Robotics and Autonomous Systems 159 (2023) 104294

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

A survey of Semantic Reasoning frameworks for robotic systems

Weiyu Liu *,[1], Angel Daruna[1], Maithili Patel[2], Kartik Ramachandruni[2], Sonia Chernova

Institute for Robotics and Intelligent Machines, Georgia Institute of Technology, Atlanta, GA, United States of America

ARTICLE INFO

Article history:
Received 14 April 2020
Received in revised form 3 October 2022
Accepted 13 October 2022
Available online 22 October 2022

Keywords:
Semantic reasoning
Robotics
Knowledge bases

ABSTRACT

Robots are increasingly transitioning from specialized, single-task machines to general-purpose systems that operate in diverse and dynamic environments. To address the challenges associated with operation in real-world domains, robots must effectively generalize knowledge, learn, and be transparent in their decision making. This survey examines *Semantic Reasoning* techniques for robotic systems, which enable robots to encode and use semantic knowledge, including concepts, facts, ideas, and beliefs about the world. Continually perceiving, understanding, and generalizing semantic knowledge allows a robot to identify the meaningful patterns shared between problems and environments, and therefore more effectively perform a wide range of real-world tasks. We identify the three common components that make up a computational *Semantic Reasoning Framework*: knowledge sources, computational frameworks, and world representations. We analyze the existing implementations and the key characteristics of these components, highlight the many interactions that occur between them, and examine their integration for solving robotic tasks related to five aspects of the world, including objects, spaces, agents, tasks, and actions. By analyzing the computational formulation and underlying mechanisms of existing methods, we provide a unified view of the wide range of semantic reasoning techniques and identify open areas for future research.

1. Introduction

Robots are increasingly transitioning from specialized, single-task machines to general-purpose systems that operate in diverse and dynamic environments. To address the challenges associated with operation in real-world domains, robots must effectively generalize knowledge, learn, and be transparent in their decision making. This survey examines *Semantic Reasoning (SR)* techniques for robotic systems, which enable robots to encode and use semantic knowledge, including concepts, facts, ideas, and beliefs about the world. Just as recognizing linguistic semantics helps a human interpret language, continually perceiving, understanding, and generalizing semantic knowledge allows a robot to identify the meaningful patterns shared between problems and environments, and therefore more effectively perform a wide range of real-world tasks. SR leverages semantic knowledge as an *abstraction* to connect previous experience with new situations, as a *structured prior* to guide robots to efficiently explore new environments and problem domains, and as a *common language* to exchange motives and rationales with humans.

In this survey, we identify the three common components (Fig. 1) that make up a computational *Semantic Reasoning Framework (SRF)*:

- *knowledge sources* from which raw data is received and semantic knowledge can be extracted,
- *computational frameworks* that define mathematical relationships relating known concepts, and that are used to perform inference (e.g., Bayesian networks), and
- *world representations* that enable the robot to model its environment (objects, spaces, and agents) and behaviors (actions and tasks).

Though semantic reasoning has been applied to a wide range of robotics problems, to our knowledge there exists no established structure for concretely placing work within the broader field. Even approaches solving similar problems often make different assumptions, and their relations to the rest of the field remain largely unaddressed. A categorical structure therefore aids in comparative assessments among applications, as well as in identifying open areas for future research. In contributing our categorization of current approaches, we aim to lay the foundations for such a structure.

The remainder of this section formally discusses the space of SR problems and places this survey in the context of existing research. Section 2 presents the key design decisions for developing a semantic reasoning framework. Sections 3, 4, and 5 each cover

* Corresponding author.
E-mail addresses: wliu88@gatech.edu (W. Liu), adaruna3@gatech.edu (A. Daruna), maithili@gatech.edu (M. Patel), kvr6@gatech.edu (K. Ramachandruni), chernova@gatech.edu (S. Chernova).

[1] These two authors contributed equally to the work as first authors.

[2] These two authors contributed equally as the second authors.

https://doi.org/10.1016/j.robot.2022.104294
0921-8890/

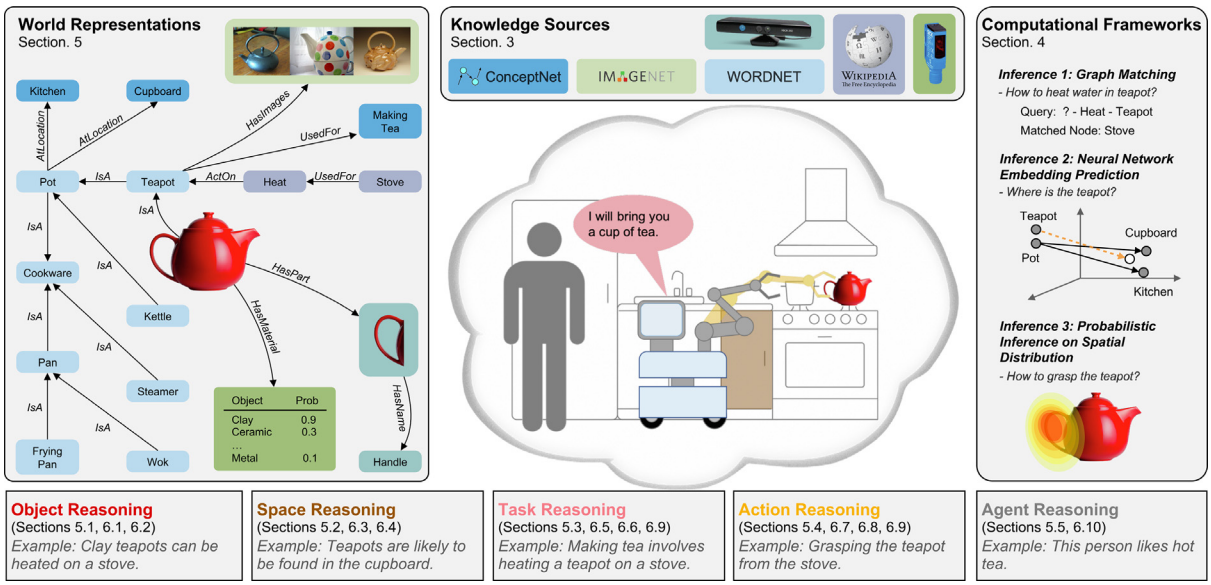


Fig. 1. Three core components of semantic reasoning frameworks and five aspects of the world to which semantic reasoning can be applied.

one of the three core SRF components listed above. Section 6 follows by analyzing SR applications enabled by the integration of SRF components. To conclude, we discuss open areas of research for future work in Section 7.

1.1. Problem domain

The space of problems in semantic reasoning for robotics is large and growing. In this survey, we divide the problem domain into categories related to five entity types: *objects, spaces, agents, tasks,* and *actions*. Semantic reasoning about objects, spaces, and agents allows a robot to better understand its environment, while reasoning about tasks and actions aids a robot in achieving more context-aware and robust operation in complex environments. The division of the problem domain into these five categories, along with the three core components our survey is organized around, allows prior systems and problems to be fully categorized and compared.

Additionally, each of the five categories is inherently different in the modalities and underlying structures of its related data. For example, objects can be modeled as independent entities linked by common attributes and affordances. Tasks, on the other hand, require sequential or hierarchical structures to connect causes and effects. However, the distinctions between these problem categories do not prevent them from overlaps and interactions. In fact, one of the strengths of a multi-modal, multi-relational, and multi-domain robot SRF is to enable all concepts to interact and contribute to decision making.

1.2. Survey scope and context

This survey introduces the audience to the general subject matter of semantic reasoning within robotics, organizes the technical contributions in space of semantic reasoning for robotics, and gives insight into design decisions along with the implications that should be taken into account for implementations on robots.

Within the published literature, reasoning about world knowledge has been studied in a variety of distinct areas, including knowledge representation and reasoning, commonsense reasoning, ontology-based reasoning, contextual reasoning, knowledge-enabled robotics, and cognition-enabled robotics. Some of these areas, such as cognition-enabled robotics, draw the analogy with the human mind, while others, such as ontology-based reasoning, focus on implementation specifics. Due to the different emphases of these areas, the approaches proposed within them intersect but do not completely overlap.

We refer to the general category of algorithms that reasons about world knowledge as *semantic reasoning*. As such, we include methods that obtain semantic information from knowledge sources, internally represent the information in a structured manner, and reason over the representations using computational frameworks towards robotics applications. Regarding our chosen organization, we note that many legitimate criteria could be used to subdivide SR research. Our review aims to analyze the fundamental mechanisms. We therefore categorize existing approaches according to the techniques required to develop a semantic reasoning framework in robotic applications. By analyzing the computational formulation and underlying mechanisms of existing methods, we aim to provide a united view of the wide range of techniques.

Interested readers may also find useful other surveys related to semantic reasoning for robotics. A broad overview of the cognitive skills required for robot manipulation in the human environment is presented in [1]. The work discusses methods for tackling different subproblems, such as learning affordances and human–robot collaboration. A more focused discussion of knowledge representations for service robotics is presented in [2], covering both comprehensive and specific knowledge representations. Another survey, [3], highlights algorithmic details for using logic, probabilistic, and planning-based techniques to reason about robot-related concepts, such as time, space, and action plans. Using declarative knowledge for sequential decision-making under uncertainty is reviewed in [4]. Additional reviews related to sub-areas of SR research are highlighted in respective sections throughout the article. Our work differs from the above publications in that it seeks to present a broad perspective on semantic reasoning for robotics, highlighting the many dependencies, inter-relations, and interactions that occur between the core components of a SRF.

2. Design choices

A developer faces many design choices when constructing a SRF. Some of these decisions may depend on the problem, while

others are up to the preference of the developer. As we discuss in later sections, these design choices strongly influence how the reasoning task is structured and solved. In this section, we highlight several key decisions with a running example in which a robot is tasked to make a cup of tea.

As design choices exist in all three core components of a semantic reasoning framework – knowledge source, computational framework, and world representation – we organize this section accordingly. Despite the order presented here, the designing process is flexible and can vary case-by-case.

Knowledge Source: Developers must consider which types of knowledge are available to the robot, and how they can be represented. Knowledge sources used to seed semantic reasoning frameworks can be categorized as containing *class-level knowledge* that generalizes across an entire class of entities (e.g., "`cups are often found in cupboards`") or *instance-level knowledge* that pertains to a specific instance of an entity (e.g.,"`blueCup2 is on the table`"). Other factors that influence knowledge acquisition include data quality, recoded data modalities, and data representation structure.

Computational Framework: When selecting a computational framework to store semantic knowledge and support inference, a developer must consider five essential characteristics: the need to *model uncertainty*, *expressiveness*, *adaptability*, *explainability*, and *scalability*. The selection of a computational framework further depends on the requirements from other design choices and their associated effects, such as the scale of the problem and the noise level in the selected knowledge sources.

World Representation: A semantic reasoning framework must capture details about the world that are relevant to the reasoning capabilities and objectives of the system. Most existing framework model some subset of *objects, spaces, tasks, actions*, and *agents*. For example, for finding a teacup, we may model space semantics, user preferences, or even object affordances, to help determine likely object locations. Semantic knowledge is most often represented as symbols. Symbols are discrete and abstract, therefore reduce the dimension of semantic space and facilitate generalization. Within our example, the action of a robot grasping and lifting a teacup can be abstractly encoded as *pickup cup*. Alternately, non-symbolic representations can be used, such as a point cloud representation of the object. When designing a semantic reasoning framework, whether to use symbols depends on many factors such as interpretability, the desired level of abstraction, and the complexity of the data.

Developers must carefully consider each of the core components for semantic reasoning when designing a complete system. Furthermore, interactions between choices made in each component must be compatible. The following sections describe existing implementations of each core component and discuss the synergy of these components to achieve semantic reasoning.

3. Knowledge sources

Observations of the world through onboard sensors provide the most direct and easily accessible information to a robot. Prior to being incorporated into a semantic reasoning framework, raw sensor readings are typically abstracted into more compact representations [5–7]. However, learning all knowledge from scratch only through local observations is inefficient, particularly for robots that must perform multiple tasks or operate in multiple environments. Thus, it is useful for robots to have access to other, more general, sources of data to supplement knowledge obtained from local observations.

In this survey, we define the most important characteristic of a data source as whether it contains class-level or instance-level knowledge. *Instance-level knowledge* grounds concepts physically. This type of knowledge can be discovered from robots' interactions with the world and therefore can be verified by the robots. *Class-level knowledge* facilitates generalization across domains and provides a prior for new situations. This type of knowledge can be directly encoded by human users or generalized from instance-level knowledge. Ultimately, a robust semantic reasoning framework must have the ability to reason about both types of information, as well as to perform information exchange across these complementary and interdependent data types. In section, we discuss data sources – knowledge bases, datasets, and ontologies – of both types, as well as the interaction between them.

3.1. Class-level knowledge

Class-level knowledge is often symbolic and represents information that is asserted across domains, or that generalizes across an entire class of entities. Class-level knowledge is available in various sources, such as encyclopedias, formalized knowledge bases, and specialized semantic networks. Respectively, these sources provide different types of information, such as summaries of concepts from different disciplines, common-sense knowledge, and domain-specific knowledge pertaining to robots. Table 1 shows a representative list of class-level knowledge sources that have been utilized in prior work to seed semantic reasoning frameworks with information. We provide a detailed description for each knowledge source, emphasizing its content and intended use. In addition to the types of information, each of the knowledge sources can further be characterized by two factors that affect the design of semantic reasoning frameworks, the *structure* and *noise level* of the data, as shown in Table 2.

3.1.1. Structure of class-level knowledge

The structure of the data encoded in a knowledge base determines what relations can be stored and what inferences can be made from the data. Hierarchical data structures, such as WordNet and OpenCyc, encode super-/sub-class relations (e.g., *isA (apple, fruit)*), thereby facilitating generalization. Graph-based data structures, such as ConceptNet, encode the data in a flat, highly inter-connected representation (e.g., *hasProperty(apple, green)* and *atLocation(apple, table)*), which is more commonly used to represent highly varied or probabilistic data. Ordered data structures organize data according to certain metrics; for example, WikiHow data is ordered chronologically. Finally, unstructured data, such as that found in Wikipedia, contains statements and facts that must first be extracted or parsed before they can be applied to robotic systems. Many knowledge sources that aim to capture a wide variety of information have multiple structures; for example, ConceptNet has both class hierarchies and multi-relational information. The structure of the knowledge source is an important consideration in the design of semantic reasoning frameworks because the structure impacts how knowledge can be applied. For example, the step-by-step instructions in WikiHow naturally map to the sequential arrangement of actions in a task, whereas the hierarchical data encoded within WordNet can be used to better understand the objects the robot is dealing with in the world.

3.1.2. Imperfect class-level knowledge

Class-level knowledge is imperfect as it often has noisy and missing information. The noise of the data determines how much the information can be trusted, and therefore influences the choice of world model and computational framework used within a semantic reasoning framework. Inaccurate information may result from knowledge sources being crowdsourced or automatically mined; many of the existing data sources have avoided

Table 1
Class-level knowledge sources.

AfNet [8]	An ontology of affordances for over 250 commonly found object classes. AfNet describes unique geometric mappings for each affordance (e.g., *Contain-ability* is defined by high convexity). AfNet also includes definition of object classes in terms of affordances and topological structures of components. *Sample use cases:* [9,10]
ConceptNet [11]	A large-scale knowledge graph encoding common-sense knowledge from various sources, including the Open Mind Common Sense (OMCS) project [12], WordNet [13], OpenCyc [14], DBPedia [15], and etc. Data is organized as a weighted graph structure, with edge weights used to convey the estimated reliability of the information. In total, ConceptNet stores over 8 M nodes and 21 M edges, with 1.5 M nodes in English. Example edge relations particularly useful for robotic applications include *IsA*, *AtLocation*, *HasProperty* and *UsedFor*. Since information comes from various sources and are partially mined in an unsupervised way, inaccurate information is common within the dataset. *Sample use cases:* [16–19]
KnowRob Ontology [16,20–22]	A robot-centered ontology built on top of OpenCyc. The ontology includes 8 K classes covering both a broad range of human knowledge and domain-specific knowledge for robots such as everyday tasks, household objects, and robot parts. The most important branches are the *TemporalThings* describing temporal concepts such as events and actions, *Spatialthings* describing spatial concetps such as places, objects, and body parts, and *MathmaticalObjects* describing abstract concepts such as coordinate systems and linear algebra. *Sample use cases:* [16,22–24]
Open Robots Ontology (ORO) [25]	An ontology that focuses on concepts useful for human–robot interaction. Similar to the KnowRob ontology, the Open Robots ontology inherits OpenCyc concepts such as *TemporalThings* and *Spatialthings*. The ontology includes additional agent related concepts like *desiredBy*, *focusedOn*, and *BodyPart*. This ontology is also hand coded and has minimum amount of noise. *Sample use cases:* [25,26]
OpenCyc [14]	A publicly available subset of the Cyc ontology, which contains formalized common sense knowledge suitable for reasoning and problem-solving in a variety of domains. OpenCyc consists of over 40 K terms and over 200 K handcrafted commonsense axioms, including taxonomic information and semantic knowledge (i.e., additional facts and rules of thumb). *Sample use cases:* [16,25,27]
WikiHow and EHow	Public websites containing more than 1 M step-by-step how-to guides, providing natural language instructions for various tasks, including thousands of household tasks for everyday activities. Because these databases are designed for human users, the instructions often assume prior domain knowledge or require common-sense reasoning to complete. *Sample use cases:* [19,28–30]
Wikipedia [31]	A multilingual, web-based, free-content encyclopedia containing over 40 M articles. For robotics applications, it can be used to mine general knowledge about classes of objects. *Sample use cases:* [27]
WordNet [13]	A large lexical database of English. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of semantic and lexical relations, such as *Hyponymy* and *Meronymy*. *Sample use cases:* [16,19,27,32]
YAGO [33]	A large knowledge base automatically extracted from Wikipedia categories, Wikipedia infoboxes, WordNet, and GeoNames. YAGO combines the clean taxonomy of WordNet with the richness of the Wikipedia category system, assigning the entities to more than 350 K classes. The knowledge base contains over 16 M entities anchored in time and space, further connected to each other by 76 hand-defined relations. Due to the selective way information is mined, the data encoded in the dataset is relatively free of noise. *Sample use cases:* [34]

Table 2
Grouping of class-level knowledge sources by data structure and noise.

		Noise level	
		Low	High
Data structure	Hierarchical	KnowRob, ORO, YAGO, OpenCyc, WordNet	
	Graph-based	AfNet, KnowRob, ORO, YAGO, OpenCyc, WordNet	ConceptNet
	Ordered	WikiHow	
	Unstructured	Wikipedia	

this problem through hand-coding by experts (e.g., WordNet) or strong verification policies (e.g., Wikipedia). In general, the amount of noise present in the data affects the choice of computational framework to be used for semantic reasoning, with probabilistic methods being favored when the data is noisier. As another form of imperfect information, missing data is an issue that is harder to avoid and faced by any dataset to some degree. Despite containing millions of facts, large-scale knowledge bases still suffer from this problem [35]. When key concepts are missing from a knowledge source, no evidence is available to support important inferences. For example, when any one of the two concepts: *milk is perishable* and *perishable food is stored in refrigerators* is missing, the robot may be unable to deduce that milk should be stored in a refrigerator. Techniques for combining multiple knowledge sources [36] and explicitly inferring missing information [37] can help mitigate issues regarding incomplete knowledge.

3.2. Instance-level knowledge

Instance-level knowledge refers to knowledge that describes specific instances of a class of entities. To connect individual instances to abstract class-level concepts, instance-level knowledge typically includes a text-based label or description in addition to its raw data. Table 3 presents a list of instance-level knowledge sources that have been utilized in prior work. Unlike in the case of class-level knowledge, there is often more freedom on the structure of the data, with most sources consisting of sets of images or lists of entities. Noise remains a contributing factor, with some sources being more reliable than others. However, the most important characteristics of instance-level knowledge sources are the *modality of the data* and its *generalizability* (Table 4), which we discuss next.

3.2.1. Data modalities of instance-level knowledge

Instance-level knowledge must encode the specific distinguishing characteristics of individual entities, thus relying on richer data modalities, including image, depth, shape, trajectory, and pose information. By contrast, information in class-level knowledge sources is typically encoded in the form of text as commonalities of instances allow for the use of compact and abstract textual representations. For instance-level knowledge, the most commonly used modality is image data, supporting

Table 3
Instance-level knowledge sources.

