions of the concepts `Bedroom` and `Microwave`, codifying some of their properties and relations with other concepts.

3.2 Modeling space: the S-Box

The spatial box (S-Box) contains factual knowledge from the robot workspace, including the morphology and topology of the space, geometric/appearance information about the perceived spatial elements, symbols representing those elements, and beliefs concerning their grounding to concepts in the T-Box. The S-Box also adopts a hierarchical structure, ranging from sensory-like knowledge at the ground level to abstract symbols at the top one (see S-Box in Figure F.1). This representation is the common choice in the robotics community when dealing with large environments [74].

At the bottom of this hierarchy is the *spatial level*, which builds and maintains a metric map of the working space. *MvSmaps* do not restrict the employed metric map to a given one, but any geometric representation can be used, *e.g.* point-based [75], feature-based [76], or occupancy grid maps [1]. This map permits the robot to self-localize in a global frame, and also to locate the perceived elements in its workspace.

The top level of the S-Box is the *symbolic level*, envisioned to maintain an abstract representation of the perceived elements through *symbols*, including the robot itself (*e.g.* `obj-2`, `room-1`, `robot-1`, etc.), which are modeled as nodes. Arcs between nodes state different types of relations, as for example, objects connected by a relation of proximity (see *close* relations in the *symbolic level* in Figure F.1), or an object and a room liked by a relation of location (*at* relations). In this way, the symbolic level constitutes a topological representation of the environment, which can be used for global navigation and task planning purposes [77].

Finally, the intermediate level maintains the nexus between the S-Box and the T-Box. This level stores the outcome of an *anchoring process*, which performs the critical function of creating and maintaining the correspondence between percepts of the environment and symbols that refer to the same physical elements [14, 78]. The result is a set of the so-called *anchors*, which keep geometric/appearance information about the percepts (location, features, relations, etc.) and establish links to their symbolic representation. Additionally, in a *MvSmap* anchors are in charge of storing the beliefs about the grounding of their respective symbols, as well as their compatibility with respect to the grounding of related elements.

For illustrative purposes, the middle level in Figure F.1 exemplifies two anchors storing information of a percept from a microwave (in orange) and from a kitchen (in green). The coloured doted lines are pointers to their location in the metric map and their associated symbols, while the black doted lines point at the percepts of these elements from the environment. As an example, the outcome of a symbol grounding process is shown (field `Concept` within the anchor), which gives a belief for `obj-1` being grounded to `Microwave` and `Nightstand` of 0.65 and 0.35 respectively, while those for `room-1` are 0.33 for `Kitchen` and 0.67 for `Bedroom`. It is also shown the beliefs, or compatibility, for the symbols `obj-1` and `room-1` (related through the connection r_1) being grounded to certain pairs of concepts, *e.g.* 0.95 for `Microwave` and `Kitchen`, while 0.05 for `Microwave` and `Bedroom`.

3.3 Multiple semantic interpretations: the Multiverse

MvSmaps define the possible sets of symbols' groundings as *universes*. For example, by considering only the elements represented by `obj-1` and `room-1` in Figure F.1, four universes are possible: U_1:{(`obj-1` *is-a* `Nightstand`), (`room-1` *is-a* `Kitchen`)}, U_2:{(`obj-1` *is-a* `Microwave`), (`room-1` *is-a* `Kitchen`)}, U_3:{(`obj-1` *is-a* `Nightstand`), (`room-1` *is-a* `Bedroom`)}, and U_4:{(`obj-1` *is-a* `Microwave`), (`room-1` *is-a* `Bedroom`)}. This multiverse considers the possible explanations to the elements in the robot workspace. Additionally, *MvSmaps* annotate universes with their probability of being the plausible one, computed as the joint probability of grounding the symbols to the different concepts, giving a measure of certainty abut the current understanding of the robot about its workspace. Thus, a universe can be understood as an instance of the codified ontology with a set of grounded symbols and annotated probabilities.

To highlight the importance of the multiverse, let's us consider the simplified scenario depicted in Figure F.1. Under the title *Multiverse*, the four possible universes are displayed, with their probabilities annotated in brackets along with their names. The coloured (green and orange) concepts in those universes state the symbols that are grounded to them. We can see how the most plausible universe, *i.e.* , combination of groundings, is *Universe 3 (U_3)* (represented with a bold border), which sets `obj-1` as a nightstand and `room-1` as a bedroom. Suppose now that the robot is commanded to store a pair of socks in the nightstand. If the robot relies only on the most probable universe, we could end up with our socks heated in the microwave. However, if the robot also considers other universes, it could be aware that *Universe 2 (U_2)* is also a highly probable one, considering it as a different interpretation of its knowledge. In this case the robot should disambiguate both understandings of the workspace by, for example, gathering additional information from the environment, or in collaboration with humans.

It is worth mentioning that the information encoded in the Multiverse can be exploited, for example, by probabilistic conditional planners (*e.g.* those in [27] or [28]) for achieving a more coherent robot operation. Also, when a certain universe reaches a high belief, it could be considered as the ground, categorical truth, hence enabling the execution of logical inference engines like Pellet [79], FaCT++ [80], or Racer [81].

3.4 Formal description of *MvSmaps*

Given the ingredients of *MvSmaps* provided in the previous sections, a *Multiversal Semantic Map* can be formally defined by the quintuple $\mathscr{M}v\mathscr{S}map = \{\mathscr{R}, \mathscr{A}, \mathscr{Y}, \mathscr{O}, \mathscr{M}\}$, where:

- $\mathscr{R}$ is the metric map of the environment, providing a global reference frame for the observed spatial elements.
- $\mathscr{A}$ is a set of anchors internally representing such spatial elements, and linking them with the set of symbols in $\mathscr{Y}$.

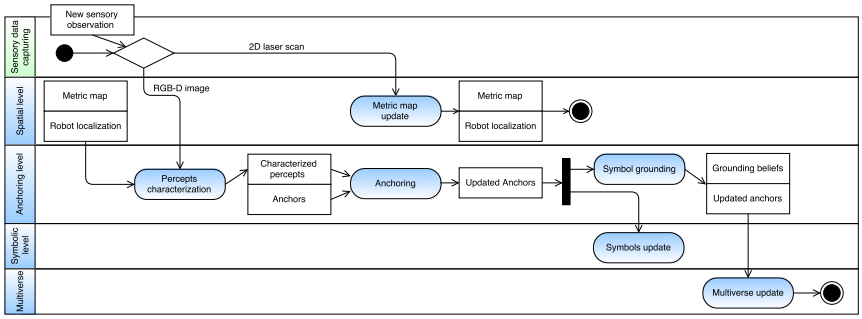


Figure F.2: UML activity diagram illustrating the pipeline for the building and maintaining of a *MvSmap* according to the sensory information gathered during the robot exploration. Blue rounded boxes are processes, while white shapes stand for consumed/generated data. The processes or data related to the same component of the semantic map are grouped together.

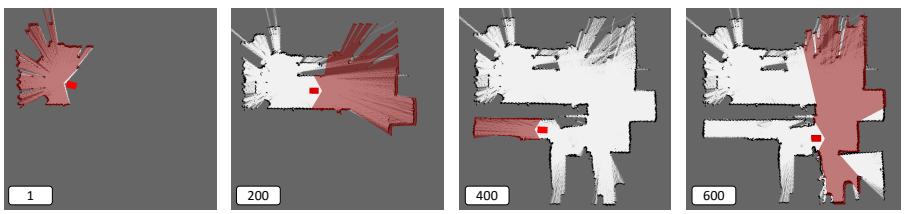


Figure F.3: Example of the progressive building of an occupancy grid map from a home environment. The 2D laser scans in red are the scans currently being aligned with the map, while the red boxes represent the estimated robot location. White cells in the map stand for free space, while black ones are occupied areas. Grey cells represent unknown space. Quantities in boxes are the number of scans registered so far to build the corresponding map.

- $\mathscr{Y}$ is the set of symbols that represent the spatial elements as instances of concepts from the ontology $\mathscr{O}$.
- $\mathscr{O}$ is an ontology codifying the semantic knowledge of the domain at hand.
- $\mathscr{M}$ encodes the multiverse, containing the set of universes.

Notice that the traditional T-Box and S-Box are defined in a *MvSmap* by $\mathscr{O}$ and $\{\mathscr{R}, \mathscr{A}, \mathscr{Y}\}$ respectively. Since the robot is usually provided with the ontology $\mathscr{O}$ beforehand, building a *MvSmap* consists of creating and maintaining the remaining elements in the map definition, as described in the next section.

4 Building the Map

This section describes the processes involved in the building of a *MvSmap* for a given environment according to the sensory information gathered by a mobile robot (see Figure F.2). In our discussion, we assume that the robot is equipped with a 2D range laser scanner and a RGB-D camera, two sensors commonly found in robotic platforms, although they could be replaced by any other sensory system able to survey the spatial elements in the environment.