50 Salad Dataset [38]	A video dataset of people preparing salad recipes, focusing on complex interactions between hands, tools, and manipulable objects (50 video sequences in total). Data includes synchronized RGB-D video and accelerometer data of all the kitchen tools, and annotations for before, during, and after every action in the recipe. *Sample use cases:* [39,40]
AI2Thor [41]	A virtual simulator that includes 30 highly realistic kitchens, bedrooms, bathrooms, and living rooms (120 rooms total) with actionable objects and physics. Simulated actions include *pick up*, *put down* and *open* (e.g., *pick up tomato*, *open microwave*). The simulator provides class names and location information for all objects. *Sample use cases:* [42,43]
Amazon, Ebay, etc.	Consumer product websites that offer a wealth of knowledge specific to each type of product. The data is constantly monitored by manufacturers, customers, and sellers so mistakes are relatively infrequent. Previous works have scraped such websites to obtain item size, weight, description, and image information. *Sample use cases:* [16,32]
BEHAVIOR [44]	A benchmark of 100 diverse household activities for embodied AI, such as *chopping vegetables*, *putting away toys*, and *cleaning bathroom*. Activities are long-horizon and induce object state changes (e.g., *soaking materials* and *cleaning surfaces*). Simulation environment includes 15 models of real-world homes and 1217 object models with in WordNet labels. *Sample use cases:* [44]
COCO [45]	A large-scale object detection, segmentation, and captioning dataset containing over 200 K labeled images of approximately 1.5 M object instances, labeled within 80 unique object categories (e.g., fork, dog) and 91 stuff categories (e.g., sky, grass). COCO also includes 250 K instances of people with labeled keypoints such as the body, limbs, and facial features. *Sample use cases:* [46]
Ego4D [47]	An egocentric video dataset containing 3670 h of daily-life activities such as *watching tv*, *playing cards*, and *walking on the street* collected in diverse scenarios. The dataset contains a variety of modalities such as 3D scans, audio, stereo and gaze. Videos are divided into 5 min intervals and each interval is annotated with short narrations at individual timesteps and a 1–3 sentence summary. *Sample use cases:* [48]
Habitat 2.0 [49]	An interactive physics-enabled virtual simulator with 111 photo-realistic apartment layouts rendered in 3D, each layout consisting of multiple rooms and hundreds of actionable objects. The simulator is high-performance with a simulation speed of 850 times of real-time. Sample tasks include *tidy house* and *set table*. *Sample use cases:* [50]
ImageNet [51]	A large-scale image database that provides an average of over 600 images for the majority of synsets in WordNet, with a total of 3.2 M images. Images of each concept are human-annotated with an average labeling precision of 99.7%. *Sample use cases:* [27,32]
ShapeNet [52]	A large-scale dataset of 3D CAD models of common objects, organized based on the WordNet taxonomy and including semantic annotations such as real-world dimensions and material composition. The dataset contains 3 M 3D shape models, including 220 K categorized shape models into 1 of 3135 categories from WordNet synsets. *Sample use cases:* [53]
Stanford 40 Action Dataset [54]	A medium-scale image dataset labeled with bounding boxes of each person in the image, and the name the action being performed. The dataset contains 9 K images, with 180–300 images per action class. *Sample use cases:* [32]
SUNCG [55]	A virtual simulator with 45 K scenes of realistic room and furniture layouts occupied by objects rendered from meshes. Depth maps and ground truth for volumetric semantic labels are provided with each scene. *Sample use cases:* [56]
ThreeDWorld [57]	A set of virtual 3D environments where all objects respond to physics, and a robot agent can be controlled using a fully physics-driven navigation and interaction API. The simulator contains a total of 15 different environments each with 6 to 8 interconnected rooms populated with objects from 50 different categories. *Sample use cases:* [58]
TaskGrasp [59]	A task-oriented grasping dataset consisting of 250 K stable grasps for 191 household objects (75 object categories) and 56 manipulation tasks, with each object suitable for 7 tasks on average. The task–object pairs are both prototypical (ladle for pouring) and uncommon (tongs for stirring) to impose diverse semantic constraints on grasping. Objects and tasks are also mapped to their WordNet synsets to incorporate semantic knowledge. *Sample use cases:* [59]
Visual Genome [60]	A large-scale image dataset consisting of 108 K images of 75 K unique objects grounded in WordNet. Each image is structured as a scene graph of concepts (e.g, *dog*) and relations (e.g, *in*, *on*). The graph also includes object attributes about abstract concepts (e.g., *shirt is pink*) and region descriptions (e.g., *the woman teaching the little girl to cook*). *Sample use cases:* [46,61]

the common idiom "a picture is worth a thousand words". Frequently, multiple modalities are used to capture the details of a scene or task. For example, works have paired image data with various types of sensor data such as spectrometer data [62], audio feedback [58] and haptic feedback [63] for robot manipulation and navigation. Video datasets annotated with textual captions such as daily activity labels [44] and symbolic state representations [63] are used to train neural models that can reason about semantic knowledge embedded in visual data. Additionally, simulators have been used to simulate the sounds of falling objects [58], generate multiple views of a scene to learn useful scene abstractions [64,65], complete the geometry of segmented objects [66], predict spatial relations between objects [66,67], and learn a co-occurrence probability distribution of objects in environments [68,69]. The symbolic labels associated with sensor data take on many different forms, ranging from text labels of object categories [45,51], to object affordances [8,62], to spatial relations between objects [70,71], to logical predicates indicating world states [63], to referring expressions of 3D objects [72,73], to natural language instructions for manipulation [74–76] and navigation actions [77,78]. Reasoning about multimodal information is nontrivial because making inference across different modalities is challenging. However, multimodal instance-level knowledge is essential to achieve precise grounding of abstract concepts, as well as for robust task execution.

3.2.2. Generalizability of instance-level knowledge

The second critical factor for instance-level data is its degree of generalizability across domains. In Table 4, we distinguish

Table 4
Grouping of instance-level knowledge sources by data modality and domain generalization.

		Domain generalization	
		Single	Multiple
Data modalities	Text, RGB	Amazon/Ebay, ImageNet, Stanford 40 Action	AI2Thor, BEHAVIOR, COCO, Habitat, SUNCG, Visual Genome
	Text, RGBD, & more	50 Salad, ShapeNet	Ego4D, ThreeDWorld, TaskGrasp

datasets that are designed for a single application (e.g., classification) or domain (e.g., kitchens), v.s. multiple applications/domains. Examples of data sources designed to address a single application/domain include the Stanford 40 Action Dataset, which contains images for only 40 actions, and 50 Salads Dataset, which is limited to recipes for making salads. Multipurpose data sources, such as COCO and AI2Thor, contain annotations useful to multiple types of applications (e.g., full object segmentations with labels for object detection and classification) or multiple domains (e.g., images from kitchens, bathrooms, and bedrooms). We are gradually seeing more multi-domain multi-task datasets [44,47] as the field moves towards developing generalist robots.

3.3. Integrating class and instance knowledge

It is critical to highlight the complementary nature of class and instance knowledge. Class-level knowledge captures high level, generalizable patterns in the data, while instance-level knowledge enables those abstract concepts to be grounded to a specific environment, as well as captures the statistical characteristics of the data. When either type of knowledge is used alone in a semantic reasoning framework, undesirable simplifying assumptions often have to be made. When class-level knowledge is used alone, the process to ground abstract concepts, known as the symbol grounding problem [79,80], is usually simplified. For example, grounding has been simplified by manually defining mapping functions [81], using visual cues such as fiducial markers [26], or representing objects as blocks with different colors and shapes [82]. When instance-level knowledge is used alone, general rules in the domain are induced solely from raw data; no prior knowledge is incorporated [62,83,84]. Ultimately, a robust semantic reasoning framework must have the ability to reason about both types of information.

At the data level, the synergy between class and instance knowledge can be facilitated through several instance-level knowledge sources, such as ImageNet, ShapeNet, and Visual Genome, which integrate both class and instance knowledge by organizing instances according to the WordNet hierarchy. At the system level, multiple techniques have been developed for integrating instance and class knowledge [59,85,86], as will be examined more closely in Section 6.4.

3.4. Language as a knowledge source

Language exists as both a class-level and instance-level knowledge source, and numerous semantic reasoning frameworks use language to teach robot agents new semantic concepts and skills. For example, novel perceptual concepts of objects can be learned from human dialogue using an active learning approach [87]. A work uses this active learning algorithm in a conversational agent to update the agent's perceptual concept models on the fly [88]. Natural language interaction can also used to learn and improve an interactive dialogue system composed of a reinforcement learning based dialogue strategy and semantic parser [89]. Cognitive robot architectures such as DIARC [90] and BWIBots [91] provide a robot platform to learn from human interaction by providing dialogue, perception, reasoning, and action execution capabilities. For example, one work extends a cognitive robot architecture to perform one-shot learning of robot actions for multi-agent interactions by combining natural language instruction with human demonstration [92]. Another work uses a cognitive architecture to learn object affordances with socio-contextual dependencies from natural language instructions [93].

In addition to human dialogue, large language models trained on text corpora like [13,31] have been successfully used in robotics applications such as task planning [94,95], visual navigation [50,96], spatial-language grounding [73,97], and tool manipulation [98]. These pre-trained language models can provide knowledge specific to the current context while enabling zero-shot generalization to new scenes and tasks, thereby differentiating from other class-level or instance-level knowledge sources. For instance, pre-trained language models have been used in task planning to pick an available action based on language-similarity [94] or predict the probability of an action being successful [95]. However, since these language models are trained purely on text data, grounding knowledge obtained from language models to the real world remains a challenge. Recent approaches have leveraged multi-modal models such as CLIP [99] trained on paired visual-language data to ground language to vision [74,100] and determine navigation actions for long-horizon tasks [50,96].

4. Computational frameworks

Given data acquired from one or more of the data sources described in the previous section, we now discuss computational frameworks — the organizational structures that enable robots to reason about semantics in the data. A wide range of computational methods has been applied across the literature for knowledge representation and reasoning, with no single approach applicable across all scenarios. Each representation offers different mathematical structures, assumptions, and types of inference. In turn, the combination of these factors leads to different performance characteristics with respect to *modeling uncertainty*, *expressiveness*, *adaptability*, *explainability*, and *scalability*. Each of these characteristics varies in its importance for different applications and use cases. For example, the scalability requirements of a computational framework for a single robot reasoning about semantic grasping [62,101] are different from a framework designed as a shared cloud-based knowledge repository for many robots across different environments [21,27].

In Sections 4.1–4.7 we present a representative list of computational frameworks that have been used in literature. We describe the different data and mathematical structures upon which these frameworks are built, how their assumptions lead to various tradeoffs and use cases in the context of semantic reasoning. Finally, we compare these frameworks in terms of the high-level characteristics that are essential for SRFs in Section 4.8.

4.1. Semantic graphs

A semantic graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is a set of vertices $\mathcal{V}$ connected by directed and/or undirected edges $\mathcal{E}$, represented by triples $\{(v_i, e_j, v_k) | v_i, v_j \in \mathcal{V} \wedge e_j \in \mathcal{E}\}$. In the context of SRF, vertices $\mathcal{V}$ store one or multiple types of entity information (e.g., text, images, sounds, trajectories, and algorithm parameters) and each

edge $e \in \mathcal{E}$ has a predefined edge type that represents the relation between the connected entities. Optional confidence values can also be associated with vertices $\mathcal{V}$ and edges $\mathcal{E}$ in order to represent the certainty of the encoded knowledge. Fig. 2 shows an example semantic graph.

Tradeoffs in Context of SRF: A semantic graph is highly adaptable and expandable. New observations can be easily added to a semantic graph in the form of new vertices or edges. Repeated (or lack of) observations of various phenomena can be modeled by increasing (decreasing) confidence values associated with the relevant entities. Vertices can also be merged or split, as in [27], when new knowledge is acquired (e.g., splitting existing entity *Cup* into *Mug* and *Cup*). However, a significant limitation of the graph representation is that it does not support rigorous probabilistic inferences. Furthermore, the belief of an edge or node is not well established, making it difficult to assess the relative certainty of various types of information. An approach uses the Katz centrality to assign an 'importance' score to nodes corresponding to particular objects or motions [102], while another incorporates beliefs from disparate knowledge sources or algorithms [27].

Uses and Applications: Reasoning over semantic graphs can be performed at the node, local subgraph, or global graph levels. At the node level, node similarity can be computed by comparing the locations of nodes in the semantic graph. For example, Wu-Palmer similarity [103] is used on WordNet [13] data to generalize manipulation sequences [104] and organizational preferences [105] between similar objects. At the subgraph level, information retrieval on large-scale semantic graphs is performed by matching a query template with the graph [27,102,106]. Queries typically take the form (u, e, v), in which the variables u and v are nodes in the semantic graph and the variable e is a directed edge from u to v. For example, the query $(u, \mathit{HasAffordance}, \mathit{scoop})$ can be used to retrieve a list of objects that provide the scooping affordance, and the query $(\mathit{spoon}, e, \mathit{kitchen})$ can be used to identify the relationship between a spoon and a kitchen. More complex reasoning can be achieved by chaining multiple queries together, such as the above examples that can be used to identify that going to the kitchen may allow the robot to find an object for scooping [27]. At the global level, graph matching assesses similarity between different models. For example, graph matching over topological models of human spaces and objects provides a solution for place recognition and place classification [83], and graph matching over object models of constituent parts enables object recognition [107]. A different approach is used in [108], in which a score over an entire object graph is computed based on object properties and neighboring objects. The importance of different features is learned from demonstrations in order to encode trajectory preference.

4.2. Markov networks

A Markov network (MN), or Markov Random Field, is a probabilistic graphical model represented by the pair $(\mathcal{H}, \mathcal{P})$. The joint probability distribution $\mathcal{P}$ factorizes over the undirected graph $\mathcal{H}$, whose nodes represent a set of propositional random variables and edges represent the correlations between random variables, as illustrated in Fig. 3. A commonly used type of Markov network is a Conditional Random Field (CRF), in which random variables are divided into a set of target variables Y and a set of observed variables X. Rather than encoding the joint distribution $P(Y, X)$, a CRF represents the conditional distribution $P(Y|X)$.

Tradeoffs in Context of SRF: As a probabilistic model, a MN provides a computational framework for representing a complete probability distribution — the probability of every possible event as defined by the values of all the random variables. Additionally, the independence assertions encoded in the graphical structure

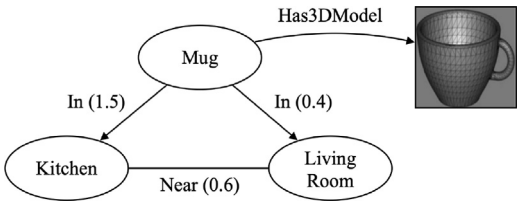


Fig. 2. An example semantic graph that includes multimodal data, different types of relations, and local confidence values defined for edges (shown as values in parentheses). The confidence scale is not bounded.

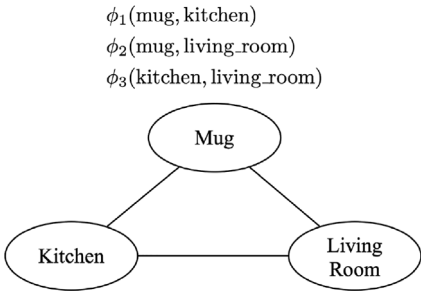


Fig. 3. An example Markov network with undirected edges between random variables. The joint distribution can be factorized into clique potentials, denoted by ϕ's.

allow a distribution to be compactly represented as products of factors, or clique potentials. Since the factorization is over cliques, which are fully connected subsets of the random variables in the graph (e.g., pairs of variables), a Markov network is especially suitable for modeling symmetric or associative relations between variables. When relations between certain variables are hard to elicit due to overlapping information or implicit correlations, a CRF can be used to avoid representing a probabilistic model over these variables. However, the flexibility in defining MNs and CRFs results in a lack of clear semantics, which has several disadvantages. First, each clique contributing to the overall inference result does not help to reveal which random variables affect the result the most. Second, the use of cliques to define a joint distribution makes parameterizing the model by hand more difficult. The second limitation leads to the convention of learning clique potentials from training examples, which requires apriori data to converge to reasonable parameters [109–111].

Uses and Applications: Markov networks have been used to model spatial and contextual relations between objects. Jointly reasoning about these relations helps to improve the robustness of object classification algorithms over those that are based solely on visual features. For example, Relational Markov Networks, an extension of CRFs, have been used to represent the spatial relations between walls and doors in 2D laser scans [110]. Modeling the spatial relations allows this approach to infer labels for line segments that are not confidently classified from the 2D map features alone. Similarly in [111], CRFs are used to exploit contextual and spatial relations between objects in a scene to improve object classification. Particle filter based belief propagation over a CRF has been used to represent belief over various objects in the scene [112].

4.3. Bayesian networks

A Bayesian network (BN) is a probabilistic graphical model represented by the pair $(\mathcal{G}, \mathcal{P})$, where the probability distribution $\mathcal{P}$ factorizes over a directed acyclic graph $\mathcal{G}$. The nodes in

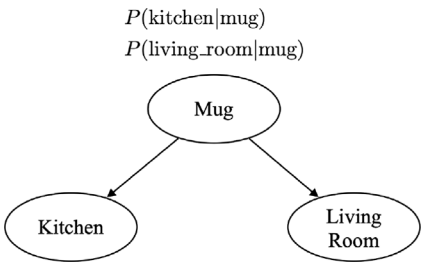


Fig. 4. An example Bayesian network with directed edges between random variables. The joint distribution can be factorized into conditional probabilities, denoted by P's.

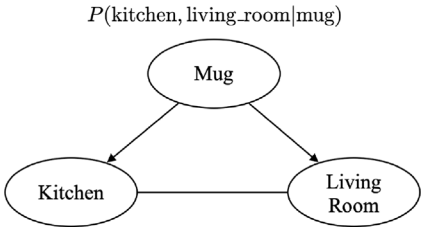


Fig. 5. An example partially directed acyclic graph with both directed and undirected edges between random variables. The edge between the two nodes in the same chain component is undirected, while the edges between two nodes in different chain components are directed. The joint distribution can be factorized into conditional probabilities of chain components given their respective parents.

$\mathcal{G}$ represent propositional random variables, and edges represent informational or casual dependencies between the variables. Fig. 4 shows an example of a BN.

Tradeoffs in Context of SRF: The main advantages of a Bayesian network as a computational framework are its precise probabilistic interpretation and its adaptability. The Markov assumption and directed-acyclic constraints that define the scope of the network, allow for simple interpretation of conditionally independent variables and causal relations between variables. The structure of BNs allows for direct inference over variables and learning of parameters/structures via efficient approximations, such as importance sampling or Gibbs sampling. Factorizing the full joint distribution over $\mathcal{G}$ via conditional probability tables (or distributions) also makes determining the influences of inference results more accessible than MNs. However, managing large-scale conditional probability tables leads to drawbacks in terms of scalability. While the richness of the resulting probabilistic representation is useful to robots reasoning about specific problems, inference and learning in BNs is often intractable for real-world problem sizes involving many random variables and edges in a dense network. Thus, due to limited scalability, BNs are rarely used in complex robot environments, but instead are commonly utilized as a foundation for more complex computational frameworks.

Uses and Applications: BNs have been used for semantic grasping by encoding relations between grasps, object features, and task constraints [101]. In this context, Gaussian Mixture Models are used to discretize continuous data for BNs in order to learn the network structures for factors such as object convexity and grasp location. BNs have also been used to generate situated probabilistic models of the environment, enabling the robot to predict likely locations for previously unseen objects [17]. In [113], BNs are extended to incorporate context and temporal relations into action selection using Dynamic Bayesian Networks (DBNs), a representation that presents a compromise between state and space complexity.

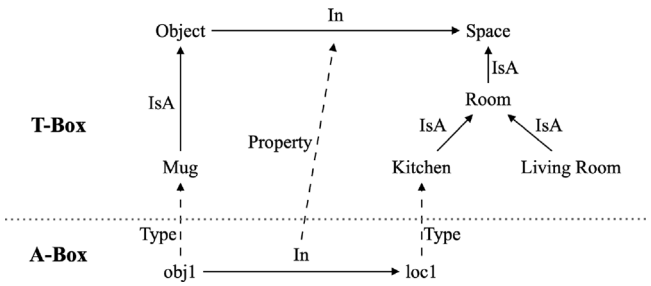


Fig. 6. A graphical representation of an example description logic ontology. The T-Box contains axioms defining relations between class-level concepts. The A-Box contains a single fact, which is governed by the constraints defined in T-Box.

4.4. Partially directed acyclic graphs

A partially directed acyclic graph (PDAG), or Chain Graph, is a graphical model represented by the pair $(\mathcal{I}, \mathcal{P})$, where the probability distribution $\mathcal{P}$ factorizes over the hybrid graph $\mathcal{I}$, which consists of both directed and undirected edges that represent influences between the propositional random variables encoded in the nodes of $\mathcal{I}$ [109]. Fig. 5 shows an example PDAG.

Tradeoffs in Context of SRF: PDAGs, which can be thought of as a combination of MNs and BNs, allow for modeling causal as well as associative relationships. However, similar to MNs, the PDAG joint distributions are defined over chain components of cliques in the moralized graph of $\mathcal{I}$ instead of individual conditional probabilities, as in BNs. Factorization over cliques leads to confounding inference results for reasons similar to that of MNs, namely it is difficult to distinguish the variables that contribute to an inference result. Additionally, PDAGs lack clear semantics for model parameters, which makes model parameters difficult to elicit from experts. As a result, the convention, much like for MNs, is to estimate model parameters from training data [114].

Uses and Applications: PDAGs have been used to perform causal and associative reasoning of spatial commonsense knowledge in order to build spatial models of indoor environments. In [114], each room instance is connected to one another by undirected edges according to a topological map. The potentials on undirected edges are used to describe typical connectivity between room categories. Within each room, the variable representing the room's category is linked via directed edges to the room shape, size, appearance, and objects in it, capturing the causal relations between these attributes and the room type. Similarly in [115], undirected edges are used to model connectivity; however in this case, they connect nodes representing waypoints in the map. For each waypoint, causal relations between viewing angles, expected objects in a view, and observations from a robot are modeled by directed edges.

4.5. Description logics

Description Logics (DLs) are a family of formal knowledge representation languages that are widely used in ontological modeling. DLs represent an application domain using the pair $(\mathcal{T}, \mathcal{A})$, where $\mathcal{T}$, the T-Box, contains terminological axioms describing relationships between concepts, and $\mathcal{A}$, the A-Box, contains assertional axioms capturing knowledge about named individuals, i.e., the concepts to which they belong and how they are related to each other [116]. Fig. 6 shows an example of a DL ontology.

Tradeoffs in Context of SRF: As a logic language, DLs provide a precise specification of the meaning of ontologies. This precise specification allows DL ontologies to be exchanged without ambiguity of their meaning, and also makes it possible to use

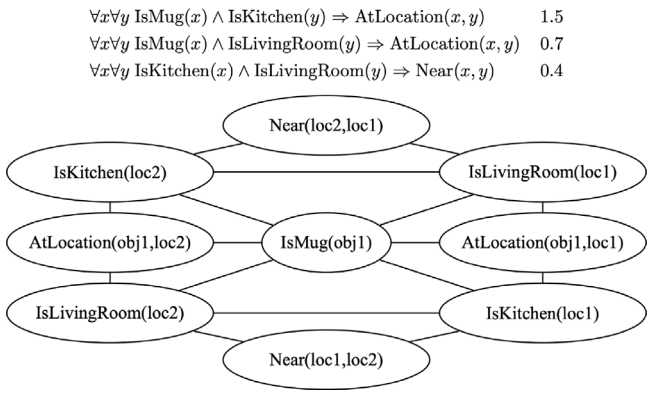


Fig. 7. An example Markov logic network and its grounded Markov network. Each formula defining the Markov logic network has an associated weight that reflects how strong a constraint it is. The Markov network has 3 grounded atom: obj1, loc1, and loc2.

logical deduction to infer additional information from the facts stated explicitly in an ontology. As decidable fragments of first-order logic, many DLs also have effective methods to always derive the correct answer. Due to these benefits, DLs are widely used in ontological modeling, and they provide the formalism for the OWL Web Ontology Language, which is standardized by the World Wide Web Consortium (W3C). As a result of the wide use in the Semantic Web community, there are many mature libraries providing tools to manipulate, reason, and query DL ontologies. However, DL cannot be used to reason about uncertainty because DL offers only deterministic reasoning about logic statements. The lack of uncertainty modeling makes DL brittle when incorporating new knowledge in situations where logical contradictions arise [25]. However, ignoring uncertainty allows DL-based semantic reasoning frameworks to scale to a greater number of instances, type, and predicates [20]. Therefore, DLs are typically used for providing robots with large-scale knowledge bases containing concepts from various domains.

Uses and Applications: Large-scale DLs have been used to encode contexts, spaces, objects, actions, and features, along with axioms that express inter- and intra-group relations [117–119]. Custom DLs have also been used to formalize knowledge in specific domains, including industrial robotics [120], swarm robotics [121], and object manipulation [122]. Instead of designing the whole ontology from scratch, the KnowRob ontology is constructed by combining a manually designed ontology with the public OpenCyc ontology, therefore bootstrapping available declarative knowledge with general knowledge that could be leveraged during tasks [81]. The ORO ontology is created in a similar fashion by integrating with the OpenCyc ontology [25]. However, the ORO ontology focuses on human–robot interaction, therefore adding new concepts designed to facilitate interaction.

4.6. First-order probabilistic models

First-order Probabilistic Models are formalisms widely used in Statistical Relational Learning (SRL) that combine graphical models with first order relational representations [123], such as Markov Logic Networks (MLNs) [124] and Bayesian Logic Networks (BLNs) [125]. A first-order probabilistic model is typically represented as a collection of first-order logic formulas with confidence scores, as illustrated in Fig. 7. More details about SRL can be found in [126,127].

Tradeoffs in Context of SRF: The most obvious benefit that results from combining probabilistic reasoning with first-order logic is that relational inferences can be used to model uncertainty, becoming more flexible to noisy or contradictory evidence. Another advantage of this representation is that its world definitions are more compact because variables act as placeholders for entities, which allows them to make relational rules interchangeable among entities. In contrast, languages based on propositional logic or propositional probabilistic graphical models (e.g.,MNs, BNs, and PDAGs) assume each symbol represents a concrete fact or entity. However, while first-order probabilistic models allow the compact writing of rules, their representation rapidly expands when performing probabilistic inference or learning because first-order probabilistic models still need to be grounded to their constituent probabilistic graphical models for computations. For example, a MLN needs to be grounded to a MN, which contains every possible assignment to the variables in the MLN, as illustrated in Fig. 7. The scalability of inference and learning can be partially addressed by leveraging structures in the models. For example, domain-lifted inference algorithms exploit symmetries within Markov networks by identifying symmetries directly from first-order structures without grounding MLNs. However, symmetries are difficult to find and can be easily destroyed by evidence. Due to this scalability issue, applications of first-order probabilistic models in large-scale SRFs remain limited.

Uses and Applications: First-order probabilistic models have been used to construct multi-relational probabilistic knowledge bases. In [32], a MLN is used to store knowledge between object properties and affordances by using probabilistic relations such as *isA*, *hasAffordance*, and *hasVisualAttribute*. The MLN allows for a variety of queries, for example, predicting affordance based on properties extracted from object images and inferring typical features of objects with a specific affordance. Similarly in [128], a MLN encoding relations between object properties (e.g., shape, size, and logo) is used to fuse information from different perception routines for collective classification. In [129], a probabilistic programming language, ProbLog, is used to construct a multi-objects affordance model. The probabilistic logical rules can deal with uncertainty in perception and action outcomes. Through the use of placeholder variables in place of individual objects, the relational representation is able to generalize the affordance model learned from 2 objects to any arbitrary numbers of objects. In [130], Distributional Clauses, which is a first-order probabilistic model that supports modeling continuous probability distributions, enables occluded object search by encoding different spatial relations between objects such as co-occurrence and stacking in addition to affordance related relations. A dynamic version of Distributional Clauses is used for object tracking during human activities [131]. The physics laws and common sense knowledge (e.g., if an object is on top of another object, it cannot fall down) that are encoded in probabilistic and continuous first-order rules help robots robustly track objects even with occlusion.

4.7. Neural networks

A Neural Network (NN) is a parameterized function approximator $\mathcal{F}$ that defines a mapping from input data $\mathcal{X}$ to output results $\mathcal{Y}$ (i.e. $\mathcal{Y} = \mathcal{F}(\mathcal{X}, \Theta)$, where Θ represents parameters of $\mathcal{F}$) [132]. The NNs used in semantic reasoning frameworks usually have two distinctive features. First, these NNs often make use of data containing semantic features, such as natural language descriptions of manipulation actions [84], or subgraphs from a knowledge graph [133]. Second, the objective of these NNs is to learn not only the mapping from input to output, but also the structure and semantic relations that underlie the data [62]. Embedding methods [134], in particular, represent a family of NNs that focus on encoding the structure of data by projecting input data into vector spaces in which spatial relations reflect the semantic relations of the input data.

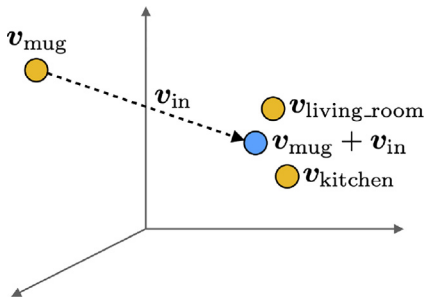


Fig. 8. An example neural network embedding. Entities and relations are represented as vectors in a multi-dimensional space. As the embedding aims to store the knowledge that a mug can appear in a kitchen or a living room, the values of both rooms should be close to the value of the *Mug* plus a vector that represents the *In* relation.

Tradeoffs in Context of SRF: NNs are a flexible computational framework capable of learning highly complex relations from data that are often difficult to encode manually. NNs can also take data of almost any form (e.g., task label, natural language instruction, image, point cloud, grasp pose, and trajectory), thereby allowing multimodal data to be combined and reasoned about collectively in a principled way. However, the adaptability of NNs comes at a cost to semantic reasoning for robotic applications. First, training NNs requires large amounts of data, which is often challenging to obtain, particularly in physical environments. Second, NNs are not as transparent as logic or probabilistic models, therefore reducing the explainability of the decision-making process. Additionally because the learned semantics within the NNs are challenging to extract, a NN trained in one domain may not transfer well to another (e.g., simulation vs real-world). While progress continues to be made on this front in the deep learning community, it is still a serious concern for practical applications involving physical robots operating autonomously in real-world environments (see Fig. 8). .

Uses and Applications: Neural networks have been used in most recent semantic reasoning frameworks. Different neural network architectures offer inductive biases suitable for different data structures. Recurrent Neural Networks have been used to model sequential data for storing history of observations and actions [61,135–137]. Graph Neural Networks have been applied to graph-structured data, modeling spatial relations between objects in both small scenes and large spaces [69,138–140], connections between concepts in knowledge graphs [77,86], and functional interactions between keypoints on objects [141]. Transformers have been shown to be effective at capturing long-range dependencies in sequential data [142]; they can also reason about binary and higher-order relations between entities [64,143–145]. Besides exploring different architectures, SRFs apply neural networks to reason about multimodal data and combine inference on symbolic and sensorimotor data. Different modality-specific networks can be used to map metric-level data (e.g., images, language instructions, point clouds, and tactile signals) to latent codes, which can then be organized in meaningful ways [75,84]. Early fusion of multimodal data at the feature level has also been observed, such as the LingUNet widely used for language-conditioned manipulation [74,146–149]. Transformer networks also serve as an suitable architecture to perform multimodal learning [97,144]. We also see an recent trend in leveraging large pretrained language models as a knowledge sources, such as in the application of grounding natural language instructions to task plans [94,95,150]. Pretrained vision and language models such as CLIP have been used to closely align visual observations and novel language for improving generalization [50,74,76]

Embedding methods also have been widely used in semantic reasoning frameworks to capture relations within the data. In [151], knowledge graph embeddings are used to capture relations between household objects and their attributes, enabling the robot to predict locations of objects, likely materials for objects, and affordances of objects. Word embeddings capturing similar word meanings are used in [152,153] to perform multimodal language grounding and learn common sense navigational knowledge, respectively. The work of [154] explores a multimodal embedding that mapped trajectory, language instruction, and object point cloud data to the same embedding space.

4.8. Summary of computational frameworks

Robots operating in complex human environments, such as homes, offices, and hospitals, require the ability to model uncertainty in the environment, reason about the world at varying levels of abstraction, adapt to changes in schedule, task requirements or object placement, be transparent in their reasoning and choices, and scale to multiple domains. Each of these challenges can be aided by semantic reasoning. Therefore in the context of SRF, we associate these challenges to five crucial characteristics of computational frameworks:

1. *Modeling uncertainty* - ability to model the inherent uncertainty and variability of the real world;
2. *Expressiveness* - ability to represent reasoning patterns at different levels of complexity, such as propositional, first-order, second-order, and etc;
3. *Adaptability* - ability to efficiently adapt the knowledge representation in response to new observations;
4. *Explainability* - ability to communicate information in a clear way and the transparency of the decision making process;
5. *Scalability* - ability to effectively model complex, real-world environments consisting of hundreds of objects.

Existing computational frameworks have succeeded in meeting one or more of these challenges, though no framework to date has excelled in all five areas.

Modeling uncertainty: Probabilistic models, including MNs, BNs, PDAGs, and first-order probabilistic models, are inherently effective at modeling uncertainty. When using other frameworks in situations with nondeterministic information, probabilistic variants of these frameworks, such as Bayesian Neural Networks [155], Deep Ensembles [156] and Probabilistic Description Logics [157], can be selected.

Expressiveness: First-order frameworks such as DLs and first-order probabilistic models are more expressive than probabilistic graphical models (e.g., MNs, BNs, and PDAGs), which use propositional random variables. Prior work has shown that neural representations can approximate first-order logic, therefore producing more expressive reasoning patterns [158–160].

Adaptability: Incorporating new information into probabilistic models is hard as it often entails learning new parameters or structures. In contrast, new knowledge can be more easily added as new assertions in DLs and as new nodes or edges in semantic graphs. Since NNs are typically capable of generalizing learned models to new data, new information can also be reasoned without retraining.

Explainability: Symbolic frameworks, including all frameworks previously introduced except NNs, are typically easier to interpret than non-symbolic approaches. Within symbolic approaches, reasoning in a DL or a semantic graph often is involved with a subset of contained information while reasoning in a probabilistic model depends on all random variables in the model. The modular and local reasoning mechanism, as a result, provides more interpretability.

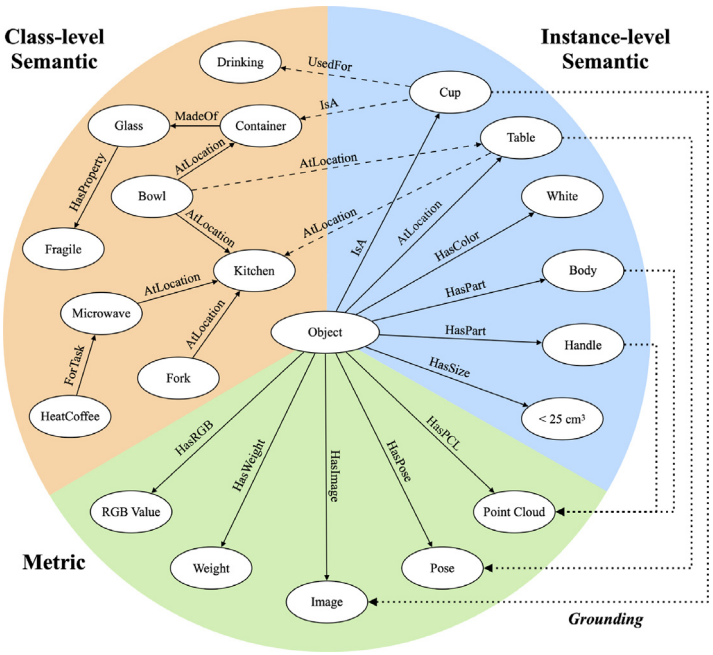


Fig. 9. Structure of object representations.

Scalability: NNs, semantic graphs, and DLs have all been used with large-scale data [27,151,161]. In contrast, probabilistic models are less efficient at handling a large amount of information. As for first-order probabilistic models specifically, scalable inference and learning are still open research problems.

When using existing computational frameworks for semantic reasoning, trade-offs between the five characteristics mentioned will need to be considered according to the reasoning problem domain, available knowledge, and formal verification requirements. However, a direction for future work is to continue to push toward a framework that excels in all five areas.

5. Building world representations from data

In this section, we turn our attention to constructing world representations that encode various types of semantic knowledge. Specifically, we consider world representations that model five semantically meaningful entity types: *objects*, *spaces*, *agents*, *tasks*, and *actions*. These five categories cover different aspects of the world a robot interacts with, and each is also inherently different in the modality and underlying structure of its data. By analyzing these five categories of representations, we aim to provide developers the useful languages to build up multi-modal, multi-relational, and multi-domain semantic reasoning frameworks that can accurately model and reason about the world.

5.1. Representing objects

In this survey, objects refer to the physical entities that robots can perceive and interact with. Representations of objects model object properties and the relations between these properties. Leveraging object representations helps create robot behaviors that are intelligent (e.g., manipulation based on object functionality [62,162]) and robust (e.g., recovery of task failure with object substitution [163]). In addition to storing information associated with objects, object representations also serve as a foundation for representations of other entity types. For example, modeling the function of a space depends on modeling its contained objects, and modeling the belief state of an agent often requires modeling the agent's perception and understanding of objects.

In discussing the structure of object representations, we refer to three types of encoded information: metric, instance-level semantic, and class-level semantic, as illustrated in Fig. 9:

- *Metric:* Metric representations of objects contain information unique to each object instance (e.g., image, point cloud, and pose). As these representations store high-fidelity information, they are often numerical and continuous. Aggregating these low-level data for each instance or across instances allows robots to build features that are used to recognize objects and properties, enabling grounding [80]. Meaningful semantics and suitable abstraction can also be automatically discovered from the metric data [154,164]. Maintaining the raw data, in addition, helps avoid the combinatorial explosion that would arise from storing all possible qualitative semantic representations (e.g., pair-wise relations between objects) [161]. Implicit representations of objects can be used to achieve view-invariance [165] and efficiently encode multimodal properties [166]
- *Instance-level Semantic:* Instance-level semantic representations include abstract concepts about each object instance, which are grounded in a robot's observations. For example, in Fig. 9, the object is classified as a cup from its image. Since these representations are abstract, they tend to be qualitative and discrete. The compact form of instance-level semantic representations facilitates efficient semantic reasoning but they also maintain enough information that can distinguish between object instances and guide precise robot behaviors [62,72,73,101].
- *Class-level Semantic:* Class-level semantic representations encode abstract knowledge that generalizes across an entire class of objects. For example, a cup is a container and can be used for drinking. Similar to instance-level semantic representations, the class-level representations often are symbolic. Though class knowledge cannot be readily perceived, it encodes many useful priors that can be extracted from semantic knowledge sources or manually created by experts [26,81,167,168].

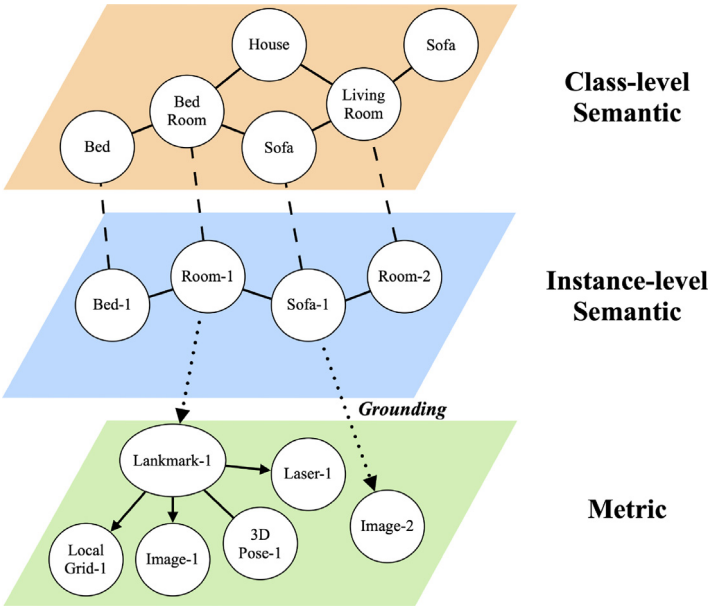


Fig. 10. Structure of space representations.

Together these three types of object representations allow a robot to ground perceivable symbols, learn class attributes from instances, and relate objects to other abstract semantic concepts. These three types of information are observed in many complete semantic reasoning frameworks such as KnowRob [81], RoboBrain [27], ORO [26], and OUR-K [119]. To provide concrete examples, we now discuss how a variety of object properties and relations have been modeled in representations of each type.

Metric representations have been used to store complete and accurate information about object appearance, shape, location, and use. Specifically, 2D images [27] and 3D CAD models, meshes, and point clouds [27,81,84,154,169] encode raw visual and geometric information. Intermediate representations such as image features [27,117], primitive shape models [81,117], and object dimensions [130] store processed 2D and 3D information. Objects are also represented by parts extracted from methods like curvature-based segmentation [81,162,170] and Reeb-graph segmentation [107]. Keypoints [171] and neural descriptor fields [165] have been used to establish shape correspondence between objects in the same categories. Apart from visual and geometric information, object poses are also maintained in the numeric form, which enables both metric reasoning [83,111,131,172] and extractions of qualitative spatial relations [81,161]. Object functions are often described in the language of affordance, which is introduced by Gibson as the properties of an object that determine possible actions to perform on it [173]. At the metric level, the spatio-temporal representation of affordance defines both the 3D location of interaction on an object, and the motion trajectory for manipulating it [172,174]. Articulation models are also included in prior work to specify manipulation trajectories for objects such as doors and drawers [169,170].