In a nutshell, when a new 2D laser scan is available, it triggers the update of the 2D metric map $\mathscr{R}$ in the *spatial level* (see Section 4.1). In its turn, if a new RGB-D observation is collected, it is processed in order to characterize the percepts of the surveyed room and the objects therein, as well as their contextual relations (see Section 4.2). The characterized percepts fed an anchoring process that compares them with those from previously perceived elements, which are stored in the form of *anchors* in the *anchoring level* (see Section 4.3). When a percept is matched with a previous one, its corresponding anchor is updated, otherwise a new anchor, including a new symbol in the *symbolic level*, is created. Finally, the information encoded in the *anchoring level* is used to build a Conditional Random Field, which is in charge of grounding the symbols of the spatial elements to concepts in the T-Box, also providing a measure of the uncertainty concerning such groundings in the form of beliefs (see Section 4.4). These beliefs are stored in the anchors, and are employed to update the multiverse $\mathscr{M}$. The next sections describe the core processes of this pipeline in detail.

4.1 Building the underlying metric map

During the robot exploration, the collected 2D laser scans are used to build a metric representation of the environment in the form of an occupancy grid map [1]. For that, we rely on standard Simultaneous Localization and Mapping (SLAM) techniques to jointly build the map and estimate the robot pose [82].

Thus, the building process is based on an Iterative Closet Point (ICP) algorithm [83], which aligns each new scan to the current reference map. Once aligned, the scan measurements are inserted into the map, hence building it incrementally. Given that the robot is also localized in the map at any moment, the spatial information coming from the sensors mounted on it (*e.g.* RGB-D cameras) can be also located. For that, those sensors have to be extrinsically calibrated, that is, the sensors' position in the robot local frame must be known. Figure F.3 shows an example of the incremental building of a metric map from an apartment in the Robot@Home dataset [29].

4.2 Characterizing percepts

Concurrently with the metric map building, when a RGB-D observation is collected it is processed in order to characterize the percepts of the spatial elements therein. This information is required by the posterior anchoring process, so it can decide which

percepts correspond to elements previously observed and which ones are perceived for the first time, being consequently incorporated to the semantic map.

Typically, a RGB-D observation contains a number of percepts corresponding to objects, while the whole observation itself corresponds to the percept of a room (see Figure F.6-left). On the one hand, objects' percepts are characterized through geometric (planarity, linearity, volume, etc.) and appearance features (*e.g.* hue, saturation, and value means). On the other hand, room percepts are prone to not cover the entire room, *i.e.* it is common to not survey the whole room with a single RGB-D observation, so the extracted geometric and appearance features (footprint, volume, hue, saturation and value histograms, etc.) are, in addition, averaged over time by considering those from past room percepts. Moreover, the metric map hitherto built for that room is also considered and characterized, since it supposes a rich source of information for its posterior categorization [38]. The upper part of Table F.1 lists the features used to describe those percepts.

In addition to objects and rooms, the contextual relations among them are also extracted and characterized. We have considered two types of relationships, one linking objects that are placed closer than a certain distance (*close*), and another one relating an object and its container room (*at*). The lower part of Table F.1 lists the features employed to characterize such relations. It is worth mentioning the function of the *bias* feature characterizing the object–room relations, which is a fixed value that permits the CRF to automatically learn the likelihood of finding a certain object type into a room of a certain category (see Section 4.4). The outcome of this characterization process is known as the *signature* of the percept.

4.3 Modeling and keeping track spatial elements: Anchoring

Once characterized, the percepts feed an *anchoring process* [14], which establishes the correspondences between the symbols of the already perceived spatial elements (*e.g.* `obj-1` or `room-1`) and their percepts. For that, it creates and maintains internal representations, called anchors, which include: the features of the spatial elements and their relations, their geometric location[2], their associated symbols, the beliefs about the groundings of those symbols, and their compatibility with the groundings of related elements. The content of an anchor was previously illustrated in the *anchoring level* in Figure F.1. In its turn, the sub-components of the anchoring process are depicted in Figure F.4.

Let $\mathscr{S}_{in} = \{s_1, \ldots, s_n\}$ be the set of characterized percepts surveyed in the last RGB-D observation. Then, the signatures of these percepts are compared with those of anchors already present in the semantic map, which produces two disjoint sets: the set $\mathscr{S}_{update}$ of percepts of spatial elements that have been previously observed in the environment, and the set $\mathscr{S}_{new}$ of percepts of elements detected for the first time. We

[2]Notice that although the underlying metric map is 2D, the extrinsic calibration of sensors can be used to locate an element in 6D (3D position and 3D orientation).

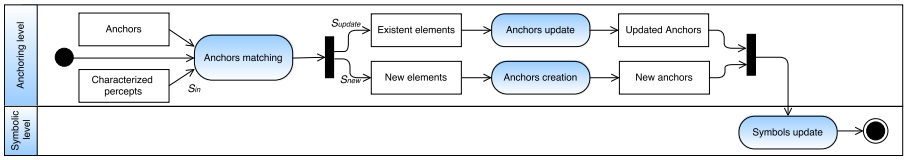


Figure F.4: UML activity diagram showing the sub-processes (blue rounded boxes) and consumed/produced data (white shapes) involved in the anchoring process.

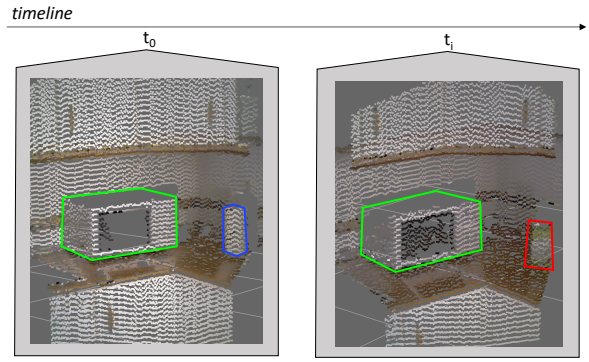


Figure F.5: Example of the matching step within the anchoring process, showing two point clouds gathered from a kitchen at different time instants. The green shapes contain percepts that are matched as belonging to the same spatial element, while the percepts enclosed in the blue and red ones have been correctly considered as corresponding to different elements due to their different appearance (they contain a paper roll and a milk bottle respectively).

have considered a simple but effective matching algorithm that checks the location of two percepts, the overlapping of their bounding boxes, and their appearance to decide if they refer to the same physical element.

The two sets of percepts resulting from the matching step are processed differently: while the set $\mathscr{S}_{update}$ triggers the update of their associated anchors, *i.e.* their locations, features, and relations are revised according to the new available information, the set $\mathscr{S}_{new}$ produces the creation of new anchors. As a consequence, the content of the *symbolic level* is also revised: the symbols representing updated anchors are checked for possible changes in their relations, while new symbols are created for the new anchors. As an example, Figure F.5 shows two point clouds representing RGB-D images gathered from the same kitchen at different time instants. At time t_0, two new anchors are created for accommodating the information from the two percepts (highlighted in green and blue). Then, at time t_1, the signature of the percept in green is matched with the one with the same color at t_0, while the percept in red, despite their similar location and size, is considered different from the one in blue

Table F.1: Features used to characterize the percepts (objects and rooms) and contextual relations among them (object-object and object-room). These features are grouped according to their type, geometric or appearance, stating in parentheses the type of information from where they come, RGB-D images or metric maps. Values in parentheses in the features' names give the number of features grouped under the same name (for example the centroid of an object has x, y and z coordinates).

Object
Geometric (RGB-D)
Planarity
Scatter
Linearity
Min. height
Max. height
Centroid (3)
Volume
Biggest area
Orientation
Appearance (RGB-D)
H, S, V, means (3)
H, S, V, Stdv. (3)
H, S, V, histograms (15)

Room
Geometric (RGB-D)
Scatter (2)
Footprint (2)
Volume (2)
Appearance (RGB-D)
H, S, V, means (6)
H,S,V, Stdv. (6)
H, S, V, histograms (30)
Geometric (Metric map)
Elongation
Scatter
Area
Compactness
Linearity

Object-Object
Geometric (RGB-D)
Perpendicularity
Vertical distance
Volume ratio
Is on relation
Appearance (RGB-D)
H, S, V, mean diff.
H, S, V, Stdv. diff.

Object-Room
Bias

at t_0 due to their appearance, and a new anchor is created. Notice that to complete the aforementioned content of anchors the beliefs about the grounding of their symbols, as well as the compatibility with the groundings of related elements, must be computed. This is carried out by the probabilistic techniques in the next section.

Although the described anchoring process could appear similar to a tracking procedure, it is more sophisticated regarding the information that is stored/managed. For example, in typical tracking problems, it is usually not needed to maintain a symbolic representation of their tracks, nor to ground them to concepts within a knowledge base. Further information in this regard can be found in the work by Coradeschi and Saffiotti [14].

4.4 Probabilistic symbol grounding

We holistically model the symbol grounding problem employing a Conditional Random Field (CRF) (see Section 4.4), a probabilistic technique first proposed by Lafferty *et al.*[84] that, in addition to exploiting the relations among objects and rooms, also provides the beliefs about such groundings through a probabilistic inference process (see Section 4.4). These belief values are the main ingredients for the generation and update of the multiverse in the *MvSmap* (see Section 4.5).

CRFs to model the symbol grounding problem

The following definitions are required in order to set the problem from this probabilistic stance:

- Let $s = [s_1, .., s_n]$ be a vector of n of spatial elements, stating the observed objects or rooms in the environment, which are characterized by means of the features in their associated anchors.
- Define $L_o = \{l_{o_1}, .., l_{o_k}\}$ as the set of the k considered object concepts (*e.g.* `Bed`, `Oven`, `Towel`, etc.).
- Let $L_r = \{l_{r_1}, .., l_{r_j}\}$ be the set of the j considered room concepts (*e.g.* `Kitchen`, `Bedroom`, `Bathroom`, etc.).
- Define $y = [y_1, .., y_n]$ to be a vector of discrete random variables assigning a concept from L_o or L_r to the symbol associated with each element in s, depending on whether such symbol represents an object or a room.

Thereby, the grounding process is jointly modeled by a CRF through the definition of the probability distribution $P(\mathbf{y} \mid s)$, which yields the probabilities of the different assignments to the variables in $\mathbf{y}$ conditioned on the elements from s. Since its exhaustive definition is unfeasible due to its high dimensionality, CRFs exploit the concept of independence to break this distribution down into smaller pieces. Thus, a CRF is represented as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where the set of nodes $\mathcal{V}$ models the random variables in $\mathbf{y}$, and the set of undirected edges $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ links contextually related nodes. Notice that this graph can be built directly from the codified information within the *symbolic level*. Thus, mimicking the representation in that level, the same types of edges are considered in the CRF: proximity of two objects, and presence of an object into a room. Intuitively, this means that, for a certain object, only the nearby objects in the environment and its container room have a direct influence on its grounding, while the grounding of a room is affected by the objects therein. Figure F.6-right shows an example of a CRF graph built from the spatial elements in the observation depicted in Figure F.6-left, also including elements that were perceived in previous observations of the same room and were stored in the S-Box.

According to the Hammersley-Clifford theorem [85], the probability $P(\mathbf{y} \mid s)$ can be factorized over the graph $\mathcal{G}$ as a product of *factors* $\psi(\cdot)$:

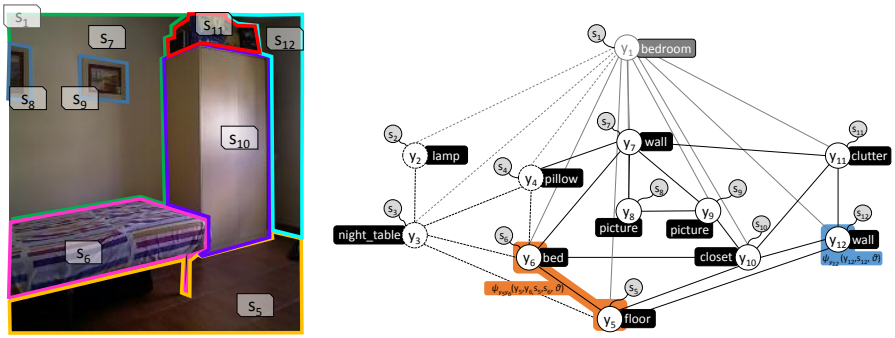


Figure F.6: Left, RGB image from a RGB-D observation of a sequence where the robot is exploring a bedroom. The objects' percepts are enclosed in coloured shapes and represented by s_5-s_{12}, while the whole image is considered the room percept and is represented by s_1. Right, CRF graph representing the spatial elements and relations in such image as random variables and edges respectively (solid lines), as well as the elements and relations from previously surveyed objects (doted lines, represented as $s_2 - s_4$). The area highlighted in blue states the scope of an unary factor, while the one in orange stands for the scope of a pairwise factor.

$$p(\mathbf{y}|s;\theta) = \frac{1}{Z(s,\theta)} \prod_{c\in\mathscr{C}} \psi_c(y_c, s_c, \theta) \tag{F.1}$$

where $\mathscr{C}$ is the set of maximal cliques[3] of the graph $\mathscr{G}$, and $Z(\cdot)$ is the also called partition function, which plays a normalization role so $\sum_{\xi(\mathbf{y})} p(\mathbf{y}|s;\theta) = 1$, being $\xi(y)$ a possible assignment to the variables in y. The vector θ stands for the model parameters (or weights) to be tuned during the training phase of the CRF. Factors can be considered as functions encoding pieces of $P(\mathbf{y} \mid s)$ over parts of the graph. Typically, two kind of factors are considered: *unary factors* $\psi_i(y_i, s_i, \theta)$, which refer to nodes and talk about the probability of a random variable y_i belonging to a category in L_o or L_r, and *pairwise factors* $\psi_{ij}(y_i, y_j, s_i, s_j, \theta)$ that are associated with edges and state the compatibility of two random variables (y_i, y_j) being tied to a certain pair of categories. As a consequence, the cliques used in this work have at most two nodes (see Figure F.6-right). The expression in Eq.F.1 can be equivalently expressed for convenience through log-linear models and exponential families as [86]:

$$p(y|s;\theta) = \frac{1}{Z(s,\theta)} \prod_{c\in C} \exp(\langle \phi(s_c, y_c), \theta \rangle) \tag{F.2}$$

being $\langle \cdot, \cdot \rangle$ the inner product, and $\phi(s_c, y_c)$ the sufficient statistics of the factor over the clique c, which comprises the features extracted from the spatial elements (recall Table F.1). Further information about this representation can be found in [55].

[3] A maximal clique is a fully-connected subgraph that can not be enlarged by including an adjacent node.

Training a CRF model for a given domain requires the finding of the parameters in θ, in such a way that they maximize the likelihood in Eq.F.2 with respect to a certain i.i.d. training dataset $\mathscr{D} = [d^1, \dots d^m]$, that is:

$$\max_{\theta} \mathscr{L}_p(\theta : \mathscr{D}) = \max_{\theta} \prod_{i=1}^{m} p(\mathbf{y}^i \mid s^i; \theta) \tag{F.3}$$

where each training sample $d^i = (\mathbf{y}^i, s^i)$ consists of a number of characterized spatial elements (s^i) and the corresponding ground truth information about their categories ($\mathbf{y}^i$). If no training dataset is available for the domain at hand, the codified ontology can be used to generate synthetic samples for training, as we have shown in our previous work [51, 55]. The optimization in Eq.F.3 is also known as Maximum Likelihood Estimation (MLE), and requires the computation of the partition function $Z(\cdot)$, which in practice turns this process into a $\mathscr{NP}$-hard, hence intractable problem. To face this in the present work, the calculus of $Z(\cdot)$ is estimated by an approximate inference algorithm during the training process, concretely the *sum-product* version of the *Loopy Belief Propagation* (LBP) method [56], which has shown to be a suitable option aiming at categorizing objects [23].

Performing probabilistic inference

Once the CRF representation modeling a given environment is built, it can be exploited by probabilistic inference methods to perform different probability queries. At this point, two types of queries are specially relevant: the *Maximum a Posteriori* (MAP) query, and the *Marginal* query. The goal of the MAP query is to find the most probable assignment $\hat{\mathbf{y}}$ to the variables in $\mathbf{y}$, *i.e.* :

$$\hat{y} = arg\max_{\mathbf{y}} p(\mathbf{y} \mid s; \theta) \tag{F.4}$$

Once again, the computation of the partition function $Z(\cdot)$ is needed, but since given a certain CRF graph its value remains constant, this expression can be simplified by:

$$\hat{y} = arg\max_{\mathbf{y}} \prod_{c \in C} \exp(\langle \phi(s_c, y_c), \theta \rangle) \tag{F.5}$$

Nevertheless, this task checks every possible assignment to the variables in $\mathbf{y}$, so it is still unfeasible. An usual way to address this issue is the utilization of approximate methods, like the *max-product* version of LBP [87]. The alert reader may think that, in the end, the MAP assignment provides crispy results. Although this is undoubtedly true, the computation of those results considers both the relations among the spatial elements in the environment, and the belief about their belonging to different categories, so it is clearly differentiated from the crispy results given by an off-the-shelf categorization method working on individual elements. The black boxes

in Figure F.6-right show an example of the outcome of a MAP query over the defined CRF graph.

In its turn, the Marginal query, which can be performed by the aforementioned *sum-product* version of LBP, provides us the beliefs about the possible groundings. In other words, this query yields the marginal probabilities for each symbol being grounded to different concepts, as well as the compatibility of these groundings with respect to the grounding of contextually related symbols. Therefore, it is also possible to retrieve the probability of a certain assignment to the variables in $\mathbf{y}$, which is of interest for managing universes (see Section 4.5). Recall that, in a *MvSmap*, these beliefs are stored in their corresponding anchors for their posterior exploitation during the robot operation (see anchors in Figure F.1). Section 5 will show both MAP and Marginal queries in action.