Instance-level semantic representations have been used to encode a variety of abstract information about individual objects. Class labels represent object categories, which can be predicted from end-to-end image classifiers or grounded in low-level and semantic features [107,111,117,175,176]. Shape labels extracted from metric data are used to infer affordance [8,129,130], guide grasping [101]. Affordance labels are also directly extracted from visual data [8,62], and could be learned from physical interactions with objects, especially in the case of tool use [141,177,178]. Another widely used representation is object part labels, the definitions of which are based on common object parts (e.g., door knob) [117,161,162,170], affordances (e.g., pourable and containable) [8,62,174], and other heuristics (e.g., top and bottom) [175]. Object states [62,175], materials [8,62], and other more open-ended language descriptions [23,73,152] have also been used to characterize object instances.

Class-level semantic representations have been used to store general knowledge about object classes. External knowledge graphs and ontologies, such as WordNet, ConceptNet, and Cyc, are connected to objects through class labels [17,86,151–153,167]. To deal with partial and inaccurate information from these knowledge sources, some methods have used these priors as high-level constraints or probabilistic statements instead of ground-truths [32,46,81,161]. General spatial knowledge is also expressed in terms of object classes (e.g., a kitchen typically has a sink and a microwave), which we will discuss in more detail in Section 5.2. Prior work has also modeled the relations between different types of class-level knowledge and used them to infer missing or unknown information such as affordances and possible object locations [17,151,153,179]. A unique representation at the class level is word senses, which eliminate the ambiguities between objects with the same name but different meanings (e.g., a bowl as a container or as a stadium) [17,19,180].

5.2. Representing spaces

Spaces in this survey refer to continuous expanses with specific volumes defined by their boundaries. We distinguish the concepts of *map* and *semantic spatial model* in the context of semantic reasoning. The primary purpose of a map, as used in the literature [181–183], is for use in navigation and localization. We introduce the concept of *semantic spatial models*, which are also designed to represent spaces, but with the purpose of facilitating task execution. For example, semantic spatial models often encode semantic object types and attributes in addition to space occupancy or room topology, which enables task planning [167,184], visuomotor navigation for object search [139], and visual room rearrangement [43]. By using semantic spatial models, queries can be made to reason about not only spaces but also other objects, agents, actions, or tasks associated with them.

In discussing the structure of semantic spatial models, we define three main layers: metric, instance-level semantic, and

class-level semantic (Fig. 10). Note that the structures of semantic spatial models and object representations are closely related as they share similar levels of abstraction.

- *Metric:* The metric layer of a semantic spatial model describes spaces and contained entities quantitatively through various numeric measurements (e.g., laser scan, 3D pose of landmark, occupancy grid). Metric representations allow robots to interpret the geometry of environments but lack the abstraction and semantic meaning required to perform efficient reasoning.
- *Instance-level Semantic:* The instance-level semantic layer of a semantic spatial model enriches the metric layer by assigning semantic meaning to measurements [82]. This layer has nodes that represent observed concepts (e.g., recognized objects and rooms) and edges that represent observed relations (e.g., traversability and part-whole relations), from which belief states can be extracted to be used by task planners.
- *Class-level Semantic:* The class-level semantic layer of a semantic spatial model allows for further generalization by including general knowledge of class types. This layer can store abstract properties about the class types (e.g., bedrooms and living rooms are rooms in house environments) and encode common rules (e.g., sofa and TV are typically observed in living rooms) [114,167].

Prior work has explored techniques for combining information across various subsets of these layers, as discussed below.

In many semantic spatial models, all three layers are in use. This is particularly true for models of large spaces composed of many rooms or locations, such as an entire kitchen [20] and a whole office floor [114]. In these spaces, all three layers are crucial as robots need to navigate with the metric information, efficiently reason about individual entities with the instance-level semantic knowledge, and generalize with the class-level semantic knowledge. The exact details of how information is encoded at each layer differ across frameworks. In [82,118,119,167], class-level knowledge about typical room types (e.g., a bedroom has a bed) are encoded in expert-created ontologies. In order to overcome the rigidity of rule-based systems, co-occurrence statistics of objects and room types are stored as conditional probabilities [83,185] and encoded in factor graphs which can be integrated with probabilistic measurement and prediction models to infer likely object locations [112]. In [114,115,184], room-object relations, room-appearance relations, and room-room connections are simultaneously represented in PDAGs. In [20], class-level knowledge about objects and spaces are obtained by establishing connections to external commonsense ontologies. Recently, large language models is used to enhance the instance-level map with commonsense knowledge [186].

All three spatial layers are not always needed, and subset pairs have been used in some applications. With only abstract knowledge, the instance- and class-level semantic layers are used to qualitatively reason about missing objects in different scenes [161,179]. Some approaches model spaces only at the class-level. The abstract knowledge about types of spaces provides high-level guidance for object search and rearrangement [86,151].

Metric and instance-level semantic layers are commonly used to closely integrate perception and spatial reasoning. Predicates representing inter-object spatial relations, such as *in*, *on*, and *left of*, can be inferred through functions learned from data [67,143,187,188]. These spatial relations have been used for contextual reasoning [130], language grounding [189,190], modeling world dynamics [191], and specifying manipulation goals (e.g., moving the cup to the right of the bowl) [67,192–194]. 2D and 3D scene graphs offer a compact representation to simultaneously encode spatial and semantic information of multiple entities, where metric information such as object poses, visual appearances, and 3D models can provide precise specifications for the qualitative concepts and relations [69,138,195–197]. Beyond predefined spatial relations, continuous valued embeddings can be used to encode scene graphs relating objects in a view-invariant representation [65]. To maintain a history of observed instance-level scene information from egocentric observations and aid in long-term memory recall, spatial maps have been built by projecting instance segmentation masks onto a 3D voxel space [43,149,198] or using learned Gated Recurrent Unit [136]

5.3. Representing tasks

Tasks in this survey are defined as structures that encapsulate and organize grounded actions. Task planning of robots in the physical domain poses challenges such as intractable domains, partial observations, stochastic effects, and disconnected goals. Semantic reasoning is used to alleviate these problems by adding organized information. We show three essential structures of semantic task models in Fig. 11. The two upper panels of Fig. 11 show how tasks can be organized as sub-tasks in a class-level task hierarchy. This structure is used to represent tasks at an abstract level. The two lower panels of Fig. 11 show how each sub-task could be represented and linked to information in representations of other entity types in class-level sub-task template and class-level sub-task instantiation. The example structures illustrated in Fig. 11 and defined below illustrate one approach by which semantic knowledge can be incorporated into tasks.

- *Class-level Task Hierarchy:* recursively decomposes a task into sub-tasks until a known set of class-level sub-task templates are reached. A task hierarchy uses different levels of abstraction to enable re-use of task plans and effectively represents the sequential nature of tasks with multiple steps [199].
- *Class-level Sub-task Template:* defines a sub-task by placeholder actions and placeholder conditions. This structure serves as the blueprint of the class-level sub-task instantiations and allows for better generalization by grouping instantiations from the same template [184]. This representation also pertains to lifted operators in the task and motion planning literature [200].
- *Class-level Sub-task Instantiation:* inherits all the restrictions from the class-level sub-task template by instantiating pre- and post-conditions in terms of class-level concepts in other world representations. Connection to other representations help infer missing information in tasks and infer the correct ordering of sub-tasks. (For the example in the figure, because the refrigerator is connected to pancake mix in object representation and kitchen in space representation, perceiving refrigerator is added as a pre-condition to help the robot find pancake mix in execution).

It is worth noting that the instance level of task specification is not shown here because its conditions are specified as concepts in other representations. For example, the realization of a sub-task is an action. The grounding of class-level concepts of tasks are discussed separately in Sections 5.1, 5.2, 5.4 and 5.5.

Prior work has explored techniques for utilizing different subsets of structures to organize tasks. Incorporating hierarchical structure into task representations, which can be learned from a repository of tasks [163,201,202] and encoded by an expert [119,199,203] promotes robustness and efficiency of task execution and planning. Decomposing tasks in hierarchies helps promote

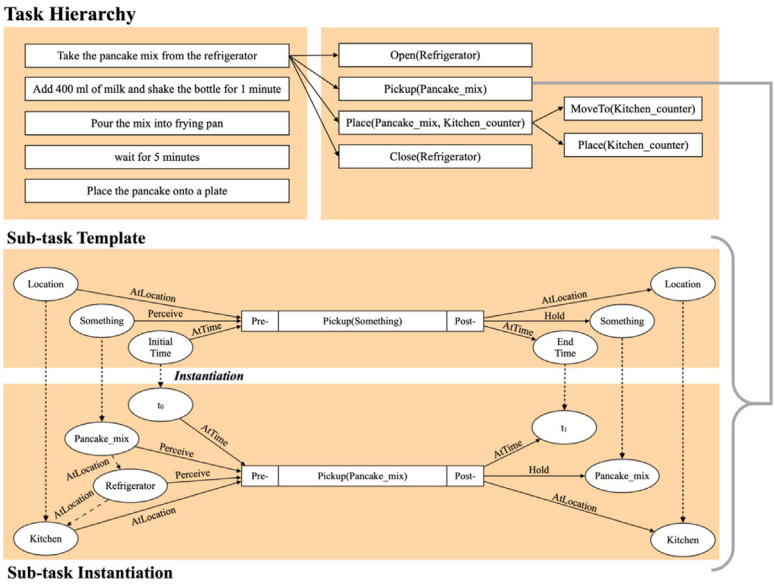


Fig. 11. Structure of task representations.

robustness to stochastic task outcomes [199,204] and representational efficiency through the re-use of tasks [21,201]. Additionally, a hierarchical task decomposition enables more efficient planning by planning at different abstraction levels (e.g., planning at class-level) [204], or executing partial plans (e.g., only planning to decision points which require execution) [199]. In addition to modeling robot tasks, class-level task hierarchies have been used to model instructions provided by humans during interactions [19,26]. Tasks defined in natural language can be decomposed into sub-tasks in natural language [94,95,150], or expanded into programs [205] through pre-trained language models.

In addition, class-level task hierarchies have been modeled in prior work to express the ordering of task steps both explicitly in symbols [19,102,184,195,206,207], or implicitly in neural networks [201,208–210] The ordering constraints within the class-level task hierarchy structure enables robots to infer a sequence of actions to reach a goal state [82,211], select the next action given the current state [42,208,209], and repair failed task plans [115,184]. Graphical task structure to represent ordering constraints has be learned from demonstrations and provide the flexibility of generalizing to unseen tasks with similar skill requirements [212,213].

Besides encoding sequential and hierarchical structures of tasks in task representations, prior work has also adopted class-level sub-task template and instantiations by ensuring correspondences between generic and task-specific information [81,204,214]. Learned state and action abstractions have been used to generate an abstract sub-task template that yields efficient and scalable task-specific policies [215]. Several methods leverage class-level sub-task instantiations to establish the connection to other world representations. In [119], pre- and post-conditions are defined in terms of concepts in object and space representations. Similarly in [82], conditions link to semantic maps and topological maps. In [81], conditions additionally incorporate temporal information for inferring and explaining temporal order of tasks.

5.4. Representing actions

Actions in this survey refer to physically grounded robot behaviors. Inherently, actions are continuous, stochastic, and require feedback. In the context of SRF, each action also has related semantic representations, which create a high-level abstraction for the low-level sensor–motor experience and associate the action with contextual information. The semantic representations of actions are compact and can be easily connected with high-level objectives and task plans. By leveraging the encoded information in these representations, robots can generalize behaviors to novel situations [84,216] and take actions appropriate for the context [108].

In discussing the structure of action representations used in prior work, we build on the framework of object–action complex (OAC) [203], which aims to bridge the gap between high-level abstract reasoning and low-level control knowledge for actions. We adapt the OAC representation in the context of SRF so it is consistent with the language of this survey. Specifically, action representations have two layers: metric and instance-level semantic, as illustrated in Fig. 12.

- *Metric:* The metric representations contain low-level information related to actions. The representations can be divided into representations of the world states (e.g., an image of the scene, a pose of the manipulated object) and representations of actions themselves (e.g., a trajectory, a feedback controller). Executing an action causes changes in the physical world that transform the initial world state $\mathcal{WS}_0$ to the resulting world state $\mathcal{WS}_r$.
- *Instance-level Semantic:* The instance-level semantic representations create high-level abstractions for actions. The word instance-level emphasizes that these representations are grounded in the metric level data. Similar to the metric layer, the semantic layer consists of semantic representations of world states and semantic representations of actions. The semantic representation of an action can be a prediction function that models how the semantic world state will transform from the initial semantic state $\mathcal{S}_0$ to the predicted semantic state $\mathcal{S}_p$ if the action is executed [191,203]. However, the semantic representation of an action can also be as simple as a language description of the action [74,75,84,137].

Another essential feature of action representations is the tight connections between these two layers. As robot actions are noisy, keeping the mappings from metric to semantic representations of world states allow robots to discover the discrepancies between

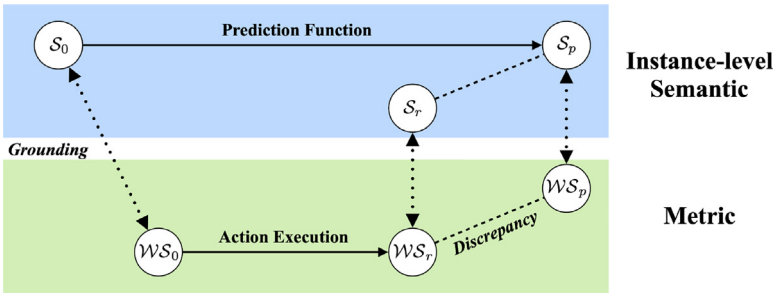


Fig. 12. Structure of action representations.

the physical actions and imagined actions. For example, the difference between $\mathcal{S}_r$ and $\mathcal{S}_p$, or the difference between $\mathcal{WS}_r$ and $\mathcal{WS}_p$, as illustrated in Fig. 12.

All the described components of action representations are observed in prior work. In OAC [203], the discrepancies between the effects of actions in the semantic and metric layers have been used as feedback to both improve the prediction function for push actions and facilitate the learning of task-agnostic and specific grasps. In [129], a prediction function for primitive actions is implemented in a SRL formalism, while [64,191] learn to predict the results of an action represented in natural language in the form of logical predicates. Having the conceptual model of actions allows a robot to pick actions most likely to produce desired effects on objects.

Besides learning conceptual models of actions, semantic representations of actions are also commonly used for creating generalizable actions. In [216–219] the controllers of the robot manipulators are parameterized by task functions grounded in geometric features of objects (e.g., keep the main axis of the spatula pointed at the center of the oven). Actions are generalized to new objects by imposing these task functions as semantic constraints on functionally meaningful geometric features (e.g., handles). Another approach combines low-level trajectories and point clouds of object parts with high-level language instructions in NNs to create generalizable representations of actions [84,154]. Recent work on language-conditioned manipulation directly maps from natural language instructions to continuous actions and leverages compositional generalization supported by language commands [74,137,147,148,220]

Semantic representations of actions also allow high-level constraints to be easily incorporated. In [108], user preferences of trajectories are modeled as a scoring function that is computed from both metric information (e.g., robot arm configurations and distances to surrounding surfaces) and semantic representations (e.g., properties of objects that are near to the trajectory). Similarly, language is used to correct robot motions by predicting a cost map for path planning [221]. Many existing approaches to semantic grasping have also leveraged semantic representations to compactly encode contextual information [62,101,175]. Instead of mapping language directly to actions, correspondences between language description and image-based subgoals are learned from offline data for model-based planning [76].

5.5. Representing agents

Agents in this survey are defined as a robot itself, other robots, and humans in the environment. Inferring human belief states is often vital for Human–Robot Interaction (HRI), while understanding the capabilities of the robot's self and others is often necessary for task-collaboration and knowledge sharing. The representation of each agent (i.e., the self, other robots, and humans) includes that agent's capabilities, configurations, and belief states, as illustrated in Fig. 13. While an agent has direct access to its own belief states and capabilities, the belief states and capabilities of other agents can be received through message exchange between agents [21] or inferred from perceptions [222].

- *Capabilities & Configuration:* The capabilities of an agent can be inferred from its physical configuration. In addition to spatial poses and dynamics of body parts, physical configurations can include semantic descriptions linking properties, algorithms, and other semantics to reason about overall agent capability with respect to actions or tasks [22].
- *Belief States:* The belief states of agents can consist of the previously mentioned representations (i.e., object, space, task, action) or attributes specific to agents (e.g., intent, preferences) [26,223]. Within the agent's belief state, the different representations will interact as the world state changes and the representations are updated.

Human agents have been modeled in semantic reasoning frameworks to enable naive user interaction with complex semantic reasoning frameworks [19,25,26] and to model user preferences [16,16,32,105,174,224]. Semantic reasoning frameworks that leverage ontologies [20,25,26] have modeled human agents by including relevant classes (e.g., agent, embodied agent, human, robot, body part), properties (e.g., *is desired*, *sees*, *has in hand*), and alternative cognitive models. The addition of these classes and properties has enabled complex interaction [25,26] and learning of human preferences such as where to perform actions [20,170]. Preferences of human users in context of a task can be encoded as latent vectors, which the robot's policy is conditioned on [140,225], or reward functions, which can be learned from demonstrations [226] or from static observations of the world, since the world is already optimized for the preferences of a human user acting on it [227]. In addition to learning preferences, manipulation tasks have been learned from human demonstrations through text [180] or videos [2,202], and [226] propose to handle suboptimal demonstrations by underestimating user capabilities. Other methods have attempted to capture structured human biases in decision-making [228] and state estimation [229], which lead to deviation of human actions from the optimal actions. MLNs have been leveraged to reason probabilistically about human actions in relation to objects in use [32] and human commands in relation to the robot's perceived environment [19]. Object arrangements have been used as context to predict future human activities through temporally evolving sequence of scene graphs [230], or conditional random fields [27,231], and to represent spatial user preferences through probabilistic graphs representing object layouts with human pose as context [232].

When multiple robot agents need to collaborate, it is crucial to distinguish between robots and their capabilities. In [120,121], each robot has its own ontology and all robots are defined in a global ontology. Because different robots perceive and interact with the world differently due to different capabilities or perspectives, each robot should have its own world representation. The

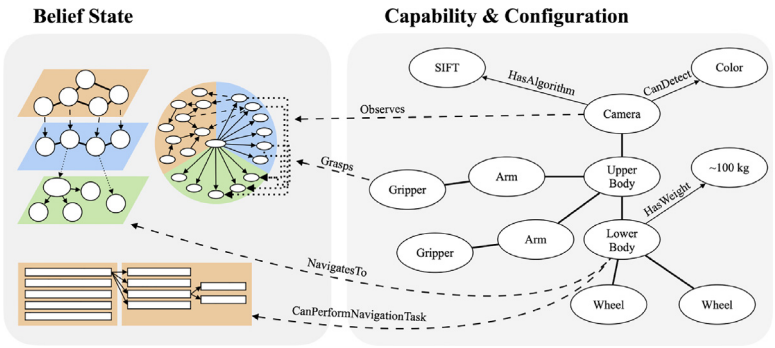


Fig. 13. Structure of agent representations.

work of [176] differentiates object representations of different agents by defining object affordances with respect to not only object properties but also locations, properties, and capabilities of agents (e.g., push affordance of refrigerator door requires the agent to have push ability and be in kitchen). A coordinator, which has access to each agent's ontology and the common goal, can then reason about task assignments based on each agent's affordances in the environment. Similarly in [233], an ontology is combined with a PDDL planner to coordinate multiple heterogeneous robots by using robot specific skills. The semantic robot description language (SRDL) [22] has been used in [21] to model agent capabilities. SRDL allows for the unambiguous definition of a robot's hardware, configuration, and software, which in turn helps specify its capabilities. Using SRDL, a robot can reason about the likelihood of successfully performing specific tasks based on both its capabilities and experience from previous attempts.

6. Combining SRF components for intelligent robot behavior

Many intelligent robot behaviors can be achieved through semantic reasoning — the inference and computation processes that require all three core component of semantic reasoning frameworks. Examples of such processes include discovering the semantic similarity between environments, inferring missing information from underspecified instructions or partially observable world state, or parameterizing robot tasks to be appropriate for the context. In this section, we study the reasoning capabilities that are enabled by combining multiple of the previously discussed SRF core components.

In Table 5, we characterize various semantic reasoning tasks by the world representations and reasoning objectives. For example in [114], a robot joins the object model (O), which encodes attributes that potentially influence objects' locations, with the spatial model (S), which stores typical spatial appearance of objects, to accomplish the "Inferring Object Location" reasoning objective. While the table provides a full summary of techniques, in Sections 6.1–6.9 below, we more closely examine the integration of SRF components in several reasoning tasks.

6.1. Top-down and bottom-up object perception

Object classification techniques typically fall into one of two approaches: bottom-up and top-down. Bottom-up machine learning algorithms, such as [6,255], train models to recognize objects based entirely on low-level object appearance features. Top-down techniques, such as those in [8,23,117,118], classify objects based on abstract characteristics of objects (e.g., a refrigerator is a white hexahedron with a door). Both approaches have limitations — bottom-up techniques are prone to provide ambiguous results [111] and have lower performance when objects are partially occluded [131], while top-down techniques are too rigid to capture the diversity of instances in many object classes.

Several semantic reasoning frameworks have developed techniques for combining bottom-up and top-down reasoning to leverage their relative strengths. In [237], maximum a posteriori estimation integrates bottom-up object classification scores with top-down inference over occurrence probabilities of spatially related objects. Similarly in [110,111], MNs model object features and spatial relations to combine reasoning from both directions. In addition to object recognition, in [131], physical laws and commonsense knowledge (e.g., a small object can fall inside a large box) defined in a SRL model help track objects even when they go out of sight. High-level knowledge can also result in more efficient learning of bottom-up methods. In [46,133], few-shot and zero-shot object recognitions are achieved by generalizing visual classifiers in Graph Neural Networks that capture high-level semantic similarities between objects.

6.2. Inferring object affordances

The ability to reason about affordances enables robots to choose actions that are suitable for a given object and produce the desired effects. Most prior work has modeled affordances as intrinsic properties of an object that determine possible actions to perform on it (e.g., a cup is pour-able, and a fork is stab-able and lift-able). Symbolic affordance labels provide a general prior for possible ways to utilize an object. Such labels are retrieved from a semantic graph based on objects' class labels in [27]. By modeling the correlations between affordance labels and other object properties (e.g., materials, locations, and shapes), an object's affordance can be predicted from its properties using an MLN [32], BLN [17], or NN Embedding [151].

However, reasoning about affordance labels alone ignores the essential interaction between objects and the environment. Grounded affordances can be discovered from robots' interactions with the world and encoded in semantic representations. In [129], a BLN models this interaction by representing objects, actions, and effects as separate random variables. Reasoning about affordances is modeled as various sub-tasks, such as estimating the effect of an action performed on an object, or predicting the action given the desired effect on an object. Although this decoupled affordance model reflects the connection between objects and the environment, discrete labels of actions and effects still cannot be directly used by robots to produce continuous behaviors. As a result, a spatio-temporal representation of affordance is used in [172] that reasons about continuous manipulation actions through joint probabilistic inference over spatial and temporal distributions. Physics simulators provide a way for robots to simulate interactions between objects [234–236] and discover functions of objects through trail and error [141,177,178]. These knowledge can be distilled into models that predicts key regions on objects and manipulation policy that potentially generalizes to real-world objects.

Table 5
Referenced object (O), space (S), task (T), action (Ac), and agent (Ag) semantic reasoning works with related world representations. Check marked world representations on the left side of the table are used to accomplish corresponding reasoning objectives listed on the right. Gray shading is used to highlight groupings of techniques that address object-, space-, task-, action-, and agent-focused problems.

World representation					Reasoning objectives	Referenced works
O	S	T	Ac	Ag		
✓					Inferring Object Affordance	[27,234]
✓	✓					[17,129,151,179]
✓			✓			[141,172,174,177,178,235,236]
✓				✓		[32]
✓					Object Perception	[8,46,117–119]
✓	✓					[110,111,131,179,237]
✓			✓			[23]
✓	✓				Inferring Missing Object	[179]
✓				✓		[238]
✓	✓				Space Classification	[82,83,114,118,119,184,186,239]
✓	✓				Semantic Navigation	[42,43,50,61,77,78,82,86,96,119,135,146,149,240–244]
✓	✓				Localizing object	[66,72,73,97,208,244,245]
✓	✓				Inferring Object Location	[17,20,68,81,82,85,112,114,118,130,139,151,153,167,185,246]
✓	✓			✓		[105,140]
	✓	✓			Inferring Suitable Location for Task	[21,169]
✓	✓				Reasoning about Spatial Relations	[81,111,130,179] [63,64,67,143]
✓		✓			Inferring Task Hierarchy and Ordering	[16,19,21,94,95,104,129,138,163,184,195,201,205,212–214,214,225,247]
✓		✓			Inferring Manipulation Task from Objects	[19,106,178,202,248]
✓	✓	✓			Invoking Task for Decision Confidence	[82,115,184,204,208]
✓	✓	✓			Enlarging Task State Space	[214]
		✓			Task Instance Domain Reduction	[214]
✓	✓	✓			Task Monitoring & Perception	[23,119,195,238,242]
✓			✓		Semantic Specification of Manipulation	[24,191,196,216]
✓			✓		Semantic Grasping	[62,101,107,162,175]
✓			✓		Generalizing Manipulation Trajectory	[84,154]
✓			✓		Learning Preference over Trajectory	[108]
✓		✓	✓		Learning Task Level Behavior from Action	[129,203]
✓		✓	✓		Language-Conditioned Manipulation	[48,74,76,137,147,148,209,220,221]
✓	✓			✓	Learning from Human-Robot Dialogue	[87–89,249]
✓	✓			✓	Natural Language Interaction	[91,136,198,249–254]
✓	✓	✓		✓		[19,26]
✓			✓	✓	Human Pose Prediction	[32]
✓	✓	✓		✓	Heterogeneous Robot Team Planning	[176,233]
		✓		✓	Task Capability Matching	[21,22,233]
✓	✓	✓		✓	Learning Human Task Preferences	[140,225,227]

6.3. Reasoning about the semantics of space

Reasoning about semantic spatial models gives rise to new capabilities, such as visual navigation [243], and provides further robustness to localization and mapping. Using the knowledge of common objects in different room types, a robot can recognize its location (i.e., which room it is in) [83,114,118,119,184,186,239] and detect high-level localization errors when odometry fails [82]. Querying an ontology containing typical spatial relations between objects (e.g., a tv is near a sofa) allows a robot to find a target object that is not directly in its view by continuously navigating to spatially related objects [82,118] or incorporating the knowledge into a POMDP formulation aimed at searching a specific object [68]. The same reasoning pattern is behind many recent methods for visual semantic navigation [42,50,86,96,146,240–244]. In these methods, inference on neural networks, instead of logic rules, is used to determine navigation actions given observations of the scene. Semantic spatial models are also used to perform multi-object rearrangement tasks, where the model encodes the desired configuration of objects in the environment via agent exploration and reasons about an action sequence to achieve the desired configuration [43,61]. Besides representing object knowledge in semantic spatial models, task information can also be incorporated. In [169], reasoning about the relation between execution success rates and robot locations in a Gaussian Mixture Model facilitates task reproduction.

6.4. Inferring object locations based on class- & instance-level knowledge

Class-level knowledge provides a general prior of typical object locations, while instance-level knowledge models the appearances of objects in each specific environment. Both types of knowledge have been used in prior work.

Class-level knowledge treats spaces and containing objects as general concepts (e.g., a bowl can be often found in cabinets, sinks, and dishwashers). Inferring object locations is achieved by retrieving typical locations of objects from ontologies [81,118,119] or logic rules [82,167]. Leveraging the intuition that related objects appear in similar places, a work predicts object locations

by finding semantically similar objects using the Wu-Palmer similarity measure [103] computed on an object hierarchy [81]. A method based on neural network embedding takes a step further by modeling different types of semantic relations between object properties and locations to infer locations of objects [151].

A different approach is to utilize instance-level knowledge, which pertains to the locations of objects in a specific environment. Existing methods often learn the co-occurrence statistics of objects in different locations. In [185], these statistics are modeled by conditional probabilities. In [112], co-occurrence statistics are used as priors to inform the joint likelihood of object locations, and are mined from large-scale Flickr dataset. In [105], the Collaborative Filtering technique from the data mining community for addressing personalized user recommendations stores user-specific statistics. The work in [130] applies a SRL model to reason about different types of co-occurrences distinguished by spatial relations.

Ultimately, class- and instance-level knowledge should be reasoned collectively. In [85], a factor graph joints commonsense knowledge at the class-level as well as long term and short term memory at the instance-level. Similarly in [77,86], a Graph Neural Network combines continuous local observations with general prior knowledge from ConceptNet to help a robot navigate to target objects.

6.5. Task planning and learning

Reasoning about task semantics (see Section 5) has enabled improvements over the execution of task reasoning objectives in Table 5. Two fundamental challenges when reasoning about tasks involve either planning sequences of primitive actions that lead to the completion of goals [19,26,184,195,211,214,233,256] or learning from sequences of primitive actions that demonstrate the completion of goals [42,102,138,163,180,201,209,209,212,213,225]. Reasoning about semantic properties of tasks can help to improve task planning efficiency and accuracy while promoting generalization within task learning.

Many semantic properties of tasks have been leveraged to improve task planning and learning. Planning at different levels of abstraction leads to planning computational efficiencies by reducing the planning state space to classes of objects (i.e., t-box in description logics) while executing over instances of objects [204,214,257,258]. Modeling a task as a sequence of primitive skills to be executed and reasoning over modeled affordances [178,248] or latent representations [209,259] of these skills has led to successful generalization of planning algorithms across different task domains. The integration of uncertainties with distinct semantics through graphical or relational models helps infer the most likely task plan given unobserved world states or under-specified instructions [19,184]. Additionally, due to semantically ambiguous world states, planners with multiple world states can generate plans that reduce such uncertainties [82,115,119,208]. The hierarchical structure of tasks helps task learning generalize because of modularization enabled through the reuse of subtasks [21,67,201,256]. The prediction of future world states as tasks are executed, planned, or learned allows agents to detect, interpret, or explain plan failures [119,184,209]. The propagation of pre- and post-conditions in under-specified tasks allow missing task specifications to be inferred [16,19,211]. The grounding of natural language task instructions in world observations and actions allows agents to learn to generalize execution of novel instructions in novel environments [146,147,243].

6.6. Enriching task planners

Although tasks themselves contain rich semantics (e.g., preconditions, hierarchies), many works have exposed task planners to even more semantic knowledge. Enriching task planners with semantic knowledge further promotes reasoning objectives, such as task generation for decision confidence, task monitoring, and other reasoning objectives specific to tasks in Table 5. Generally, the semantic knowledge being exposed includes precepts about the situated environment or default knowledge (e.g., fridge likely contains food). These newly exposed semantics further enable task planners to make more robust primitive action plans.

Streaming the perceived world state to task planners increases situational awareness, making task plans more fit to the current world state. In [67,195], the world state is exposed in semantically meaningful ways to task planners to enable data-efficient state updates while promoting generalization. Sensor streams are processed into semantically meaningful axioms and predicates in [23,63,238] to update planners with environment variables that might become useful in later parts of tasks, such as the location of an item required in a later step of a task. In [23,238], planners can call perception routines to allow for active acquisition of environment states and semantic information when planning.

In addition to the perceived world state, exposing semantics about default knowledge regarding specific tasks, domains, and general concepts enables more robust task planning. Observations from multiple environments or execution histories are used to learn generalized rules or correlations between semantic concepts, such as rooms and objects, enabling task planners to prioritize task plans that are more likely to satisfy task goals based on observed correlations from prior environments [184,204,214]. Ontologies about common domains, such as households, provide task planners information useful during task execution such as object locations, uses, and suitable locations for certain actions and tasks [16,20,119,214]. General knowledge such as type hierarchies, semantic similarity, generic relationships, and other forms of general knowledge can be used to bootstrap task planners with a knowledge base when operating in new environments or repairing existing task plans to add robustness [17,19,163]. Encoded task-specific semantic knowledge can improve planning efficiency when the set of tasks are predefined by enabling planners to prune branches of plans that cannot satisfy task goals (e.g., searching a bathroom for an oven.) [82,115]. Additionally, semantics about human agents have been exposed to task planners to improve human-aware task planning capabilities, such as including human preferences in tasks and modeling human belief states during task execution [26,105,225].

6.7. Semantic grasping

Instance-level semantic knowledge provides precise and generalizable information for semantic grasping, which is the problem of selecting stable grasps that are functionally suitable for specific object manipulation tasks (e.g., when passing a knife to a person, a robot should grasp the blade instead of the handle) [260]. Some existing approaches to semantic grasping use metric data because reasoning about the specific locations and orientations of grasps requires detailed information about an object [260,261]. However, discovering structures in high-dimensional and irregular low-level features is hard and has prevented these methods from generalizing to a wider range of objects and tasks. Compared to metric data, instance-level semantic knowledge provides the necessary abstraction while also maintaining enough distinctive information to guide accurate selections of semantic grasps. A commonly used instance-level semantic feature is object part information. Suitable grasp regions based on

segmented object parts have been manually defined in logic assertions [162] and a SRL formalism [175]. In data-driven approaches, semantic grasps learned from task demonstration are reproduced on new objects by matching Reeb graphs that represent object decompositions [107]. Learning the relations between object parts and other contextual information, such as materials, states, and tasks, in a NN enables generalizable yet accurate reasoning of semantic grasps [62].

6.8. Reasoning about continuous actions

Semantic reasoning about actions requires the ability to perform inference on continuous data. Determining the appropriate level of abstraction for actions is challenging, as illustrated in the study of the egg cracking problem [262]. A large number of complex logic assertions are required to axiomatize the reasoning of the egg cracking action because associated continuous behaviors and events have to be discretized differently for different aspects of the problem. Instead of developing a complete symbolic representation of actions, many works explore certain aspects of action semantics. For example, semantic representation of grasps are derived from affordances and materials of object parts [62], user preferences for trajectories are based on nearby objects [108], constraint-based motion control are grounded to abstract descriptions of object parts [216], and tool-use trajectories are characterized by physical properties such as displacement and velocity [174]. Other works leverage computational frameworks that can directly perform inference on continuous data. Learning in these computational frameworks also allows structure and appropriate level of abstraction to be discovered directly from data. In [101], a Gaussian Mixture Model is used within a BN to reason about continuous variables such as grasp position and orientation. In [172], Gaussian Processes with graphical models together aid representation and reasoning of manipulation trajectories of objects. In [84,154], neural networks are used to encode manipulation trajectories in the shared semantic space with their matching language descriptions and point cloud of the target objects. In [147,148], actions are selected in novel states using LingUNets that fuse continuous representations of natural language instructions and environment observations. Executing actions in simulation is another direction for reasoning about continuous actions. Though predicting action effects and retrieving motion parameters are demonstrated in [169], the speed and fidelity of simulation-based reasoning are still open challenges.

6.9. Connecting language to tasks and actions

Natural language is an organic way to communicate with naive users, and data-driven semantic reasoning has made immense progress in leveraging unstructured natural language sources to extract knowledge in an unsupervised fashion. Such semantic knowledge tied to natural language has not only allowed robots to communicate more naturally with human users, but also understand and plan tasks, by grounding the semantic knowledge in the real world.

Language models have been used to follow action instructions [74,78,220], understand goal definitions [77,149], and ground natural language action expressions in the real world [191]. Multi-modal models trained simultaneously on vision and language can be used to localize the referred object in a scene [72,73,97,208,244,245], even if the object is very small or occluded and is referred to using a reference object [66]. In addition to understanding the human user, the robot can use natural language to answer the user's queries about the environment [136,250,251].

Pre-trained language models are hypothesized to extract generalizable knowledge and have been empirically shown to be applicable in robotics towards task planning. Some methods show that LLMs when directly applied to task planning lead to non-executable plans [94,150], but using them to pick an available action based on language-similarity [94], using a value function for predicting probability of an action being successful [95], using an interactive decision process with feedback about the environment and/or changing task goals [263], or prompting them to generate programmatic expansions of given tasks [205] lead to improved performance. They can also learn task policies over sequential state, action histories, and goal representations, and their performance drops only slightly on manufactured languages created by randomly permuting the words so that their actual meaning is lost, implying that these models can work on any sequential data, not limited to the languages it was trained on [247]. Additionally, pretrained visual-language models are used in language-conditioned navigation [50,96] and manipulation [74] to ground human instructions to visual data using the model's learned visual-language correspondence.

6.10. Enabling interaction through agent semantics

Reasoning about agent semantics is crucial for robots to collaborate with other agents successfully. Robots operating in human environments often need to fill in knowledge gaps required to interpret and execute tasks. In addition, modeling and reasoning about other agents is especially useful to enable interaction.

Interacting with other agents is a difficult task because differences in mental models result in knowledge gaps across agents, which is especially true of human–robot interaction. Several works have leveraged the diverse semantic knowledge of humans by directly incorporating human declarations into ontologies and knowledge bases [19,25,26,253]. Cognitive architectures such as [91,264] perform probabilistic logical reasoning over human dialogue to clarify user requests and perform object retrieval. A work combines a cognitive robot architecture that performs logical reasoning over human dialogue with hierarchical semantic mapping to resolve and execute natural language navigation goals [249]. In addition to incorporating declarations, works have used human–robot dialogues to capture and update human beliefs, following theory-of-mind [25,26]. In [243], natural language instructions from humans are used to label agent trajectories in many visual semantic navigation works. Other works have filled human–robot knowledge gaps when interpreting semantic commands by gathering and interpreting demonstrations of tasks from resources like online recipes, web videos and images, or user preferences to enable capabilities like inferring tools involved during tasks or preferred organizations of items [20,32,104,105,180,202,231,265]. Recognizing the value of such resources, many works have moved further to organize various agent representations into large knowledge bases for reuse across tasks or agents [21,24,27,32,104]. Clarifying questions that resolve ambiguities in human dialogue can further reduce human–robot knowledge gaps. These follow-up questions are generated by combining language with class-level knowledge [91,266] or grounding language to visual data [88,252,254,267]. In addition to clarifying questions, in [268] explanations based on commonsense knowledge are used as justifications of a robot's choice of action. The justifications help the robot user understand why a robot believes an action will succeed and correct the robot when it provides a nonsensical justification.

Aside from human agents, reasoning about robot agents, including a robot itself and other robots, allows for interaction between robot systems and the sharing of learned abilities. Prior work has looked at reasoning about the semantics of agent configuration and available hardware to define heuristics in a centralized task allocation model among multiple agents [176].

Other works have constructed shared knowledge bases that contained annotations encoding the semantics of action capabilities accounting for hardware and software, objects models, and maps [21,22]. These annotations allow a robot to reason about which actions it could perform, search for any missing object models or maps needed in order to perform the actions, and calculate the likelihood an action would succeed based on the robot's previous attempts. Recent work has further enabled collaboration between robotic agents by leveraging hardware capabilities across agents to enable collaborative task planning [233].

7. Open challenges in SR for robotics

Semantic reasoning for robotics represents a very broad and rapidly expanding area of interest. As highlighted by the discussion in the previous sections, current approaches to semantic reasoning address a wide variety of problems, under many different conditions and assumptions. Below, we highlight a number of open problems and challenges the research field faces, as well as promising directions for future work.

Benchmark, Datasets, Comparisons: Existing works within this research area must continually update datasets and benchmark tasks towards larger, more complex problems that more closely model real-world applications. Due to the broad range of problems and approaches within semantic reasoning, it is unlikely any single dataset would sufficiently cover all problems. Additionally, such datasets would need to include multiple modalities for each reasoning objective to enable testing of a broad range of approaches that integrate different knowledge sources, world representations, and computational frameworks. The development of these datasets and benchmarks for various system aspects would be a strong catalyst for promoting the maturation of this research area, as has been seen in the computer vision and natural language processing communities.

Computational Frameworks: As benchmarks mature to become more challenging, computational frameworks, which support the main infrastructure to relate semantic knowledge, will need to improve. There is yet to exist a computational framework that can balance scalability, model uncertainty, allow for complex conditioning of queries, and be highly adaptable. While there will likely continue to be trade-offs with regard to these aspects, improvements of existing approaches and development of new representations must continue to support systems tackling more challenging problems and domains. Exhaustive quantitative comparisons of the various computational frameworks in combinations with different semantic reasoning applications should be further explored to better understand which frameworks best suit particular sets of applications.

Using Multi-Modal Context: Targeting real-world tasks requires robots to reason about the contexts of tasks, such as visual and auditory queues [58], natural language commands or feedback [72,254], haptic feedback of the robot's interactions [63], and other forms of multi-sensory reasoning. The increasing scale and complexity of robotics problems will quickly make processing all multi-modal information infeasible. There are several open challenges in filtering the execution context to extract task relevant information, selecting the right abstraction level to reason about the task at hand, and other open problems to address data bottlenecks. Context will likely be used as a key feature in new ways as the complexity of problems increases.

Merging Knowledge Sources: Another challenge of semantic reasoning for robots that remains unexplored is updating and combining semantic knowledge sources. This is difficult because some sources might only express binary beliefs (e.g., there exists a relation between x and y) while others might include confidence levels for beliefs and still others might have distributions over possible beliefs. Combining such sources into a unified representation to perform inference is challenging. Additionally, the symbols and data types across knowledge sources could be distinct creating knowledge gaps or duplicated requiring disambiguation.