4.5 Managing the Multiverse

To conclude the building of the *MvSmap*, the outcome of the marginal query is exploited to generate and update the multiverse. The probability for each possible universe can be retrieved by means of Eq.F.1, replacing the factors $\psi(\cdot)$ by the provided beliefs $b(\cdot)$, and the partition function $Z(\cdot)$ by its approximation $Z_{LBP}(\cdot)$ computed by the LBP algorithm, that is:

$$p(\mathbf{y}|s;\theta) = \frac{1}{Z_{LBP}(s,\theta)} \prod_{c \in \mathscr{C}} b_c(y_c, s_c) \tag{F.6}$$

The exhaustive definition of such multiverse, that is, to compute and store the probabilities and groundings in each possible universe, highly depends on the complexity of the domain at hand. The reason for this is that the number of possible universes depends on both, the number of spatial elements, and the number of concepts defined in the ontology. For example, let's suppose a domain with 3 types of rooms and 4 types of objects. During the robot exploration, 5 objects have been observed within 2 rooms, so a total of $4^5 \times 3^2 = 9,216$ possible interpretations, or universes, exist. This is a large number for a small scenario, but it supposes a reduced size in memory since each universe is defined by: (i) its probability, and (ii) its grounded symbols. Concretely, in this case each universe can be codified through a *float* number for its probability (4 bytes) and 7 *char* numbers for the groundings (7 bytes in total, supposing that each concept can be identified by a *char* number as well), so the size of the multiverse is $11 \times 9,216 = 99kB$. Notice that such a size grows exponentially with the number of spatial elements, so in crowded environments this exhaustive definition is unpractical, or even unfeasible.

In those situations, the exhaustive definition can be replaced by the generation of the more relevant universes for a given task and environment. Thus, for example, the MAP grounding yielded by a MAP query permits the definition of the most probable universe. Recall that the probability of this or other universes of interest can be retrieved by inserting their respective groundings and stored beliefs in Eq.F.6. Other probable universes can be straightforwardly identified by considering the ambiguous

groundings. For example, if an object is grounded to concepts with the following beliefs {Bowl 0.5, Milk-bottle 0.45, Microwave 0.05}, and the MAP query grounds it to Bowl, it makes sense to also keep the universe where the object is grounded to Milk-bottle, and *vice versa*. As commented before, the set of relevant universes is task and domain dependant so, if needed, they should be defined strategies for their generation in order to keep the problem tractable.

To tackle this issue we propose a simple but practical strategy based on the utilization of a threshold, or *ambiguity factor*, that determines when a grounding result is ambiguous. For that, if the ratio between the belief about a symbol being grounded to a certain concept (b_i) and the highest belief for that symbol (b_h) is over this threshold (α), then these two possible groundings are considered ambiguous. Mathematically:

$$ambiguous(b_i, b_h) = \begin{cases} 1 \text{ (true)} & \text{if } b_i/b_h > \alpha \\ 0 \text{ (false)} & \text{otherwise} \end{cases} \tag{F.7}$$

Therefore, if a pair of grounding values are ambiguous according to this strategy, their associated universes are considered relevant, being consequently stored in the multiverse. Continuing with the previous example, the ratio between the beliefs for Milk-bottle and Bowl is $0.45/0.5 = 0.9$, while between Microwave and Bowl is $0.05/0.5 = 0.1$. Thus, with a value for α higher than 0.1 and lower than 0.9, this strategy would consider the first pair of groundings as ambiguous, but not the second one. The efficacy of this strategy for keeping the number of universes low, without disregarding relevant ones, is shown in Section 5.3.

5 Experimental Evaluation

To evaluate the suitability of both, the proposed probabilistic symbol grounding as well as the novel semantic map, we have carried out a number of experiments using the challenging Robot@Home [29] dataset, which is briefly described in Section 5.1. More precisely, to test the symbol grounding capabilities of our approach (see Section 5.2), it has been analyzed its performance both (i) when grounding object and rooms symbols in isolation, *i.e.* using the traditional categorization approach that works with the individual features of each spacial element (see Section 5.2), and (ii) when also considering the contextual relations among elements (see Section 5.2). To conclude this evaluation, we also describe some sample mapping scenarios in Section 5.3, aiming to illustrate the benefits of the proposed *MvSmap*.

5.1 Testbed

The Robot@Home dataset provides 83 sequences containing 87,000+ observations, divided into RGB-D images and 2D laser scans, which survey rooms of 8 different types summing up $\sim$1,900 object instances. From this repository we have extracted 47 sequences captured in the most common room types in home environments, namely:

Table F.2: Performance of baseline methods individually grounding objects and rooms. Rows index the results employing features of different nature, while columns index the different methods (CRF: Conditional Random Fields, SVM: Supported Vector Machines, NB: Naive Bayes, DT: Decision Tress, RF, Random Forests, NN: Nearest Neighbors). Please refer to App. A for a description of the used performance metrics.

	CRF			SVM	NB	DT	RF	NN
Objects	Macro p./r.	Micro p.	Micro r.	Macro p./r.	Macro p./r.	Macro p./r.	Macro p./r.	Macro p./r.
Geometric	72.86%	52.12%	42.41%	62.84%	66.67%	71.61%	73.20%	40.69%
Appearance	34.08%	18.50%	14.58%	33.72%	19.07%	25.25%	33.41%	16.39%
Geometric + Appearance	**73.64**%	**53.30**%	**51.62**%	**71.06**%	**70.00**%	**72.38**%	**74.53**%	**43.04**%
Rooms	Macro p./r.	Micro p.	Micro r.	Macro p./r.	Macro p./r.	Macro p./r.	Macro p./r.	Macro p./r.
Geometric (RGB-D)	25.53%	22.92%	18.33%	32.60%	25.00%	7.40%	22.50%	21.40%
Geometric (Metric map)	27.66%	16.25%	17.38%	40.20%	32.10%	**43.80**%	45.30%	29.80%
Geometric (All)	46.81%	36.64%	37.94%	**41.70**%	28.30%	37.90%	**52.50**%	36.10%
Appearance	44.68%	38.43%	35.73%	37.80%	32.60%	22.10%	42.40%	28.90%
Geo. (All) + Appearance	**57.45**%	**50.09**%	**48.12**%	37.40%	**38.20**%	37.90%	37.40%	**44.00** %

Figure F.7: Robotic platform used to collect the Robot@Home dataset.

bathrooms, bedrooms, corridors, kitchens, living-rooms and master-rooms. These sequences contain ∼1,000 instances of objects that belong to one of the 30 object types considered in this work, *e.g.* bottle, cabinet, sink, toilet, book, bed, pillow, cushion, microwave, bowl. etc.

The observations within the sequences come from a rig of 4 RGB-D cameras and a 2D laser scanner mounted on a mobile robot (see Figure F.7). However, to match this sensory configuration with one more common in robotic platforms, we have only considered information from the 2D laser scanner and the RGB-D camera looking ahead.

5.2 Probabilistic symbol grounding evaluation

In this section we discuss the outcome of a number of experiments that evaluate different configurations for the probabilistic symbol grounding process. To obtain the performance measurements (micro/macro precision/recall, see App. A), a *MvSmap* has been built for each sequence, and MAP queries are executed over the resultant CRFs (recall Section 4.4). Concretely, a leave-one-out cross-validation technique is followed, where a sequence is selected for testing and the remaining ones for training. This process is repeated 47 times, changing the sequence used for testing, and the final performance is obtained averaging the results yielded by those repetitions.

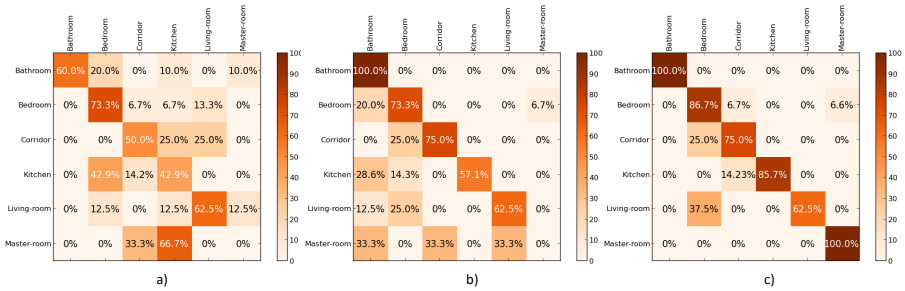


Figure F.8: Confusion matrices relating the ground truth information about rooms (rows) with the concept to which they are grounded (columns). a) Confusion matrix for a CRF only employing nodes, b) including object-room relations, and c) considering all the contextual relations.

Individual grounding of object and room symbols

The aim of this section is to evaluate the performance of our proposal without exploring contextual relations, *i.e.* only considering the geometric/appearance features characterizing the symbols. This *individual grounding* is the traditional approach in semantic mapping, and permits us to set a baseline for measuring the real enhancement of the joint grounding in the next section. Thereby, only the nodes in the CRFs have been considered, characterized by the *object* and *room* features in Table F.1.

The first three columns in Table F.2 report the results for grounding object and room symbols according to the described configuration. For objects, we can see how the used geometric features are more discriminative than the appearance ones, but their complementary nature makes that the CRFs resorting to their combination achieves the highest results (73.64%). The same happens when grounding rooms, where the winning option, reaching a performance of 57.45%, combines geometric and appearance features from the RGB-D observations, as well as geometric features from the part of the metric map corresponding to the room.