Life-long improvement: Life-long learning is necessary for robots to adapt in everyday environments [269]. Robots that operate outside of controlled labs require an evolving semantic representation to model new concepts, relationships between concepts, and sensory signals that ground concepts. Some works have begun to address subsets of these problems [270–272]. However, algorithms that allow a robot to learn autonomously through time about the external world, and incrementally develop a set of complex skills and knowledge is still an open area of research [273].

Explainable Reasoning: Human–robot trust is an important factor to promote long-term interaction and collaboration [274]. Explainable AI planning (XAIP) is a focus area of explainable AI, with the goal of explaining an AI's reasoning to humans in complex decision-making procedures to foster trust [275]. Research interest in XAIP has grown as autonomous agents become more capable and complex. Many open research topics remain in developing interpretable models and explainable agent reasoning.

Modeling Humans With Context: Service robots that have long-term interactions with humans will require a holistic understanding of human users. Human models that integrate task, object, and space representations are needed to enable the grounding of humans goals and preferences. Initial efforts have been made to use virtual reality simulation to test diverse assistive models on humans [276] and use crowd-sourced data to create a dataset representing longitudinal human routines [230]. However, it remains an open challenge to create and evaluate comprehensive representations of human behavior.

Reasoning Over Multiple World Representations: SRFs that can simultaneously reason over all world representations are required for real-world robotic tasks such as home rearrangement [277]. Currently, a majority of works can only reason over two or three world representations in their framework. Some recent works in agent-focused tasks such as human preference learning [140,225,227] and natural language interaction [19,26,91] are able to reason over *object*, *space*, *task* and *agent* representations. However, there are no methods that reason over all five world representations, and further study in enhancing the multi-modal reasoning capabilities of SRFs for different task domains can enable other robotics applications.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgments

This work is supported in part by NSF IIS 1564080, NSF GRFP DGE-1650044, ONR N000141612835, and NSF IIS 2112633.

References

[1] M. Ersen, E. Oztop, S. Sariel, Cognition-enabled robot manipulation in human environments: Requirements, recent work, and open problems, IEEE Robot. Autom. Mag. (2017).
[2] D. Paulius, Y. Sun, A survey of knowledge representation in service robotics, Robot. Auton. Syst. 118 (2019) 13–30.

[3] M. Beetz, R. Chatila, J. Hertzberg, F. Pecora, AI reasoning methods for robotics, in: Springer Handbook of Robotics, Springer, 2016, pp. 329–356.

[4] S. Zhang, M. Sridharan, A survey of knowledge-based sequential decision-making under uncertainty, AI Mag. 43 (2) (2022) 249–266.

[5] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 779–788.

[6] K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask R-CNN, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2961–2969.

[7] I. Kostavelis, A. Gasteratos, Semantic mapping for mobile robotics tasks: A survey, Robot. Auton. Syst. 66 (2015) 86–103.

[8] K.M. Varadarajan, M. Vincze, AfNet: The affordance network, in: K.M. Lee, Y. Matsushita, J.M. Rehg, Z. Hu (Eds.), Computer Vision, ACCV 2012, Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN: 978-3-642-37331-2, 2013, pp. 512–523.

[9] K.M. Varadarajan, M. Vincze, Afrob: The affordance network ontology for robots, in: Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on, IEEE, 2012, pp. 1343–1350.

[10] K.M. Varadarajan, Topological mapping for robot navigation using affordance features, in: Automation, Robotics and Applications (ICARA), 2015 6th International Conference on, IEEE, 2015, pp. 42–49.

[11] H. Liu, P. Singh, ConceptNet—A practical commonsense reasoning tool-kit, BT Technol J 22 (4) (2004) 211–226.

[12] P. Singh, T. Lin, E.T. Mueller, G. Lim, T. Perkins, W.L. Zhu, Open mind common sense: Knowledge acquisition from the general public, in: OTM Confederated International Conferences" on the Move to Meaningful Internet Systems", Springer, 2002, pp. 1223–1237.

[13] G.A. Miller, WordNet: A lexical database for english, Commun. ACM 38 (11) (1995) 39–41.

[14] D.B. Lenat, CYC: A large-scale investment in knowledge infrastructure, Commun. ACM 38 (11) (1995) 33–38.

[15] J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P.N. Mendes, S. Hellmann, M. Morsey, P. Van Kleef, S. Auer, et al., DBpedia–A large-scale, multilingual knowledge base extracted from Wikipedia, Semantic Web 6 (2) (2015) 167–195.

[16] M. Tenorth, M. Beetz, KnowRob—Knowledge processing for autonomous personal robots, in: Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on, IEEE, 2009, pp. 4261–4266.

[17] S. Chernova, V. Chu, A. Daruna, H. Garrison, M. Hahn, P. Khante, W. Liu, A. Thomaz, Situated Bayesian reasoning framework for robots operating in diverse everyday environments.

[18] J. Modayil, B. Kuipers, The initial development of object knowledge by a learning robot, Robot. Auton. Syst. 56 (11) (2008) 879–890.

[19] D. Nyga, S. Roy, R. Paul, D. Park, M. Pomarlan, M. Beetz, N. Roy, Grounding robot plans from natural language instructions with incomplete world knowledge, in: Conference on Robot Learning, 2018, pp. 714–723.

[20] M. Tenorth, L. Kunze, D. Jain, M. Beetz, Knowrob-map-knowledge-linked semantic object maps, in: Humanoid Robots (Humanoids), 2010 10th IEEE-RAS International Conference on, IEEE, 2010, pp. 430–435.

[21] M. Waibel, M. Beetz, J. Civera, R. d'Andrea, J. Elfring, D. Galvez-Lopez, K. Häussermann, R. Janssen, J. Montiel, A. Perzylo, et al., Roboearth, IEEE Robot. Autom. Mag. 18 (2) (2011) 69–82.

[22] L. Kunze, T. Roehm, M. Beetz, Towards semantic robot description languages, in: Robotics and Automation (ICRA), 2011 IEEE International Conference on, IEEE, 2011, pp. 5589–5595.

[23] M. Beetz, F. Bálint-Benczédi, N. Blodow, D. Nyga, T. Wiedemeyer, Z.-C. Márton, Robosherlock: Unstructured information processing for robot perception, in: Robotics and Automation (ICRA), 2015 IEEE International Conference on, IEEE, 2015, pp. 1549–1556.

[24] M. Beetz, M. Tenorth, J. Winkler, Open-EASE, in: Robotics and Automation (ICRA), 2015 IEEE International Conference on, IEEE, 2015, pp. 1983–1990.

[25] S. Lemaignan, R. Ros, L. Mösenlechner, R. Alami, M. Beetz, ORO, A knowledge management platform for cognitive architectures in robotics, in: Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on, IEEE, 2010, pp. 3548–3553.

[26] S. Lemaignan, M. Warnier, E.A. Sisbot, A. Clodic, R. Alami, Artificial cognition for social human–robot interaction: An implementation, Artificial Intelligence 247 (2017) 45–69.

[27] A. Saxena, A. Jain, O. Sener, A. Jami, D.K. Misra, H.S. Koppula, Robobrain: Large-scale knowledge engine for robots, 2014, arXiv preprint arXiv:1412.0691.

[28] M. Tenorth, U. Klank, D. Pangercic, M. Beetz, Web-enabled robots, IEEE Robot. Autom. Mag. 18 (2) (2011) 58–68.

[29] M. Stenmark, J. Malec, Describing constraint-based assembly tasks in unstructured natural language, IFAC Proc. Vol. 47 (3) (2014) 3056–3061, 19th IFAC World Congress.

[30] M. Tenorth, D. Nyga, M. Beetz, Understanding and executing instructions for everyday manipulation tasks from the World Wide Web, in: 2010 IEEE International Conference on Robotics and Automation, 2010, pp. 1486–1491.

[31] L. Denoyer, P. Gallinari, The wikipedia XML corpus, in: International Workshop of the Initiative for the Evaluation of XML Retrieval, Springer, 2006, pp. 12–19.

[32] Y. Zhu, A. Fathi, L. Fei-Fei, Reasoning about object affordances in a knowledge base representation, in: European Conference on Computer Vision, Springer, 2014, pp. 408–424.

[33] F.M. Suchanek, G. Kasneci, G. Weikum, Yago: A core of semantic knowledge, in: Proceedings of the 16th International Conference on World Wide Web, ACM, 2007, pp. 697–706.

[34] M. Daoutis, A. Loutfi, S. Coradeschi, Knowledge representation for anchoring symbolic concepts to perceptual data, in: Bridges Between the Methodological and Practical Work of the Robotics and Cognitive Systems Communities—From Sensors to Concepts. Intelligent Systems Reference Library, Springer, Citeseer, 2012.

[35] B. Min, R. Grishman, L. Wan, C. Wang, D. Gondek, Distant supervision for relation extraction with an incomplete knowledge base, in: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2013, pp. 777–782.

[36] Y. Li, J. Gao, C. Meng, Q. Li, L. Su, B. Zhao, W. Fan, J. Han, A survey on truth discovery, ACM Sigkdd Explor Newslett 17 (2) (2016) 1–16.

[37] M. Nickel, K. Murphy, V. Tresp, E. Gabrilovich, A review of relational machine learning for knowledge graphs, Proc. IEEE 104 (1) (2015) 11–33.

[38] S. Stein, S.J. McKenna, Combining embedded accelerometers with computer vision for recognizing food preparation activities, in: Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp 2013, Zurich, Switzerland, ACM, 2013.

[39] C. Lea, A. Reiter, R. Vidal, G.D. Hager, Segmental spatiotemporal CNNS for fine-grained action segmentation, in: European Conference on Computer Vision, Springer, 2016, pp. 36–52.

[40] K. Chen, N.S. Srikanth, D. Kent, H. Ravichandar, S. Chernova, Learning hierarchical task networks with preferences from unannotated demonstrations, in: J. Kober, F. Ramos, C. Tomlin (Eds.), Proceedings of the 2020 Conference on Robot Learning, in: Proceedings of Machine Learning Research, vol. 155, PMLR, 2021, pp. 1572–1581.

[41] E. Kolve, R. Mottaghi, D. Gordon, Y. Zhu, A. Gupta, A. Farhadi, AI2-THOR: An interactive 3D environment for visual AI, 2017, arXiv preprint arXiv:1712.05474.

[42] Y. Zhu, D. Gordon, E. Kolve, D. Fox, L. Fei-Fei, A. Gupta, R. Mottaghi, A. Farhadi, Visual semantic planning using deep successor representations, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 483–492.

[43] B. Trabucco, G. Sigurdsson, R. Piramuthu, G.S. Sukhatme, R. Salakhutdinov, A Simple Approach for Visual Rearrangement: 3D Mapping and Semantic Search, 2022, arXiv preprint arXiv:2206.13396.

[44] S. Srivastava, C. Li, M. Lingelbach, R. Martín-Martín, F. Xia, K.E. Vainio, Z. Lian, C. Gokmen, S. Buch, K. Liu, et al., Behavior: Benchmark for everyday household activities in virtual, interactive, and ecological environments, in: Conference on Robot Learning, PMLR, 2022, pp. 477–490.

[45] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C.L. Zitnick, Microsoft CoCo: Common objects in context, in: European Conference on Computer Vision, Springer, 2014, pp. 740–755.

[46] K. Marino, R. Salakhutdinov, A. Gupta, The more you know: Using knowledge graphs for image classification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2673–2681.

[47] K. Grauman, A. Westbury, E. Byrne, Z. Chavis, A. Furnari, R. Girdhar, J. Hamburger, H. Jiang, M. Liu, X. Liu, et al., Ego4D: Around the world in 3,000 hours of egocentric video, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 18995–19012.

[48] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, A. Gupta, R3M: A universal visual representation for robot manipulation, in: Conference on Robot Learning, 2022.

[49] A. Szot, A. Clegg, E. Undersander, E. Wijmans, Y. Zhao, J. Turner, N. Maestre, M. Mukadam, D. Chaplot, O. Maksymets, A. Gokaslan, V. Vondrus, S. Dharur, F. Meier, W. Galuba, A. Chang, Z. Kira, V. Koltun, J. Malik, M. Savva, D. Batra, Habitat 2.0: Training home assistants to rearrange their habitat, in: Advances in Neural Information Processing Systems, NeurIPS, 2021.

[50] S.Y. Gadre, M. Wortsman, G. Ilharco, L. Schmidt, S. Song, CLIP on Wheels: Zero-shot object navigation as object localization and exploration, 2022, arXiv preprint arXiv:2203.10421.

[51] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, in: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, IEEE, 2009, pp. 248–255.

[52] A.X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, et al., Shapenet: An information-rich 3D model repository, 2015, arXiv preprint arXiv:1512.03012.

[53] M. Kokic, J.A. Stork, J.A. Haustein, D. Kragic, Affordance detection for task-specific grasping using deep learning, in: 2017 IEEE-RAS 17th International Conference on Humanoid Robotics, Humanoids, IEEE, 2017, pp. 91–98.

[54] B. Yao, X. Jiang, A. Khosla, A.L. Lin, L. Guibas, L. Fei-Fei, Human action recognition by learning bases of action attributes and parts, in: Computer Vision (ICCV), 2011 IEEE International Conference on, IEEE, 2011, pp. 1331–1338.

[55] S. Song, F. Yu, A. Zeng, A.X. Chang, M. Savva, T. Funkhouser, Semantic scene completion from a single depth image, in: IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[56] P. Shah, M. Fiser, A. Faust, J.C. Kew, D. Hakkani-Tur, FollowNet: Robot navigation by following natural language directions with deep reinforcement learning, 2018, arXiv preprint arXiv:1805.06150.

[57] C. Gan, S. Zhou, J. Schwartz, S. Alter, A. Bhandwaldar, D. Gutfreund, D.L. Yamins, J.J. DiCarlo, J. McDermott, A. Torralba, et al., The ThreeDWorld transport challenge: A visually guided task-and-motion planning benchmark towards physically realistic embodied AI, in: 2022 International Conference on Robotics and Automation, ICRA, IEEE, 2022, pp. 8847–8854.

[58] C. Gan, Y. Gu, S. Zhou, J. Schwartz, S. Alter, J. Traer, D. Gutfreund, J. Tenenbaum, J. McDermott, A. Torralba, Finding fallen objects via asynchronous audio-visual integration, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022.

[59] A. Murali, W. Liu, K. Marino, S. Chernova, A. Gupta, Same object, different grasps: Data and semantic knowledge for task-oriented grasping, in: Conference on Robot Learning, 2020.

[60] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D.A. Shamma, et al., Visual genome: Connecting language and vision using crowdsourced dense image annotations, Int. J. Comput. Vis. 123 (1) (2017) 32–73.

[61] L. Weihs, M. Deitke, A. Kembhavi, R. Mottaghi, Visual room rearrangement, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 5922–5931.

[62] W. Liu, A. Daruna, S. Chernova, Cage: Context-aware grasping engine, in: 2020 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2020, pp. 2550–2556.

[63] T. Migimatsu, W. Lian, J. Bohg, S. Schaal, Symbolic State Estimation with Predicates for Contact-Rich Manipulation Tasks, in: 2022 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2022.

[64] C. Wang, D. Xu, L. Fei-Fei, Generalizable task planning through representation pretraining, in: IEEE Robotics and Automation Letters, 2022, pp. 8299–8306.

[65] S.Y. Gadre, K. Ehsani, S. Song, R. Mottaghi, Continuous Scene Representations for Embodied AI, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 14849–14859.

[66] H. Ha, S. Song, Semantic abstraction: Open-world 3D scene understanding from 2D vision-language models, in: Conference on Robot Learning, 2022.

[67] D.-A. Huang, D. Xu, Y. Zhu, A. Garg, S. Savarese, L. Fei-Fei, J.C. Niebles, Continuous relaxation of symbolic planner for one-shot imitation learning, in: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2019, pp. 2635–2642.

[68] K. Zheng, R. Chitnis, Y. Sung, G. Konidaris, S. Tellex, Towards optimal correlational object search, in: 2022 International Conference on Robotics and Automation, ICRA, IEEE, 2022, pp. 7313–7319.

[69] A. Kurenkov, R. Martín-Martín, J. Ichnowski, K. Goldberg, S. Savarese, Semantic and geometric modeling with neural message passing in 3D scene graphs for hierarchical mechanical search, in: 2021 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2021, pp. 11227–11233.

[70] O. Mees, A. Emek, J. Vertens, W. Burgard, Learning object placements for relational instructions by hallucinating scene representations, in: 2020 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2020, pp. 94–100.

[71] C. Paxton, C. Xie, T. Hermans, D. Fox, Predicting stable configurations for semantic placement of novel objects, in: Conference on Robot Learning, PMLR, 2022, pp. 806–815.

[72] P. Achlioptas, A. Abdelreheem, F. Xia, M. Elhoseiny, L. Guibas, Referit3D: Neural listeners for fine-grained 3D object identification in real-world scenes, in: European Conference on Computer Vision, Springer, 2020, pp. 422–440.

[73] J. Thomason, M. Shridhar, Y. Bisk, C. Paxton, L. Zettlemoyer, Language grounding with 3D objects, in: Conference on Robot Learning, PMLR, 2022, pp. 1691–1701.

[74] M. Shridhar, L. Manuelli, D. Fox, Cliport: What and where pathways for robotic manipulation, in: Conference on Robot Learning, PMLR, 2022, pp. 894–906.

[75] C. Lynch, P. Sermanet, Language conditioned imitation learning over unstructured data, in: Robotics: Science and Systems, 2021.

[76] S. Nair, E. Mitchell, K. Chen, S. Savarese, C. Finn, et al., Learning language-conditioned robot behavior from offline data and crowd-sourced annotation, in: Conference on Robot Learning, PMLR, 2022, pp. 1303–1315.

[77] C. Gao, J. Chen, S. Liu, L. Wang, Q. Zhang, Q. Wu, Room-and-object aware knowledge reasoning for remote embodied referring expression, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 3064–3073.

[78] Y. Qi, Z. Pan, S. Zhang, A.v.d. Hengel, Q. Wu, Object-and-action aware model for visual language navigation, in: European Conference on Computer Vision, Springer, 2020, pp. 303–317.

[79] S. Coradeschi, A. Saffiotti, An introduction to the anchoring problem, Robot. Auton. Syst. 43 (2) (2003) 85–96.

[80] S. Coradeschi, A. Loutfi, B. Wrede, A short review of symbol grounding in robotic and intelligent systems, KI-Künstliche Intell. 27 (2) (2013) 129–136.

[81] M. Tenorth, M. Beetz, Representations for robot knowledge in the KnowRob framework, Artificial Intelligence 247 (2017) 151–169.

[82] C. Galindo, A. Saffiotti, S. Coradeschi, P. Buschka, J.-A. Fernandez-Madrigal, J. González, Multi-hierarchical semantic maps for mobile robotics, in: Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on, IEEE, 2005, pp. 2278–2283.

[83] S. Vasudevan, S. Gächter, V. Nguyen, R. Siegwart, Cognitive maps for mobile robots-an object based approach, Robot. Auton. Syst. 55 (5) (2007) 359–371.

[84] J. Sung, S.H. Jin, A. Saxena, Robobarista: Object part based transfer of manipulation trajectories from crowd-sourcing in 3D pointclouds, in: Robotics Research, Springer, 2018, pp. 701–720.

[85] Z. Zeng, A. Röfer, S. Lu, O.C. Jenkins, Generalized object permanence for object retrieval through semantic linking maps.

[86] W. Yang, X. Wang, A. Farhadi, A. Gupta, R. Mottaghi, Visual semantic navigation using scene priors, in: International Conference on Learning Representations, 2019.

[87] J. Thomason, A. Padmakumar, J. Sinapov, J. Hart, P. Stone, R.J. Mooney, Opportunistic active learning for grounding natural language descriptions, in: Conference on Robot Learning, PMLR, 2017, pp. 67–76.

[88] J. Thomason, A. Padmakumar, J. Sinapov, N. Walker, Y. Jiang, H. Yedidsion, J. Hart, P. Stone, R. Mooney, Jointly improving parsing and perception for natural language commands through human-robot dialog, J. Artificial Intelligence Res. 67 (2020) 327–374.

[89] A. Padmakumar, J. Thomason, R. Mooney, Integrated learning of dialog strategies and semantic parsing, in: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, 2017, pp. 547–557.

[90] P.W. Schermerhorn, J.F. Kramer, DIARC: A testbed for natural human-robot interaction, 2006.

[91] P. Khandelwal, S. Zhang, J. Sinapov, M. Leonetti, J. Thomason, F. Yang, I. Gori, M. Svetlik, P. Khante, V. Lifschitz, et al., Bwibots: A platform for bridging the gap between AI and human–robot interaction research, Int. J. Robot. Res. 36 (5–7) (2017) 635–659.