To complete this baseline, they have been also evaluated some of the most popular classifiers also resorting to individual object/room features. In order to make this comparison as fair as possible the same features employed for the CRFs have been used, as well as the same leave-one-out cross-validation approach. Concretely, we have resorted to the implementation in the `scikit-learn` library [88] of the following widely-used methods[4]: Supported Vector Machines, Naive Bayes, Decision Trees, Random Forests, and Nearest Neighbors. The yielded results are reported in the last five columns of Table F.2, where it is shown how the CRF achieve a similar or even higher success than those classifiers. In fact, the more serious competitor is the one based on Random Forests, which achieves a $\sim 1\%$ higher success when categorizing objects, but a $\sim 5\%$ lower one when dealing with rooms.

[4]Further information about these classifiers can be found in the library webpage: http://scikit-learn.org/

Table F.3: Performance for grounding symbols of CRFs exploiting contextual information. Rows index the type of contextual relations modeled by the CRFs. App. A describes the used metrics.

Objects	Macro p./r.	Micro p.	Micro r.
Object-Object	78.70%	65.58%	53.34%
Object-Room	78.69%	59.38%	53.09%
Object-Object + Object-Room	**81.58**%	**70.71**%	**60.94**%
Rooms	Macro p./r.	Micro p.	Micro r.
Object-Room	80.85%	65.08%	61.33%
Object-Object + Object-Room	**91.49**%	**85.25** %	**84.98**%

Table F.4: Example of the outcome of a grounding process where the contextual relations modeled in a CRF help to disambiguate wrong individual groundings. The first column states the symbols' names, the second one their ground truth category, while the third and fourth columns report the two categories that received the highest beliefs (in parentheses) after a Marginal inference query. The MAP assignment is highlighted in bold.

Symbol	Ground truth	Beliefs	
`obj-3`	Microwave	**Microwave (0.38)**	Nightstand (0.29)
`obj-5`	Counter	Table (0.39)	**Counter (0.30)**
`obj-9`	Counter	**Counter (0.26)**	Table (0.12)
`room-1`	Kitchen	Bedroom (0.49)	**Kitchen (0.22)**

Joint object-room symbol grounding

This section explores how the progressive inclusion of different types of contextual relations to the CRFs affects the performance of the grounding method. Table F.3 gives the figures obtained from this analysis. Taking a closer look at it, we can see how the inclusion of contextual relations among objects increases the success of grounding them by $\sim 5\%$. By only considering relations among objects and rooms, the performance of grounding objects is increased almost the same percentage, while the success of rooms considerably grows from 57.45% up to 80.91%. Finally, with the inclusion of all the contextual relations, the reached grounding success is of 81.58% and 91.49% for objects and rooms respectively. Comparing these numbers with the baseline performance obtained in the previous section also employing CRFs, they achieve a notorious increment in the performance of $\sim 8\%$ for objects and $\sim 34\%$ for rooms. This approach also clearly outperforms the success reported by the other methods in Table F.2.

Table F.5: Example of grounding results yielded by the proposed method for the symbols within a simple kitchen scenario. The first and the second columns give the symbols' names and their ground truth respectively, while the remaining columns report the five categories with the highest beliefs (in parentheses) as yielded by a Marginal inference query. The MAP assignment is highlighted in bold.

Symbol	Ground truth	Beliefs		
`obj-1`	Microwave	**Nightstand (0.46)**	Microwave (0.42)	Wall (0.06)
`obj-2`	Counter	**Counter (0.70)**	Bed (0.24)	Floor (0.04)
`obj-3`	Wall	**Wall (0.99)**	Counter (0.1)	Nightstand (0.0)
`obj-4`	Wall	**Wall (0.99)**	Bed (0.01)	Microwave (0.0)
`obj-5`	Floor	**Floor (0.99)**	Bed (0.01)	Wall (0.0)
`room-1`	Kitchen	**Bedroom (0.51)**	Kitchen (0.22)	Bathroom (0.19)
Symbol	Ground truth	Beliefs		
`obj-1`	Microwave	Bed (0.04)	Counter (0.04)	Floor(0.1)
`obj-2`	Counter	Wall (0.01)	Nightstand (0.01)	Microwave (0.0)
`obj-3`	Wall	Floor (0.0)	Microwave (0.0)	Bed (0.0)
`obj-4`	Wall	Nightstand (0.0)	Floor (0.0)	Counter (0.0)
`obj-5`	Floor	Counter (0.0)	Nightstand (0.0)	Microwave (0.0)
`room-1`	Kitchen	Living-room (0.06)	Master-roomr (0.01)	Corridor (0.01)

Figure F.8 depicts the confusion matrices obtained while grounding room symbols for each of the aforementioned configurations. In these matrices, the rows index the room ground truth, while the columns index the grounded concept. We can notice how the performance reported in these matrices improves progressively (the values in their diagonals grow) with the inclusion of contextual relations.

To further illustrate the benefits of the conducted joint symbol grounding, Table F.4 shows the results of the grounding of a number of symbols from a kitchen sequence. The third and fourth columns of this table report the concepts with the two highest beliefs for each symbol, retrieved by a Marginal inference query over the CRF built from such sequence. A traditional grounding approach would only consider the concepts in the third row, while our holistic stance is able to provide the results highlighted in bold (through a MAP query), which match the symbols' ground truth.

5.3 Sample mapping scenarios

In this section we exemplify the building of *MvSmaps* for two scenarios exhibiting different complexity. We start by describing a simple scenario where the possible object categories are: floor, wall, counter, bed, nightstand, and microwave. The possible room categories are the same as in the previous section. This is an extension in a real setting of the toy example described in Section 3. The chosen sequence of ob-

servations from Robot@Home corresponds to a kitchen containing 5 objects of these categories: a counter, a microwave, two walls and the floor. Thus, the *MvSmap* built for that scenario consist of (recall Section 3.4):

- An occupancy grid map of the explored room.
- 6 anchors representing the spatial elements (5 objects and a room).
- 6 symbols in the symbolic level.
- An ontology of the home domain.
- $6^5 \times 6^1 = 46,656$ possible universes, which supposes a multiverse size of $\sim 456kB$.

Table F.5 shows the grounding results yielded by the execution of MAP and Marginal queries over the CRF representation of such map. We can see how the MAP assignment fails at grounding the symbols `obj-1` and `room-1`, but the right groundings of such symbols also receive a high belief value. As a consequence of this, their respective universes could also exhibit high probabilities, hence the importance of their consideration. Notice that the size of the multiverse could be further reduced by applying the previously proposed strategy. For example, considering an ambiguity factor of $\alpha = 0.2$, the number of possible universes is 12, being the size (in memory) of the multiverse of only 132 bytes.

We also describe a more complex scenario considering the room and object categories introduced in Section 5.1. In this case, we discuss the progressive building of the *MvSmap* at 4 different time instants during the robot exploration of a bedroom. Figure F.9 depicts the evolution of the groundings of the spatial elements perceived by the robot during such exploration, where the big and small coloured boxes represent the groundings with the two highest beliefs. In this case, the groundings provided by MAP queries match with those showing the highest beliefs.

We can see how until the time instant t_1 the robot surveyed 8 objects, being so confident about the category of 5 of them. This supposes a total of 9 anchors and 9 symbolic representations (8 objects plus a room). The most ambiguous result is for an object placed on the bed, which is in fact a towel. This ambiguity is due to the features exhibited by the object, its position, and its unusual location in a bedroom. In its turn, the belief about the room being grounded to the `Bedroom` concept is high, 0.76, as a result of the surveyed spatial elements and their relations. Until time t_2 the room is further explored, appearing three new objects: a chair, a table and a wall, hence adding 3 new anchors and their respective symbols to the *MvSmap*. The surveyed table is the only one showing an ambiguous grounding because of its features and few contextual relations. However, in the observations gathered until the time instant t_3, two new objects are perceived on top of the table, a book and a bottle, increasing the belief value about its grounding to the `Table` concept. With these new objects and relations the uncertainty about the category of the room also decreases. Finally, considering all the information gathered until the time instant t_4, where a pillow has been observed on top of the bed, the belief about the room category increases up to 0.99. Notice how