[92] T. Frasca, B. Oosterveld, E. Krause, M. Scheutz, One-shot interaction learning from natural language instruction and demonstration, Adv. Cogn. Syst. 6 (2018) 1–18.

[93] V. Sarathy, T. Edu, B. Oosterveld, E. Krause, M. Scheutz, Learning cognitive affordances for objects from natural language instruction, in: Proceedings of the Sixth Annual Conference on Advances in Cognitive Systems, 2018.

[94] W. Huang, P. Abbeel, D. Pathak, I. Mordatch, Language models as zero-shot planners: Extracting actionable knowledge for embodied agents, in: International Conference on Machine Learning, 2022.

[95] M. Ahn, A. Brohan, N. Brown, Y. Chebotar, O. Cortes, B. David, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, et al., Do as I can, not as I say: Grounding language in robotic affordances, in: Conference on Robot Learning, 2022.

[96] A. Khandelwal, L. Weihs, R. Mottaghi, A. Kembhavi, Simple but effective: Clip embeddings for embodied AI, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 14829–14838.

[97] J. Roh, K. Desingh, A. Farhadi, D. Fox, Languagerefer: Spatial-language model for 3D visual grounding, in: Conference on Robot Learning, PMLR, 2022, pp. 1046–1056.

[98] A.Z. Ren, B. Govil, T.-Y. Yang, K.R. Narasimhan, A. Majumdar, Leveraging language for accelerated learning of tool manipulation, in: 6th Annual Conference on Robot Learning.

[99] A. Radford, J.W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al., Learning transferable visual models from natural language supervision, in: International Conference on Machine Learning, PMLR, 2021, pp. 8748–8763.

[100] W. Goodwin, S. Vaze, I. Havoutis, I. Posner, Semantically grounded object matching for robust robotic scene rearrangement, in: 2022 International Conference on Robotics and Automation, ICRA, IEEE, 2022, pp. 11138–11144.

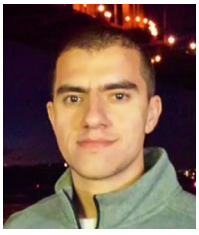
[101] D. Song, C.H. Ek, K. Huebner, D. Kragic, Task-based robot grasp planning using probabilistic inference, IEEE Trans. Robot. 31 (3) (2015) 546–561.
[102] D. Paulius, Y. Huang, R. Milton, W.D. Buchanan, J. Sam, Y. Sun, Functional object-oriented network for manipulation learning, in: Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on, IEEE, 2016, pp. 2655–2662.
[103] Z. Wu, M. Palmer, Verbs semantics and lexical selection, in: Proceedings of the 32nd Annual Meeting on Association for Computational Linguistics, Association for Computational Linguistics, 1994, pp. 133–138.
[104] D. Paulius, A.B. Jelodar, Y. Sun, Functional object-oriented network: Construction & expansion, in: 2018 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2018, pp. 5935–5941.
[105] N. Abdo, C. Stachniss, L. Spinello, W. Burgard, Robot, organize my shelves! Tidying up objects by predicting user preferences, in: 2015 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2015, pp. 1557–1564.
[106] Y. Yang, A. Guha, C. Fermüller, Y. Aloimonos, Manipulation action tree bank: A knowledge resource for humanoids, in: 2014 IEEE-RAS International Conference on Humanoid Robots, IEEE, 2014, pp. 987–992.
[107] J. Aleotti, S. Caselli, Part-based robot grasp planning from human demonstration, in: 2011 IEEE International Conference on Robotics and Automation, IEEE, 2011, pp. 4554–4560.
[108] A. Jain, B. Wojcik, T. Joachims, A. Saxena, Learning trajectory preferences for manipulators via iterative improvement, in: Advances in Neural Information Processing Systems, 2013, pp. 575–583.
[109] D. Koller, N. Friedman, Probabilistic Graphical Models: Principles and Techniques, MIT Press, 2009.
[110] B. Limketkai, L. Liao, D. Fox, Relational object maps for mobile robots, in: IJCAI, 2005, pp. 1471–1476.
[111] M. Günther, J. Ruiz-Sarmiento, C. Galindo, J. Gonzalez-Jimenez, J. Hertzberg, Context-aware 3D object anchoring for mobile robots, Robot. Auton. Syst. 110 (2018) 12–32.
[112] Z. Zeng, A. Röfer, O.C. Jenkins, Semantic linking maps for active visual object search, in: 2020 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2020, pp. 1984–1990.
[113] R. Paul, A. Barbu, S. Felshin, B. Katz, N. Roy, Temporal grounding graphs for language understanding with accrued visual-linguistic context, in: Proceedings of the 26th International Joint Conference on Artificial Intelligence, AAAI Press, 2017, pp. 4506–4514.
[114] A. Pronobis, P. Jensfelt, Large-scale semantic mapping and reasoning with heterogeneous modalities, in: Robotics and Automation (ICRA), 2012 IEEE International Conference on, IEEE, 2012, pp. 3515–3522.
[115] M. Kim, I.H. Suh, Active object search in an unknown large-scale environment using commonsense knowledge and spatial relations, Intell. Serv. Robot. 12 (4) (2019) 371–380.
[116] F. Baader, I. Horrocks, U. Sattler, Description logics, Found. Artif. Intell. 3 (2008) 135–179.
[117] W. Hwang, J. Park, H. Suh, H. Kim, I.H. Suh, Ontology-based framework of robot context modeling and reasoning for object recognition, in: International Conference on Fuzzy Systems and Knowledge Discovery, Springer, 2006, pp. 596–606.
[118] I.H. Suh, G.H. Lim, W. Hwang, H. Suh, J.-H. Choi, Y.-T. Park, Ontology-based multi-layered robot knowledge framework (OMRKF) for robot intelligence, in: Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on, IEEE, 2007, pp. 429–436.
[119] G.H. Lim, I.H. Suh, H. Suh, Ontology-based unified robot knowledge for service robots in indoor environments, IEEE Trans. Syst. Man Cybern.-A 41 (3) (2011) 492–509.
[120] L. Jacobsson, J. Malec, K. Nilsson, Modularization of skill ontologies for industrial robots, in: ISR 2016: 47st International Symposium on Robotics; Proceedings of, VDE, 2016, pp. 1–6.
[121] X. Li, S. Bilbao, T. Martín-Wanton, J. Bastos, J. Rodriguez, SWARMs ontology: A common information model for the cooperation of underwater robots, Sensors 17 (3) (2017) 569.
[122] M. Diab, A. Akbari, J. Rosell, et al., An ontology framework for physics-based manipulation planning, in: Iberian Robotics Conference, Springer, 2017, pp. 452–464.
[123] L. De Raedt, K. Kersting, Statistical relational learning, in: Encyclopedia of Machine Learning, Springer, 2011, pp. 916–924.
[124] M. Richardson, P. Domingos, Markov logic networks, Mach. Learn. 62 (1) (2006) 107–136.
[125] D. Jain, S. Waldherr, M. Beetz, Bayesian logic networks.
[126] L. Getoor, B. Taskar, Introduction to Statistical Relational Learning, MIT Press, 2007.
[127] L. De Raedt, Logical and Relational Learning, Springer Science & Business Media, 2008.
[128] D. Nyga, F. Balint-Benczedi, M. Beetz, PR2 looking at things—Ensemble learning for unstructured information processing with Markov logic networks, in: 2014 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2014, pp. 3916–3923.
[129] B. Moldovan, P. Moreno, M. van Otterlo, J. Santos-Victor, L. De Raedt, Learning relational affordance models for robots in multi-object manipulation tasks, in: Robotics and Automation (ICRA), 2012 IEEE International Conference on, IEEE, 2012, pp. 4373–4378.
[130] B. Moldovan, L.D. Raedt, Occluded object search by relational affordances, in: 2014 IEEE International Conference on Robotics and Automation, ICRA, 2014, pp. 169–174.
[131] D. Nitti, T. De Laet, L. De Raedt, Relational object tracking and learning, in: Robotics and Automation (ICRA), 2014 IEEE International Conference on, IEEE, 2014, pp. 935–942.
[132] I. Goodfellow, Y. Bengio, A. Courville, Y. Bengio, Deep Learning, Vol. 1, MIT press Cambridge, 2016.
[133] X. Wang, Y. Ye, A. Gupta, Zero-shot recognition via semantic embeddings and knowledge graphs, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 6857–6866.
[134] H. Cai, V.W. Zheng, K.C.-C. Chang, A comprehensive survey of graph embedding: Problems, techniques, and applications, IEEE Trans. Knowl. Data Eng. 30 (9) (2018) 1616–1637.
[135] M. Shridhar, X. Yuan, M.-A. Cote, Y. Bisk, A. Trischler, M. Hausknecht, ALFWorld: Aligning Text and Embodied Environments for Interactive Learning, 2022.
[136] D. Gordon, A. Kembhavi, M. Rastegari, J. Redmon, D. Fox, A. Farhadi, IQA: Visual Question Answering in Interactive Environments, 2018, pp. 4089–4098.
[137] S. Stepputtis, J. Campbell, M. Phielipp, S. Lee, C. Baral, H. Ben Amor, Language-conditioned imitation learning for robot manipulation tasks, Adv. Neural Inf. Process. Syst. 33 (2020) 13139–13150.
[138] Y. Zhu, J. Tremblay, S. Birchfield, Y. Zhu, Hierarchical planning for long-horizon manipulation with geometric and symbolic scene graphs, in: 2021 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2021, pp. 6541–6548.
[139] Z. Ravichandran, L. Peng, N. Hughes, J.D. Griffith, L. Carlone, Hierarchical representations and explicit memory: Learning effective navigation policies on 3D scene graphs using graph neural networks, in: 2022 International Conference on Robotics and Automation, ICRA, IEEE, 2022, pp. 9272–9279.
[140] I. Kapelyukh, E. Johns, My house, my rules: Learning tidying preferences with graph neural networks, in: Conference on Robot Learning, PMLR, 2022, pp. 740–749.
[141] D. Turpin, L. Wang, S. Tsogkas, S. Dickinson, A. Garg, Gift: Generalizable interaction-aware functional tool affordances without labels, in: Robotics: Science and Systems, 2021.
[142] A. Pashevich, C. Schmid, C. Sun, Episodic transformer for vision-and-language navigation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 15942–15952.
[143] W. Yuan, C. Paxton, K. Desingh, D. Fox, SORNet: Spatial object-centric representations for sequential manipulation, in: 5th Annual Conference on Robot Learning, PMLR, 2021, pp. 148–157.
[144] W. Liu, C. Paxton, T. Hermans, D. Fox, Structformer: Learning spatial structure for language-guided semantic rearrangement of novel objects, in: 2022 International Conference on Robotics and Automation, ICRA, IEEE, 2022, pp. 6322–6329.
[145] W. Liu, D. Bansal, A. Daruna, S. Chernova, Learning Instance-Level N-Ary Semantic Knowledge At Scale For Robots Operating in Everyday Environments, in: Proceedings of Robotics: Science and Systems, Virtual, 2021.
[146] D. Misra, A. Bennett, V. Blukis, E. Niklasson, M. Shatkhin, Y. Artzi, Mapping instructions to actions in 3D environments with visual goal prediction, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, 2018.
[147] V. Blukis, Y. Terme, E. Niklasson, R.A. Knepper, Y. Artzi, Learning to map natural language instructions to physical quadcopter control using simulated flight, in: Conference on Robot Learning, PMLR, 2020, pp. 1415–1438.
[148] V. Blukis, R. Knepper, Y. Artzi, Few-shot object grounding and mapping for natural language robot instruction following, in: Conference on Robot Learning, PMLR, 2021, pp. 1829–1854.
[149] V. Blukis, C. Paxton, D. Fox, A. Garg, Y. Artzi, A Persistent Spatial Semantic Representation for High-level Natural Language Instruction Execution, in: Proceedings of the 5th Conference on Robot Learning, PMLR, 2022, pp. 706–717, ISSN: 2640-3498.
[150] K. Valmeekam, A. Olmo, S. Sreedharan, S. Kambhampati, Large language models still can't plan (A benchmark for LLMs on planning and reasoning about change), 2022, arXiv preprint arXiv:2206.10498.
[151] A. Daruna, W. Liu, Z. Kira, S. Chetnova, Robocse: Robot common sense embedding, in: 2019 International Conference on Robotics and Automation, ICRA, IEEE, 2019, pp. 9777–9783.
[152] J. Thomason, J. Sinapov, R.J. Mooney, P. Stone, Guiding exploratory behaviors for multi-modal grounding of linguistic descriptions, in: Thirty-Second AAAI Conference on Artificial Intelligence, 2018.

[153] N. Fulda, N. Tibbetts, Z. Brown, D. Wingate, Harvesting common-sense navigational knowledge for robotics from uncurated text corpora, in: Conference on Robot Learning, 2017, pp. 525–534.
[154] J. Sung, I. Lenz, A. Saxena, Deep multimodal embedding: Manipulating novel objects with point-clouds, language and trajectories, in: 2017 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2017, pp. 2794–2801.
[155] R.M. Neal, Bayesian Learning for Neural Networks, Vol. 118, Springer Science & Business Media, 2012.
[156] B. Lakshminarayanan, A. Pritzel, C. Blundell, Simple and scalable predictive uncertainty estimation using deep ensembles, Adv. Neural Inf. Process. Syst. 30 (2017).
[157] J. Heinsohn, Probabilistic description logics, in: Uncertainty Proceedings 1994, Elsevier, 1994, pp. 311–318.
[158] F. Yang, Z. Yang, W.W. Cohen, Differentiable learning of logical rules for knowledge base reasoning, in: Advances in Neural Information Processing Systems, 2017, pp. 2319–2328.
[159] T. Rocktäschel, S. Riedel, End-to-end differentiable proving, in: Advances in Neural Information Processing Systems, 2017, pp. 3788–3800.
[160] W. Hamilton, P. Bajaj, M. Zitnik, D. Jurafsky, J. Leskovec, Embedding logical queries on knowledge graphs, in: Advances in Neural Information Processing Systems, 2018, pp. 2026–2037.
[161] M. Tenorth, M. Beetz, KnowRob: A knowledge processing infrastructure for cognition-enabled robots, Int. J. Robot. Res. 32 (5) (2013) 566–590.
[162] M. Tenorth, S. Profanter, F. Balint-Benczedi, M. Beetz, Decomposing cad models of objects of daily use and reasoning about their functional parts, in: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2013, pp. 5943–5949.
[163] A. Boteanu, A. St. Clair, A. Mohseni-Kabir, C. Saldanha, S. Chernova, Leveraging large-scale semantic networks for adaptive robot task learning and execution, Big Data 4 (4) (2016) 217–235.
[164] M. Thosar, C.A. Mueller, G. Jäger, J. Schleiss, N. Pulugu, R. Mallikarjun Chennaboina, S.V. Rao Jeevangekar, A. Birk, M. Pfingsthorn, S. Zug, From multi-modal property dataset to robot-centric conceptual knowledge about household objects, Front. Robot. AI 8 (2021) 87.
[165] A. Simeonov, Y. Du, A. Tagliasacchi, J.B. Tenenbaum, A. Rodriguez, P. Agrawal, V. Sitzmann, Neural descriptor fields: Se (3)-equivariant object representations for manipulation, in: 2022 International Conference on Robotics and Automation, ICRA, IEEE, 2022, pp. 6394–6400.
[166] R. Gao, Y.-Y. Chang, S. Mall, L. Fei-Fei, J. Wu, ObjectFolder: A dataset of objects with implicit visual, auditory, and tactile representations, in: Conference on Robot Learning, PMLR, 2022, pp. 466–476.
[167] C. Galindo, J.-A. Fernández-Madrigal, J. González, A. Saffiotti, Robot task planning using semantic maps, Robot. Auton. Syst. 56 (11) (2008) 955–966.
[168] Y. Bisk, R. Zellers, J. Gao, Y. Choi, et al., Piqa: Reasoning about physical commonsense in natural language, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, no. 05, 2020, pp. 7432–7439.
[169] M. Beetz, D. Beßler, A. Haidu, M. Pomarlan, A.K. Bozcuoğlu, G. Bartels, Know Rob 2.0—A 2nd generation knowledge processing framework for cognition-enabled robotic agents, in: 2018 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2018, pp. 512–519.
[170] 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, IEEE, 2012, pp. 4644–4651.
[171] L. Manuelli, W. Gao, P. Florence, R. Tedrake, KPAM: Keypoint affordances for category-level robotic manipulation, in: The International Symposium of Robotics Research, Springer, 2019, pp. 132–157.
[172] H.S. Koppula, A. Saxena, Physically grounded spatio-temporal object affordances, in: European Conference on Computer Vision, Springer, 2014, pp. 831–847.
[173] J.J. Gibson, The Ecological Approach to Visual Perception: Classic Edition, Psychology Press, 2014.
[174] Y. Zhu, Y. Zhao, S. Chun Zhu, Understanding tools: Task-oriented object modeling, learning and recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 2855–2864.
[175] L. Antanas, P. Moreno, M. Neumann, R.P. de Figueiredo, K. Kersting, J. Santos-Victor, L. De Raedt, Semantic and geometric reasoning for robotic grasping: A probabilistic logic approach, Auton. Robots 43 (6) (2019) 1393–1418.
[176] S.S. Hidayat, B.K. Kim, K. Ohba, Learning affordance for semantic robots using ontology approach, in: Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on, IEEE, 2008, pp. 2630–2636.
[177] K. Fang, Y. Zhu, A. Garg, A. Kurenkov, V. Mehta, L. Fei-Fei, S. Savarese, Learning task-oriented grasping for tool manipulation from simulated self-supervision, Int. J. Robot. Res. 39 (2–3) (2020) 202–216.
[178] D. Xu, A. Mandlekar, R. Martín-Martín, Y. Zhu, S. Savarese, L. Fei-Fei, Deep affordance foresight: Planning through what can be done in the future, in: 2021 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2021, pp. 6206–6213.
[179] I. Bozcan, S. Kalkan, COSMO: Contextualized scene modeling with Boltzmann machines, Robot. Auton. Syst. 113 (2019) 132–148.
[180] M. Tenorth, D. Nyga, M. Beetz, Understanding and executing instructions for everyday manipulation tasks from the World Wide Web, in: 2010 IEEE International Conference on Robotics and Automation, IEEE, 2010, pp. 1486–1491.
[181] M. Asada, Map building for a mobile robot from sensory data, IEEE Trans. Syst. Man Cybern. 20 (6) (1990) 1326–1336.
[182] S. Thrun, D. Fox, W. Burgard, Probabilistic mapping of an environment by a mobile robot, in: Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on, Vol. 2, IEEE, 1998, pp. 1546–1551.
[183] H. Choset, K. Nagatani, Topological simultaneous localization and mapping (SLAM): Toward exact localization without explicit localization, IEEE Trans. Robot. Autom. 17 (2) (2001) 125–137.
[184] M. Hanheide, M. Göbelbecker, G.S. Horn, A. Pronobis, K. Sjöö, A. Aydemir, P. Jensfelt, C. Gretton, R. Dearden, M. Janicek, et al., Robot task planning and explanation in open and uncertain worlds, Artificial Intelligence 247 (2017) 119–150.
[185] L. Kunze, M. Beetz, M. Saito, H. Azuma, K. Okada, M. Inaba, Searching objects in large-scale indoor environments: A decision-theoretic approach, in: 2012 IEEE International Conference on Robotics and Automation, Citeseer, 2012, pp. 4385–4390.
[186] W. Chen, S. Hu, R. Talak, L. Carlone, Leveraging large language models for robot 3D scene understanding, 2022, arXiv preprint arXiv:2209.05629.
[187] T. Migimatsu, J. Bohg, Grounding predicates through actions, in: 2022 International Conference on Robotics and Automation, ICRA, IEEE, 2022, pp. 3498–3504.
[188] K. Kase, C. Paxton, H. Mazhar, T. Ogata, D. Fox, Transferable task execution from pixels through deep planning domain learning, in: 2020 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2020, pp. 10459–10465.
[189] Y. Bisk, K.J. Shih, Y. Choi, D. Marcu, Learning interpretable spatial operations in a rich 3D blocks world, in: Thirty-Second AAAI Conference on Artificial Intelligence, 2018.
[190] R. Paul, J. Arkin, N. Roy, T. Howard, Efficient grounding of abstract spatial concepts for natural language interaction with robot manipulators, in: Robotics: Science and Systems, 2016.
[191] R. Zellers, A. Holtzman, M. Peters, R. Mottaghi, A. Kembhavi, A. Farhadi, Y. Choi, PIGLeT: Language grounding through neuro-symbolic interaction in a 3D world, in: Annual Meeting of the Association for Computational Linguistics, 2021.
[192] O. Mees, A. Emek, J. Vertens, W. Burgard, Learning object placements for relational instructions by hallucinating scene representations, in: 2020 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2020, pp. 94–100.
[193] M. Janner, K. Narasimhan, R. Barzilay, Representation learning for grounded spatial reasoning, Trans. Assoc. Comput. Linguist. 6 (2018) 49–61.
[194] R. Kartmann, D. Liu, T. Asfour, Semantic scene manipulation based on 3D spatial object relations and language instructions, in: 2020 IEEE-RAS 20th International Conference on Humanoid Robots, Humanoids, IEEE, 2021, pp. 306–313.
[195] Z. Zeng, Z. Zhou, Z. Sui, O.C. Jenkins, Semantic robot programming for goal-directed manipulation in cluttered scenes, in: 2018 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2018, pp. 7462–7469.
[196] S. Tuli, R. Bansal, R. Paul, et al., ToolTango: Common sense generalization in predicting sequential tool interactions for robot plan synthesis, in: International Joint Conference on Artificial Intelligence, 2021.
[197] A. Rosinol, A. Gupta, M. Abate, J. Shi, L. Carlone, 3D dynamic scene graphs: Actionable spatial perception with places, objects, and humans, in: Robotics: Science and Systems, 2020.
[198] S. Tan, W. Xiang, H. Liu, D. Guo, F. Sun, Multi-agent Embodied Question Answering in Interactive Environments, in: A. Vedaldi, H. Bischof, T. Brox, J.-M. Frahm (Eds.), Computer Vision, ECCV 2020, in: Lecture Notes in Computer Science, Springer International Publishing, Cham, ISBN: 9783030586010, 2020, pp. 663–678.
[199] L.P. Kaelbling, T. Lozano-Pérez, Hierarchical task and motion planning in the now, in: Robotics and Automation (ICRA), 2011 IEEE International Conference on, IEEE, 2011, pp. 1470–1477.
[200] C.R. Garrett, R. Chitnis, R. Holladay, B. Kim, T. Silver, L.P. Kaelbling, T. Lozano-Pérez, Integrated task and motion planning, Annu. Rev. Control Robot. Autonom. Syst. 4 (2021) 265–293.
[201] D. Xu, S. Nair, Y. Zhu, J. Gao, A. Garg, L. Fei-Fei, S. Savarese, Neural task programming: Learning to generalize across hierarchical tasks, in: 2018 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2018, pp. 1–8.