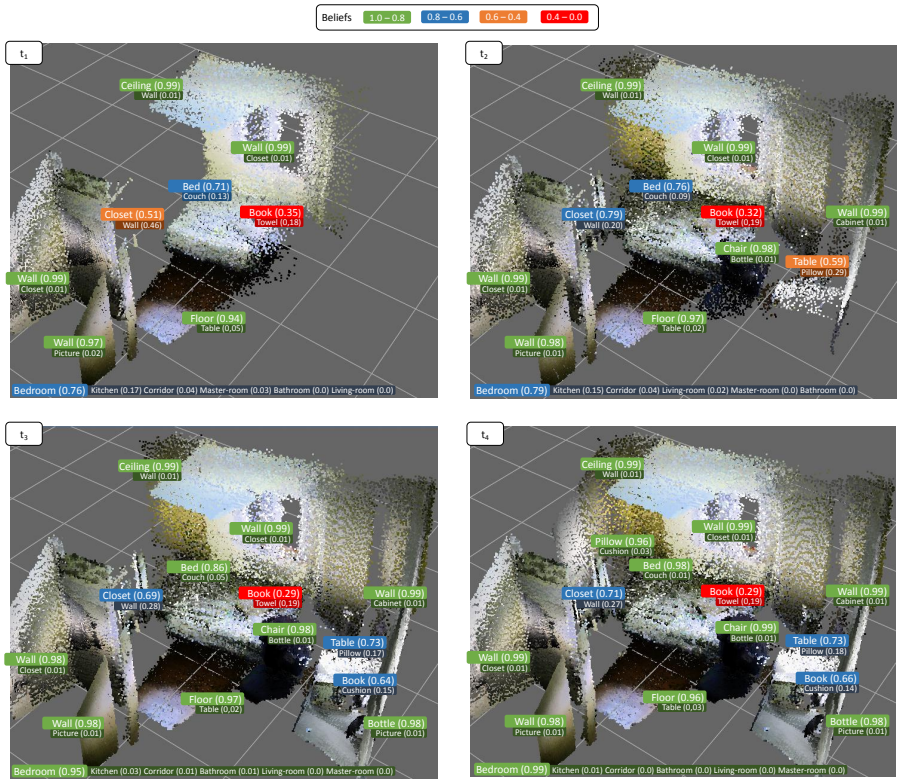


Figure F.9: Grounding results and their belief values for the symbols of spatial elements perceived during the robot exploration of a bedroom. The registered point clouds in each image are shown for illustrative purposes.

the detection of such pillow also decreases the uncertainty about the grounding of the bed. The *modus operandi* of *traditional* semantic maps is to consider the towel on the bed as a book, which can lead to, for example, the failure of a robot ordered to bring all the towels in the house to the bathroom. This can be tackled through the utilization of *MvSmaps* and the clarification of uncertain groundings.

Thereby, the *MvSmap* built in this scenario is compounded of 15 anchors (14 objects plus a room), 15 symbols at the symbolic level, and a total of $30^{14} \times 6^{1} \simeq 2.8 \times 10^{21}$ universes. This supposes a multiverse with an intractable size, however, applying the previous strategy where only uncertain results generate new universes, the size of the multiverse is considerably reduced to 40 universes and 760 bytes.

6 Potential Applications of Multiversal Semantic Maps

The main purpose of the proposed *MvSmap* is to provide a mobile robot with a probabilistic, rich representation of its environment, empowering the efficient and coherent execution of high-level tasks. For that, the *MvSmap* accommodates the uncertainty about the grounded concepts as universes, which can be seen as different interpretations of the workspace. Notice that *MvSmaps* can be exploited for traditional semantic map applications (*e.g.* task planning, planning with incomplete information, navigation, human-robot interaction, localization, etc.) by considering only a universe, albeit its potential to measure the (un)certainty of the robot's understanding can be exploited for an intelligent, more efficient robotic operation.

A clear example of this can be envisioned while planning an object search task. Let's suppose an scenario where the robot is commanded to bring the slippers to the user. If the slippers have not been detected before, the robot could infer (according to its semantic knowledge) that their most probable location is a bedroom. Fortunately, a room, corresponding to the farthest one from the robot location, has been already grounded as being a bedroom with a belief of 0.42, and 0.41 of being a kitchen. Another room, close to the robot location, has been grounded to the `Kitchen` concept with a belief of 0.47, and to the `Bedroom` one with 0.45. The utilization of only the most probable universe would lead to the exploration of the farthest room, with a 42% of being the correct place, while the consideration of both interpretations would produce the more logical plan of taking a look at the closer one first. Moreover, the Conditional Random Field employed in this work is able to provide a more fine-grained and coherent prediction than just employing semantic knowledge: it permits to hypothesize about the exact location of an object or a room, and to retrieve the likelihood of such location through an inference method [48, 16]. By repeating this process in different locations, the robot can operate according to a list of possible object locations ordered by their likelihood.

Another typical application of semantic maps resorting to logical reasoning engines is the classification of rooms according to the objects therein [25]. For example, if an object is grounded as a refrigerator, and kitchens are defined in the Knowledge Base as rooms containing a refrigerator, a logical reasoner can infer that the room is a kitchen. Again, this reasoning relying on crispy information can provoke undesirable results if the symbol grounding process fails at categorizing the object, which can be avoided employing *MvSmaps*.

Galindo and Saffiotti [18], envisages an application of semantic maps where they encode information about how things should be, also called norms, allowing the robot to infer deviations from these norms and act accordingly. The typical norm example is that "towels must be in bathrooms", so if a towel is detected, for example, on the floor of the living room, a plan is generated to bring it to the bathroom. This approach works with crispy information, *e.g.* an object is a towel or not. Instead, the consideration of a *MvSmap* would permit the robot to behave more coherently, for example

gathering additional information if the belief of an object symbol being grounded to `Towel` is 0.55 while to `Carpet` is 0.45. In this example, a crispy approach could end up with a carpet in our bathroom, or a towel in our living room. The scenarios illustrated in this section compound a – non exhaustive – set of applications where *MvSmaps* clearly enhance the performance of traditional semantic maps.

7 Conclusions and Future Work

In this work we have presented a solution for tackling the symbol grounding problem in semantic maps from a probabilistic stance, which has been integrated into a novel environment representation coined Multiversal Semantic Map (*MvSmap*). Our approach employs Conditional Random Fields (CRFs) for performing symbol grounding, which permits the exploitation of contextual relations among object and room symbols, also dealing with the uncertainty inherent to the grounding process. The uncertainties concerning the grounded symbols, yielded by probabilistic inference methods over those CRFs, allow the robot to consider diverse interpretations of the spatial elements in the workspace. These interpretations are called universes, which are encoded as instances of the codified ontology with symbols grounded to different concepts, and annotated with their probability of being the right one. Thereby, the proposed *MvSmap* represents the robot environment through a hierarchy of spatial elements, as well as a hierarchy of concepts, in the form of an ontology, which is instantiated according to the considered universes. This paper also describes the processes involved in the building of *MvSmaps* for a given workspace. We have also proposed an strategy for tackling the exponential growing of the multiverse size in complex environments, and analyzed some of the applications where *MvSmaps* can be used to enhance the performance of traditional semantic maps.

The suitability of the proposed probabilistic symbol grounding has been assessed with the challenging Robot@Home dataset. The reported success without considering contextual relations were of $\sim 73.5\%$ and $\sim 57.5\%$ while grounding object and room symbols respectively, while including them these figures increased up to $\sim 81.5\%$ and 91.5%. It has been also shown the building of *MvSmaps* according to the information gathered by a mobile robot in two scenarios with different complexity.

Typically, the semantic knowledge encoded in a semantic map is considered as written in stone, *i.e.* it is defined at the laboratory and does not change during the robot operation. We are studying how to modify this knowledge according to the peculiarities of a given domain, also in combination with a CRF [24]. We think that this line of research is interesting since it would permit the robot, for example, to consider new object or room types not previously introduced, or to modify the properties and relations of those already defined. Additionally, we plan to progressively exploit the presented *MvSmaps* for the applications analyzed in this paper and/or other of interest.

Acknowledgements

This work is supported by the research projects TEP2012-530 and DPI2014-55826-R, funded by the Andalusia Regional Government and the Spanish Government, respectively, both financed by European Regional Development's funds (FEDER).

Appendix A: Performance metrics

The *precision* metric for a given type of object/room l_i reports the percentage of elements recognized as belonging to l_i that really belong to that type. Let $recognized(l_i)$ be the set of objects/rooms recognized as belonging to the type l_i, $gt(l_i)$ the set of elements of that type in the ground-truth, and $|\cdot|$ the cardinality of a set, then the *precision* of the classifier for the type l_i is defined as:

$$precision(l_i) = \frac{|recognized(l_i) \bigcap gt(l_i)|}{|recognized(l_i)|} \tag{8}$$

In its turn, the *recall* for a class l_i expresses the percentage of the spatial elements that belonging to l_i in the ground-truth are recognized as members of that type:

$$recall(l_i) = \frac{|recognized(l_i) \bigcap gt(l_i)|}{|gt(l_i)|}. \tag{9}$$

Precision and *recall* are metrics associated to a single type. To report more general results, we are interested in the performance of the proposed methods for all the considered types. This can be measured by adding the so-called macro/micro concepts. *Macro precision/recall* represents the average value of the precision/recall for a number of types, defined in the following way:

$$macro\_precision = \frac{\sum_{i \in L} precision(l_i)}{|L|} \tag{10}$$

$$macro\_recall = \frac{\sum_{i \in L} recall(l_i)}{|L|} \tag{11}$$

being L the set of considered objects/rooms. Finally, *micro precision/recall* represents the percentage of elements in the dataset that are correctly recognized with independence of their belonging type, that is:

$$micro\_precision(l_i) = \frac{\sum_{i \in L} |recognized(l_i) \bigcap gt(l_i)|}{\sum_{i \in L} |recognized(l_i)|} \tag{12}$$

$$micro\_recall(l_i) = \frac{\sum_{i \in L} |recognized(l_i) \bigcap gt(l_i)|}{\sum_{i \in L} |gt(l_i)|} \tag{13}$$

Since we assume that the spatial elements belong to a unique class, then $\sum_{i \in L} |gt(l_i)| = \sum_{i \in L} |recognized(l_i)|$, and consequently the computation of both micro precision/recall metrics gives the same value.

References

[1] A. Elfes, Sonar-based real-world mapping and navigation, IEEE Journal on Robotics and Automation 3 (3) (1987) 249–265.

[2] S. Thrun, Learning occupancy grid maps with forward sensor models, Autonomous Robots 15 (2) (2003) 111–127.

[3] E. Remolina, B. Kuipers, Towards a general theory of topological maps, Artificial Intelligence 152 (1) (2004) 47–104.

[4] A. Ranganathan, E. Menegatti, F. Dellaert, Bayesian inference in the space of topological maps, IEEE Transactions on Robotics 22 (1) (2006) 92–107.

[5] S. Thrun, Learning metric-topological maps for indoor mobile robot navigation, Artificial Intelligence 99 (1) (1998) 21 – 71.

[6] J. Blanco, J. González, J.-A. Fernández-Madrigal, Subjective local maps for hybrid metric-topological {SLAM}, Robotics and Autonomous Systems 57 (1) (2009) 64 – 74.

[7] S. Thrun, W. Burgard, D. Fox, Probabilistic Robotics, Intelligent robotics and autonomous agents, MIT Press, 2005.

[8] A. Ranganathan, F. Dellaert, Semantic modeling of places using objects, in: Robotics: Science and Systems Conference III (RSS), MIT Press, 2007.

[9] S. Ekvall, D. Kragic, P. Jensfelt, Object detection and mapping for service robot tasks, Robotica 25 (2) (2007) 175–187.

[10] D. Meger, P.-E. Forssén, K. Lai, S. Helmer, S. McCann, T. Southey, M. Baumann, J. J. Little, D. G. Lowe, Curious george: An attentive semantic robot, Robots and Autonomous Systems 56 (6) (2008) 503–511.

[11] R. B. Rusu, Z. C. Marton, N. Blodow, M. Dolha, M. Beetz, Towards 3d point cloud based object maps for household environments, Robotics and Autonomous Systems 56 (11) (2008) 927 – 941, semantic Knowledge in Robotics.

[12] J. McCormac, A. Handa, A. Davison, S. Leutenegger, Semanticfusion: Dense 3d semantic mapping with convolutional neural networks, arXiv preprint arXiv:1609.05130.

[13] S. Harnad, The symbol grounding problem, Phys. D 42 (1-3) (1990) 335–346.

[14] S. Coradeschi, A. Saffiotti, An introduction to the anchoring problem, Robotics and Autonomous Systems 43 (2-3) (2003) 85–96.

[15] S. Coradeschi, A. Loutfi, B. Wrede, A short review of symbol grounding in robotic and intelligent systems, KI - Künstliche Intelligenz 27 (2) (2013) 129–136.

[16] A. Pronobis, P. Jensfelt, Large-scale semantic mapping and reasoning with heterogeneous modalities, in: Robotics and Automation (ICRA), 2012 IEEE International Conference on, 2012, pp. 3515–3522.

[17] B. Mutlu, N. Roy, S. Šabanović, Cognitive Human–Robot Interaction, Springer International Publishing, Cham, 2016, pp. 1907–1934.

[18] C. Galindo, A. Saffiotti, Inferring robot goals from violations of semantic knowledge, Robotics and Autonomous Systems 61 (10) (2013) 1131–1143.

[19] C. Galindo, J. Fernandez-Madrigal, J. Gonzalez, A. Saffiotti, Robot task planning using semantic maps, Robotics and Autonomous Systems 56 (11) (2008) 955–966.

[20] H. Zender, O. M. Mozos, P. Jensfelt, G.-J. Kruijff, W. Burgard, Conceptual spatial representations for indoor mobile robots, Robotics and Autonomous Systems 56 (6) (2008) 493 – 502, from Sensors to Human Spatial Concepts.

[21] D. Koller, N. Friedman, Probabilistic Graphical Models: Principles and Techniques, MIT Press, 2009.

[22] J. Ruiz-Sarmiento, C. Galindo, J. González-Jiménez, Mobile robot object recognition through the synergy of probabilistic graphical models and semantic knowledge, in: European Conf. on Artificial Intelligence. Workshop on Cognitive Robotics, 2014.

[23] J. R. Ruiz-Sarmiento, C. Galindo, J. González-Jiménez, A survey on learning approaches for undirected graphical models. Application to scene object recognition, International Journal of Approximate Reasoning (accepted), 2016.

[24] J. R. Ruiz-Sarmiento, C. Galindo, J. González-Jiménez, Probability and common-sense: Tandem towards robust robotic object recognition in ambient assisted living, 10th International Conference on Ubiquitous Computing and Ambient Intelligence, 2016.

[25] C. Galindo, A. Saffiotti, S. Coradeschi, P. Buschka, J. A. Fernandez-Madrigal, J. Gonzalez, Multi-hierarchical semantic maps for mobile robotics, in: 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2005, pp. 2278–2283.

[26] M. Uschold, M. Gruninger, Ontologies: principles, methods and applications, The Knowledge Engineering Review 11 (1996) 93–136.

[27] L. Karlsson, Conditional progressive planning under uncertainty, in: Proceedings of the 17th International Joint Conference on Artificial Intelligence - Volume 1, IJCAI'01, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2001, pp. 431–436.

[28] M. P. R. S. Al-Moadhen, Ahmed Abdulhadi, R. Qiu, Robot task planning in deterministic and probabilistic conditions using semantic knowledge base, International Journal of Knowledge and Systems Science (IJKSS) 7 (1) (2016) 56–77.

[29] J. R. Ruiz-Sarmiento, C. Galindo, J. González-Jiménez, Robot@home, a robotic dataset for semantic mapping of home environments, Submitted.

[30] R. Marfil, L. J. Manso, J. P. Bandera, A. Romero-Garcés, A. Bandera, P. Bustos, L. V. Calderita, J. C. González, Á. García-Olaya, R. Fuentetaja, et al., Percepts symbols or action symbols? generalizing how all modules interact within a software architecture for cognitive robotics, in: 17th Workshop of Physical Agents (WAF), 2016, pp. 9–16.

[31] P. Viola, M. Jones, Rapid object detection using a boosted cascade of simple features, in: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001), Vol. 1, 2001, pp. 511–518.

[32] D. G. Lowe, Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision 60 (2) (2004) 91–110.

[33] J. Knopp, M. Prasad, G. Willems, R. Timofte, L. Van Gool, Hough transform and 3d surf for robust three dimensional classification, in: Proceedings of the 11th European Conference on Computer Vision: Part VI, ECCV'10, Springer-Verlag, Berlin, Heidelberg, 2010, pp. 589–602.

[34] M. Pontil, A. Verri, Support vector machines for 3d object recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 20 (6) (1998) 637–646.

[35] D. Nister, H. Stewenius, Scalable recognition with a vocabulary tree, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol. 2, 2006, pp. 2161–2168.

[36] W. L. Hoo, C. H. Lim, C. S. Chan, Keybook: Unbias object recognition using keywords, Expert Systems with Applications 42 (8) (2015) 3991 – 3999.

[37] J. Zhang, M. Marszalek, S. Lazebnik, C. Schmid, Local features and kernels for classification of texture and object categories: A comprehensive study, in: 2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06), 2006, pp. 13–13.

[38] O. Mozos, C. Stachniss, W. Burgard, Supervised learning of places from range data using adaboost, in: Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on, 2005, pp. 1730–1735.

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

[40] H. Andreasson, A. Treptow, T. Duckett, Localization for mobile robots using panoramic vision, local features and particle filter, in: Proceedings of the 2005 IEEE International Conference on Robotics and Automation, 2005, pp. 3348–3353.

[41] A. C. Murillo, J. J. Guerrero, C. Sagues, Surf features for efficient robot localization with omnidirectional images, in: Proceedings 2007 IEEE International Conference on Robotics and Automation, 2007, pp. 3901–3907.

[42] C. Weiss, H. Tamimi, A. Masselli, A. Zell, A hybrid approach for vision-based outdoor robot localization using global and local image features, in: 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2007, pp. 1047–1052.

[43] A. Pronobis, B. Caputo, Confidence-based cue integration for visual place recognition, in: 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2007, pp. 2394–2401.

[44] A. Oliva, A. Torralba, The role of context in object recognition, Trends in Cognitive Sciences 11 (12) (2007) 520–527.

[45] S. Divvala, D. Hoiem, J. Hays, A. Efros, M. Hebert, An empirical study of context in object detection, in: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, 2009, pp. 1271–1278.