[202] Y. Yang, Y. Li, C. Fermüller, Y. Aloimonos, Robot learning manipulation action plans by"watching" unconstrained videos from the World Wide Web, in: AAAI, 2015, pp. 3686–3693.
[203] N. Krüger, C. Geib, J. Piater, R. Petrick, M. Steedman, F. Wörgötter, A. Ude, T. Asfour, D. Kraft, D. Omrčen, et al., Object–action complexes: Grounded abstractions of sensory–motor processes, Robot. Auton. Syst. 59 (10) (2011) 740–757.
[204] S. Zhang, M. Sridharan, M. Gelfond, J. Wyatt, Towards an architecture for knowledge representation and reasoning in robotics, in: International Conference on Social Robotics, Springer, 2014, pp. 400–410.
[205] I. Singh, V. Blukis, A. Mousavian, A. Goyal, D. Xu, J. Tremblay, D. Fox, J. Thomason, A. Garg, ProgPrompt: Generating situated robot task plans using large language models, 2022, arXiv preprint arXiv:2209.11302.
[206] G.E. Fainekos, H. Kress-Gazit, G.J. Pappas, Temporal logic motion planning for mobile robots, in: Proceedings of the 2005 IEEE International Conference on Robotics and Automation, IEEE, 2005, pp. 2020–2025.
[207] Y. Hristov, D. Angelov, M. Burke, A. Lascarides, S. Ramamoorthy, Disentangled relational representations for explaining and learning from demonstration, in: Conference on Robot Learning, PMLR, 2020, pp. 870–884.
[208] A. Das, S. Datta, G. Gkioxari, S. Lee, D. Parikh, D. Batra, Embodied question answering, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2018, pp. 2054–2063.
[209] C. Paxton, Y. Bisk, J. Thomason, A. Byravan, D. Foxl, Prospection: Interpretable plans from language by predicting the future, in: 2019 International Conference on Robotics and Automation, ICRA, 2019, pp. 6942–6948.
[210] B. Ichter, P. Sermanet, C. Lynch, Broadly-exploring, local-policy trees for long-horizon task planning, in: CoRL, 2021.
[211] M. Beetz, L. Mösenlechner, M. Tenorth, CRAM—A cognitive robot abstract machine for everyday manipulation in human environments, in: 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2010, pp. 1012–1017.
[212] D.-A. Huang, S. Nair, D. Xu, Y. Zhu, A. Garg, L. Fei-Fei, S. Savarese, J.C. Niebles, Neural task graphs: Generalizing to unseen tasks from a single video demonstration, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 8565–8574.
[213] D. Xu, R. Martín-Martín, D.-A. Huang, Y. Zhu, S. Savarese, L.F. Fei-Fei, Regression planning networks, Adv. Neural Inf. Process. Syst. 32 (2019).
[214] C. Galindo, J.-A. Fernández-Madrigal, J. González, A. Saffiotti, Robot task planning using semantic maps, Robot. Auton. Syst. 56 (11) (2008) 955–966.
[215] A. Curtis, T. Silver, J.B. Tenenbaum, T. Lozano-Pérez, L. Kaelbling, Discovering state and action abstractions for generalized task and motion planning, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 36, 2022, pp. 5377–5384, Issue: 5.
[216] M. Tenorth, G. Bartels, M. Beetz, Knowledge-based specification of robot motions, in: ECAI, 2014, pp. 873–878.
[217] G. Bartels, I. Kresse, M. Beetz, Constraint-based movement representation grounded in geometric features, in: 2013 13th IEEE-RAS International Conference on Humanoid Robots, Humanoids, IEEE, 2013, pp. 547–554.
[218] T. McMahon, O.C. Jenkins, N. Amato, Affordance wayfields for task and motion planning, in: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2018, pp. 2955–2962.
[219] S. Thompson, L.P. Kaelbling, T. Lozano-Perez, Shape-Based Transfer of Generic Skills, in: 2021 IEEE International Conference on Robotics and Automation, ICRA, (ISSN: 2577-087X) 2021, pp. 5996–6002.
[220] O. Mees, L. Hermann, E. Rosete-Beas, W. Burgard, CALVIN: A Benchmark for Language-Conditioned Policy Learning for Long-Horizon Robot Manipulation Tasks, IEEE Robot. Autom. Lett. (2022) Publisher: IEEE.
[221] P. Sharma, B. Sundaralingam, V. Blukis, C. Paxton, T. Hermans, A. Torralba, J. Andreas, D. Fox, Correcting robot plans with natural language feedback, in: Robotics: Science and Systems, 2022.
[222] C. Breazeal, M. Berlin, A. Brooks, J. Gray, A.L. Thomaz, Using perspective taking to learn from ambiguous demonstrations, Robot. Auton. Syst. 54 (5) (2006) 385–393.
[223] X. Puig, T. Shu, S. Li, Z. Wang, Y.-H. Liao, J.B. Tenenbaum, S. Fidler, A. Torralba, Watch-And-Help: A challenge for social perception and human-AI collaboration, in: International Conference on Learning Representations, 2020.
[224] R. Liu, X. Zhang, J. Webb, S. Li, Context-specific intention awareness through web query in robotic caregiving, in: 2015 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2015, pp. 1962–1967.
[225] V. Jain, Y. Lin, E. Undersander, Y. Bisk, A. Rai, Transformers are adaptable task planners, in: Conference on Robot Learning, 2022.
[226] A. Jonnavittula, D.P. Losey, I know what you meant: Learning human objectives by (under) estimating their choice set, in: 2021 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2021, pp. 2747–2753.
[227] R. Shah, D. Krasheninnikov, Preferences implicit in the state of the world, in: International Conference on Learning Representations, ICLR, 2019.
[228] R. Shah, N. Gundotra, P. Abbeel, A. Dragan, On the feasibility of learning, rather than assuming, human biases for reward inference, in: K. Chaudhuri, R. Salakhutdinov (Eds.), Proceedings of the 36th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 97, PMLR, 2019, pp. 5670–5679.
[229] S. Reddy, S. Levine, A. Dragan, Assisted perception: Optimizing observations to communicate state, in: Conference on Robot Learning, PMLR, 2021, pp. 748–764.
[230] M. Patel, S. Chernova, Proactive robot assistance via spatio-temporal object modeling, in: 6th Annual Conference on Robot Learning, 2022.
[231] H.S. Koppula, A. Saxena, Anticipating human activities using object affordances for reactive robotic response, IEEE Trans. Pattern Anal. Mach. Intell. 38 (1) (2016) 14–29.
[232] Y. Jiang, M. Lim, A. Saxena, Learning object arrangements in 3D scenes using human context, in: Proceedings of the 29th International Coference on International Conference on Machine Learning, 2012, pp. 907–914.
[233] A. Munawar, G. De Magistris, T.-H. Pham, D. Kimura, M. Tatsubori, T. Moriyama, R. Tachibana, G. Booch, Maestrob: A robotics framework for integrated orchestration of low-level control and high-level reasoning, in: 2018 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2018, pp. 527–534.
[234] K. Mo, Y. Qin, F. Xiang, H. Su, L. Guibas, O2O-Afford: Annotation-free large-scale object-object affordance learning, in: Conference on Robot Learning, PMLR, 2022, pp. 1666–1677.
[235] P. Abelha, F. Guerin, Transfer of tool affordance and manipulation cues with 3D vision data, 2017, arXiv preprint arXiv:1710.04970.
[236] H. Wu, G.S. Chirikjian, Can I pour into it? robot imagining open containability affordance of previously unseen objects via physical simulations, IEEE Robot. Autom. Lett. 6 (1) (2020) 271–278.
[237] L. Kunze, C. Burbridge, M. Alberti, A. Thippur, J. Folkesson, P. Jensfelt, N. Hawes, Combining top-down spatial reasoning and bottom-up object class recognition for scene understanding, in: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2014, pp. 2910–2915.
[238] D. Pangercic, M. Tenorth, D. Jain, M. Beetz, Combining perception and knowledge processing for everyday manipulation, in: Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on, IEEE, 2010, pp. 1065–1071.
[239] A. Pal, C. Nieto-Granda, H.I. Christensen, DEDUCE: Diverse scene detection methods in unseen challenging environments, in: Intelligent Robots and Systems (IROS), 2019 IEEE/RSJ International Conference on, IEEE, 2019.
[240] Y. Wu, Y. Wu, A. Tamar, S. Russell, G. Gkioxari, Y. Tian, Bayesian relational memory for semantic visual navigation, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 2769–2779.
[241] H. Wang, W. Wang, W. Liang, C. Xiong, J. Shen, Structured scene memory for vision-language navigation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 8455–8464.
[242] C.-Y. Ma, Z. Wu, G. AlRegib, C. Xiong, Z. Kira, The regretful agent: Heuristic-aided navigation through progress estimation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 6732–6740.
[243] J. Gu, E. Stefani, Q. Wu, J. Thomason, X. Wang, Vision-and-language navigation: A survey of tasks, methods, and future directions, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2022, pp. 7606–7623.
[244] A. Moudgil, A. Majumdar, H. Agrawal, S. Lee, D. Batra, Soat: A scene-and object-aware transformer for vision-and-language navigation, Adv. Neural Inf. Process. Syst. 34 (2021) 7357–7367.
[245] Y. Wu, Y. Wu, G. Gkioxari, Y. Tian, Building generalizable agents with a realistic and rich 3D environment, 2018, arXiv preprint arXiv:1801.02209.
[246] S. Chernova, V. Chu, A. Daruna, H. Garrison, M. Hahn, P. Khante, W. Liu, A. Thomaz, Situated Bayesian reasoning framework for robots operating in diverse everyday environments, in: International Symposium on Robotics Research, ISRR, 2017.
[247] S. Li, X. Puig, Y. Du, C. Wang, E. Akyurek, A. Torralba, J. Andreas, I. Mordatch, Pre-trained language models for interactive decision-making, in: NeurIPS, 2022.
[248] D. Shah, P. Xu, Y. Lu, T. Xiao, A. Toshev, S. Levine, B. Ichter, Value function spaces: Skill-centric state abstractions for long-horizon reasoning, in: International Conference on Learning Representations, 2022.
[249] T. Williams, C. Johnson, M. Scheutz, B. Kuipers, A Tale of Two Architectures: A Dual-Citizenship Integration of Natural Language and the Cognitive Map, in: Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems, AAMAS '17, International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 2017, pp. 1360–1368.
[250] A. Das, F. Carnevale, H. Merzic, L. Rimell, R. Schneider, J. Abramson, A. Hung, A. Ahuja, S. Clark, G. Wayne, F. Hill, Probing Emergent Semantics in Predictive Agents via Question Answering, in: Proceedings of the 37th International Conference on Machine Learning, PMLR, 2020, pp. 2376–2391.

[251] L. Yu, X. Chen, G. Gkioxari, M. Bansal, T.L. Berg, D. Batra, Multi-target embodied question answering, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 6309–6318.
[252] J. Hatori, Y. Kikuchi, S. Kobayashi, K. Takahashi, Y. Tsuboi, Y. Unno, W. Ko, J. Tan, Interactively picking real-world objects with unconstrained spoken language instructions, in: 2018 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2018, pp. 3774–3781.
[253] S.H. Paplu, M.N.I. Arif, K. Berns, Utilizing semantic and contextual information during human-robot interaction, in: 2021 IEEE International Conference on Development and Learning, ICDL, IEEE, 2021, pp. 1–6.
[254] F.I. Doğan, I. Torre, I. Leite, Asking Follow-Up Clarifications to Resolve Ambiguities in Human-Robot Conversation, in: Proceedings of the 2022 ACM/IEEE International Conference on Human-Robot Interaction, 2022, pp. 461–469.
[255] C.R. Qi, H. Su, K. Mo, L.J. Guibas, Pointnet: Deep learning on point sets for 3D classification and segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 652–660.
[256] M. Wächter, S. Schulz, T. Asfour, E. Aksoy, F. Wörgötter, R. Dillmann, Action sequence reproduction based on automatic segmentation and object-action complexes, in: Humanoid Robots (Humanoids), 2013 13th IEEE-RAS International Conference on, IEEE, 2013, pp. 189–195.
[257] A. Cocora, K. Kersting, C. Plagemann, W. Burgard, L. De Raedt, Learning relational navigation policies, in: 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2006, pp. 2792–2797.
[258] J. Hoffmann, B. Nebel, The FF planning system: Fast plan generation through heuristic search, J. Artificial Intelligence Res. 14 (2001) 253–302.
[259] T. Silver, A. Athalye, J.B. Tenenbaum, T. Lozano-Pérez, L.P. Kaelbling, Learning neuro-symbolic skills for bilevel planning, in: 6th Annual Conference on Robot Learning.
[260] H. Dang, P.K. Allen, Semantic grasping: Planning robotic grasps functionally suitable for an object manipulation task, in: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2012, pp. 1311–1317.
[261] M. Hjelm, C.H. Ek, R. Detry, D. Kragic, Learning human priors for task-constrained grasping, in: International Conference on Computer Vision Systems, Springer, 2015, pp. 207–217.
[262] L. Morgenstern, Mid-sized axiomatizations of commonsense problems: A case study in egg cracking, Studia Logica 67 (3) (2001) 333–384.
[263] W. Huang, F. Xia, T. Xiao, H. Chan, J. Liang, P. Florence, A. Zeng, J. Tompson, I. Mordatch, Y. Chebotar, P. Sermanet, N. Brown, T. Jackson, L. Luu, S. Levine, K. Hausman, B. Ichter, Inner monologue: Embodied reasoning through planning with language models, 2022, ArXiv Preprint arXiv:2207.05608.
[264] S. Zhang, P. Stone, CORPP: Commonsense reasoning and probabilistic planning, as applied to dialog with a mobile robot, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 29, no. 1, 2015.
[265] K. Ramirez-Amaro, M. Beetz, G. Cheng, Automatic segmentation and recognition of human activities from observation based on semantic reasoning, in: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2014, pp. 5043–5048.
[266] F. Williams, P. Suresh, M. Scheutz, M. Beetz, Dempster-Shafer theoretic resolution of referential ambiguity, Auton. Robots 43 (2) (2019) 389–414.
[267] R. Deits, S. Tellex, P. Thaker, D. Simeonov, T. Kollar, N. Roy, Clarifying commands with information-theoretic human-robot dialog, J. Hum.-Robot Interact. 2 (2) (2013) 58–79.
[268] A. Daruna, D. Das, S. Chernova, Explainable knowledge graph embedding: Inference reconciliation for knowledge inferences supporting robot actions, in: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2022.
[269] S. Thrun, T.M. Mitchell, Lifelong robot learning, Robot. Auton. Syst. 15 (1–2) (1995) 25–46.
[270] A. Daruna, M. Gupta, M. Sridharan, S. Chernova, Continual learning of knowledge graph embeddings, IEEE Robot. Autom. Lett. 6 (2) (2021) 1128–1135.
[271] B. Irfan, A. Ramachandran, S. Spaulding, S. Kalkan, G.I. Parisi, H. Gunes, Lifelong learning and personalization in long-term human-robot interaction (leap-hri), in: Companion of the 2021 ACM/IEEE International Conference on Human-Robot Interaction, 2021, pp. 724–727.
[272] A. Logacjov, M. Kerzel, S. Wermter, Learning then, learning now, and every second in between: Lifelong learning with a simulated humanoid robot, Front. Neurorobotics (2021) 78.
[273] T. Lesort, V. Lomonaco, A. Stoian, D. Maltoni, D. Filliat, N. Díaz-Rodríguez, Continual learning for robotics: Definition, framework, learning strategies, opportunities and challenges, Inf. Fusion 58 (2020) 52–68.
[274] J. Zhu, A. Liapis, S. Risi, R. Bidarra, G.M. Youngblood, Explainable AI for designers: A human-centered perspective on mixed-initiative co-creation, in: 2018 IEEE Conference on Computational Intelligence and Games, CIG, IEEE, 2018, pp. 1–8.
[275] T. Chakraborti, S. Sreedharan, S. Kambhampati, The emerging landscape of explainable automated planning & decision making, in: Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence, 2021, pp. 4803–4811.
[276] Z. Erickson, Y. Gu, C.C. Kemp, Assistive vr gym: Interactions with real people to improve virtual assistive robots, in: 2020 29th IEEE International Conference on Robot and Human Interactive Communication, RO-MAN, IEEE, 2020, pp. 299–306.
[277] D. Batra, A.X. Chang, S. Chernova, A.J. Davison, J. Deng, V. Koltun, S. Levine, J. Malik, I. Mordatch, R. Mottaghi, et al., Rearrangement: A challenge for embodied ai, 2020, arXiv preprint arXiv:2011.01975.

Weiyu Liu is a Ph.D. student at Georgia Institute of Technology, where he works in the Robot Autonomy and Interactive Learning (RAIL) lab and is advised by Professor Sonia Chernova. He received his undergraduate degree in Electrical Engineering from Georgia Institute of Technology in 2017. His research interest is on semantic reasoning for robotic systems.



Angel Daruna is a Ph.D. student in the Institute for Robotics and Intelligent Machines at the Georgia Institute of Technology and a graduate researcher in the Robot Autonomy and Interactive Learning (RAIL) lab. He received his B.S. degree from the Georgia Institute of Technology, Atlanta, GA, in 2016. His research interests include robotics, knowledge representations and reasoning, and machine learning.

Maithili Patel is a Robotics Ph.D. student at Georgia Institute of Technology, where she is a part of the Robot Autonomy and Interactive Learning (RAIL) lab advised by Professor Sonia Chernova. She received her M.S. degree from University of Michigan in 2019, where she was advised by Professor Chad Jenkins, and her B.Tech. degree from Indian Institute of Technology, Bombay, in 2017. Her research focus is on enabling proactivity in assistive robots by understanding and predicting the activities, needs and preferences of human users.

Kartik Ramachandruni is a Ph.D. student at the Georgia Institute of Technology, where he works in the Robot Autonomy and Interactive Learning (RAIL) lab and is advised by Professor Sonia Chernova. He received his undergraduate degree in 2018 from the Indian Institute of Technology in Jodhpur, India, after which he worked for two years as a Robotics Researcher at the TCS Research & Innovation Lab in Bangalore, India, before starting his Ph.D. His research interests include robotics, task planning, semantic reasoning, and machine learning.

Sonia Chernova is an Associate Professor in the School of Interactive Computing at Georgia Tech, where she directs the Robot Autonomy and Interactive Learning research lab. Her research spans semantic reasoning, human–robot interaction, interactive machine learning and cloud robotics, with the focus on developing robots that are able to effectively operate in human environments. She is the recipient of the NSF CAREER, ONR Young Investigator, and NASA Early Career Faculty awards.