[46] C. Galleguillos, S. Belongie, Context based object categorization: A critical survey, Computer Vision and Image Understanding 114 (6) (2010) 712–722.

[47] J. Ruiz-Sarmiento, C. Galindo, J. González-Jiménez, UPGMpp: a Software Library for Contextual Object Recognition, in: 3rd. Workshop on Recognition and Action for Scene Understanding, 2015.

[48] A. Anand, H. S. Koppula, T. Joachims, A. Saxena, Contextually guided semantic labeling and search for three-dimensional point clouds, In the International Journal of Robotics Research 32 (1) (2013) 19–34.

[49] J. Valentin, S. Sengupta, J. Warrell, A. Shahrokni, P. Torr, Mesh based semantic modelling for indoor and outdoor scenes, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2013), 2013, pp. 2067–2074.

[50] X. Xiong, D. Huber, Using context to create semantic 3d models of indoor environments, in: In Proceedings of the British Machine Vision Conference (BMVC 2010), 2010, pp. 45.1–11.

[51] J. R. Ruiz-Sarmiento, C. Galindo, J. González-Jiménez, Exploiting semantic knowledge for robot object recognition, Knowledge-Based Systems 86 (2015) 131–142.

[52] J. R. Ruiz-Sarmiento, C. Galindo, J. González-Jiménez, Scene object recognition for mobile robots through semantic knowledge and probabilistic graphical models, Expert Systems with Applications 42 (22) (2015) 8805–8816.

[53] J. G. Rogers, H. I. Christensen, A conditional random field model for place and object classification, in: Robotics and Automation (ICRA), 2012 IEEE International Conference on, 2012, pp. 1766–1772.

[54] D. Lin, S. Fidler, R. Urtasun, Holistic scene understanding for 3d object detection with rgbd cameras, IEEE International Conference on Computer Vision 0 (2013) 1417–1424.

[55] J. R. Ruiz-Sarmiento, C. Galindo, J. González-Jiménez, Joint categorization of objects and rooms for mobile robots, in: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2015.

[56] K. P. Murphy, Y. Weiss, M. I. Jordan, Loopy belief propagation for approximate inference: An empirical study, in: Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence, UAI'99, 1999, pp. 467–475.

[57] J. S. Yedidia, W. T. Freeman, Y. Weiss, Generalized Belief Propagation, in: Advances Neural Information Processing Systems, Vol. 13, 2001, pp. 689–695.

[58] A. Nüchter, J. Hertzberg, Towards semantic maps for mobile robots, Robots and Autonomous Systems 56 (11) (2008) 915–926.

[59] E. Prestes, J. L. Carbonera, S. R. Fiorini, V. A. M. Jorge, M. Abel, R. Madhavan, A. Locoro, P. Goncalves, M. E. Barreto, M. Habib, A. Chibani, S. Gérard, Y. Amirat, C. Schlenoff, Towards a core ontology for robotics and automation, Robotics and Autonomous Systems 61 (11) (2013) 1193 – 1204, ubiquitous Robotics.

[60] M. Tenorth, L. Kunze, D. Jain, M. Beetz, Knowrob-map - knowledge-linked semantic object maps, in: 2010 10th IEEE-RAS International Conference on Humanoid Robots, 2010, pp. 430–435.

[61] D. Pangercic, B. Pitzer, M. Tenorth, M. Beetz, Semantic object maps for robotic housework - representation, acquisition and use, in: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2012, pp. 4644–4651.

[62] L. Riazuelo, M. Tenorth, D. D. Marco, M. Salas, D. Gálvez-López, L. Mösenlechner, L. Kunze, M. Beetz, J. D. Tardós, L. Montano, J. M. M. Montiel, Roboearth semantic mapping: A cloud enabled knowledge-based approach, IEEE Transactions on Automation Science and Engineering 12 (2) (2015) 432–443.

[63] J. O. Reinaldo, R. S. Maia, A. A. Souza, Adaptive navigation for mobile robots with object recognition and ontologies, in: 2015 Brazilian Conference on Intelligent Systems (BRACIS), 2015, pp. 210–215.

[64] M. Günther, T. Wiemann, S. Albrecht, J. Hertzberg, Building semantic object maps from sparse and noisy 3d data, in: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2013), 2013, pp. 2228–2233.

[65] E. Bastianelli, D. D. Bloisi, R. Capobianco, F. Cossu, G. Gemignani, L. Iocchi, D. Nardi, On-line semantic mapping, in: Advanced Robotics (ICAR), 2013 16th International Conference on, 2013, pp. 1–6.

[66] G. Gemignani, D. Nardi, D. D. Bloisi, R. Capobianco, L. Iocchi, Interactive semantic mapping: Experimental evaluation, in: A. M. Hsieh, O. Khatib, V. Kumar (Eds.), Experimental Robotics: The 14th International Symposium on Experimental Robotics, Vol. 109 of Springer Tracts in Advanced Robotics, Springer International Publishing, 2016, pp. 339–355.

[67] I. Kostavelis, A. Gasteratos, Semantic mapping for mobile robotics tasks: A survey, Robotics and Autonomous Systems 66 (2015) 86–103.

[68] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, P. F. Patel-Schneider (Eds.), The Description Logic Handbook: Theory, Implementation, and Applications, Cambridge University Press, New York, NY, USA, 2007.

[69] K. Zhou, M. Zillich, H. Zender, M. Vincze, Web mining driven object locality knowledge acquisition for efficient robot behavior, in: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2012), 2012, pp. 3962–3969.

[70] R. Speer, C. Havasi, Conceptnet 5: a large semantic network for relational knowledge, in: The People's Web Meets NLP. Theory and Applications of Natural Language, Springer, 2013, pp. 161—176.

[71] R. Gupta, M. J. Kochenderfer, Common sense data acquisition for indoor mobile robots, in: Proceedings of the 19th National Conference on Artifical Intelligence, AAAI'04, AAAI Press, 2004, pp. 605–610.

[72] R. Fergus, L. Fei-Fei, P. Perona, A. Zisserman, Learning object categories from google's image search, in: IEEE International Conference on Computer Vision (ICCV 2005), Vol. 2, 2005, pp. 1816–1823 Vol. 2.

[73] J. Philbin, O. Chum, M. Isard, J. Sivic, A. Zisserman, Object retrieval with large vocabularies and fast spatial matching, in: 2007 IEEE Conference on Computer Vision and Pattern Recognition, 2007, pp. 1–8.

[74] S. Thrun, Robotic mapping: A survey, in: Exploring Artificial Intelligence in the New Millennium, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2003, pp. 1–35.

[75] F. Lu, E. Milios, Globally consistent range scan alignment for environment mapping, Autonomous Robots 4 (4) (1997) 333–349.

[76] J. J. Leonard, H. F. Durrant-Whyte, Mobile robot localization by tracking geometric beacons, IEEE Transactions on Robotics and Automation 7 (3) (1991) 376–382.

[77] C. Galindo, J. Fernandez-Madrigal, J. Gonzalez, A. Saffiotti, Using semantic information for improving efficiency of robot task planning, in: IEEE International Conference on Robotics and Automation (ICRA), Workshop on Semantic Information in Robotics, Rome, Italy, 2007.

[78] A. Sloman, J. Chappell, The altricial-precocial spectrum for robots, in: International Joint Conference on Artificial Intelligence, Vol. 19, Lawrence Erlbaum Associates LTD, 2005, p. 1187.

[79] E. Sirin, B. Parsia, B. C. Grau, A. Kalyanpur, Y. Katz, Pellet: A practical owl-dl reasoner, Web Semantics: Science, Services and Agents on the World Wide Web 5 (2) (2007) 51–53.

[80] D. Tsarkov, I. Horrocks, FaCT++ Description Logic Reasoner: System Description, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006, pp. 292–297.

[81] V. Haarslev, K. Hidde, R. Möller, M. Wessel, The racerpro knowledge representation and reasoning system, Semantic Web Journal 3 (3) (2012) 267–277.

[82] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. D. Reid, J. J. Leonard, Simultaneous localization and mapping: Present, future, and the robust-perception age, arXiv preprint arXiv:1606.05830.

[83] P. J. Besl, N. D. McKay, A method for registration of 3-d shapes, IEEE Trans. Pattern Anal. Mach. Intell. 14 (2) (1992) 239–256.

[84] J. D. Lafferty, A. McCallum, F. C. N. Pereira, Conditional random fields: Probabilistic models for segmenting and labeling sequence data, in: Proceedings of the Eighteenth International Conference on Machine Learning, ICML '01, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2001, pp. 282–289.

[85] J. Hammersley, P. Clifford, Markov fields on finite graphs and lattices, unpublished manuscript (1971).

[86] M. J. Wainwright, M. I. Jordan, Graphical models, exponential families, and variational inference, Found. Trends Mach. Learn. 1 (1-2) (2008) 1–305.

[87] Y. Weiss, W. T. Freeman, On the optimality of solutions of the max-product belief-propagation algorithm in arbitrary graphs, IEEE Trans. Inf. Theor. 47 (2) (2006) 736–744.

[88] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.