Learning to Recognize Novel Objects in One Shot through Human-Robot Interactions in Natural Language Dialogues

Evan Krause
HRI Laboratory
Tufts University
200 Boston Ave
Medford, MA 02155, USA
evan.krause@tufts.edu

Michael Zillich
Inst. for Automation and Control
Technical University Vienna
Gusshausstr 27-29/E376
1040 Vienna, Austria
zillich@acin.tuwien.ac.at

Thomas Williams
HRI Laboratory
Tufts University
200 Boston Ave
Medford, MA 02155, USA
thomas.e.williams@tufts.edu

Matthias Scheutz
HRI Laboratory
Tufts University
200 Boston Ave
Medford, MA 02155, USA
matthias.scheutz@tufts.edu

Abstract

Being able to quickly and naturally teach robots new knowledge is critical for many future open-world human-robot interaction scenarios. In this paper we present a novel approach to using natural language context for one-shot learning of visual objects, where the robot is immediately able to recognize the described object. We describe the architectural components and demonstrate the proposed approach on a robotic platform in a proof-of-concept evaluation.

Introduction

Data-driven techniques have made rapid advances in automatic knowledge acquisition in various areas in robotics, in particular, machine vision, where new objects, scenes and activities can be learned from large image and movie databases (Fergus et al. 2005; Leibe, Leonardis, and Schiele 2008; Wohlkinger et al. 2012). Two core assumptions of data-driven methods are that (1) data sets are available for training (Fergus, Weiss, and Torralba 2009) and that (2) training is not time-critical. However, neither assumption is typically met in "open-world" scenarios such as *Urban Search and Rescue missions*, where robots must be able to acquire new knowledge *quickly, on the fly, during task performance*. Hence, we need to augment data-driven methods with other methods that allow for online learning from possibly only a few examplars.

One-shot learning is a special case of such a method where an agent can learn something new from just one exposure. For example, a typical version of visual one-shot learning would entail seeing a new object (maybe together with a linguistic label) and then being able to recognize novel instances of the same object type (e.g., from different angles and in different poses). Unfortunately, most approaches to one-short object learning are very limited with respect to the allowable teaching inputs (e.g., they are typically unimodal) and often require multiple trials (e.g., (Fei-Fei, Fergus, and Perona 2006; Lake, Salakhutdinov, and Tenenbaum 2012)). Recent integrated approaches to one-shot learning, however, are starting to address these problems by utilizing contextual information available in a cognitive architecture (e.g., (Mohan et al. 2012; Cantrell et al. 2012)).

In this paper, we will build on the successes of these recent integrated approaches and show how deep interactions between vision and natural language processing algorithms can enable one-shot object learning from natural language descriptions. By "one-shot object learning" we intend that the robot needs to hear a "sufficiently specified object description" (which we will make more precise below) *only once* before it is able to recognize the object in its environment in different contexts and poses. The key idea is to exploit structured representations in object descriptions to first infer intended objects, their parts and structured mereological relationships and to then build structured visual representations together with visual processing schemes that allow for object recognition in different contexts and poses.

Motivation and Background

The human visual system seems to use multiple representations and processing schemes to detect and recognize perceivable objects (e.g., (Logothetis and Sheinberg 1996)). At the very least, there seem to be two major systems that differ significantly in both their representations and the processes operating on them: (1) a system that represents objects by combining multiple views or aspects of the object, and (2) another system seems to represent objects in a compositional way, in terms of objects parts and their spatial, mereological relationships. The differences in representation have important consequences for how perceptions are processed. In the first system, representations of different object views are matched directly against parts of a scene, whereas in the second different object parts have to be recognized together with their spatial relationships. Thus, while the former does not require notions of "primitive object" and "spatial relationship" as object views or aspects are processed in a holistic fashion, the lack of a distinction between object parts and how they are composed to form composite objects also prevents it from being able to recognize the compositional structure of objects and their parts. As a result, unless two objects are similar in their appearance, the process cannot detect structural similarities (e.g., that two objects are containers with handles on them). The latter approach allows for object specifications in terms of structural features (parts and their relationships) and can exploit deep structural similarities among objects that differ in their visual appearance.

The difference in functional capability comes at a price:

direct matching is faster and requires fewer computational resources compared to structural processing. On the other hand, structural processing can lead to more compact representations of objects (in terms of structural features that are more abstract and succinctly represented than different object views). Most importantly for our purposes here, structural processing allows for natural language descriptions of object representations as long as there are corresponding natural language words for the involved primitive objects and their relationships, while view-based object processing only allows for "atomic" natural language labels (which do not reveal the structure of the object). Moreover, humans are very good at learning new object types based on a single natural language description, especially when the item to be learned can be broken down into previously known parts (Lake, Salakhutdinov, and Tenenbaum 2012).

Both approaches to object representation and recognition have been pursued in the machine vision communities, each with varying success. Most methods for recognizing object categories rely on sufficiently large sets of training views or training exemplars to support robust statistical inference methods. In fully supervised learning schemes these exemplars are provided as manually labeled images, e.g., (Leibe, Leonardis, and Schiele 2008), which quickly becomes unmanageable with an increasing number of categories.

Using prior information that is common to many categories (Miller, Matsakis, and Viola 2000) reduces the number of required training exemplars. Moreover, describing objects by their attributes allows for a much richer description of a scene than just naming objects. (Ferrari and Zisserman 2007) present a system for learning object attributes, such as "striped" in a semi-supervised manner from large sets of training data. (Farhadi et al. 2009) then use combinations of attributes to describe objects, such as "furry with four legs", or identify unusual attributes such as "spotty dog". This allows for learning new object classes with few examples, as objects share attributes, but learning attributes still requires large amounts of labeled training data. In a related approach (Hwang, Sha, and Grauman 2011) train their classifier jointly over attributes and objects classes for improved robustness in the attribute naming as well as classification task, again learning from large datasets of thousands of labeled images. All the above approaches treat objects as bounding boxes in 2D images, not taking into account spatial composition of objects (except in some cases implicitly within some learned features).

These statistical learning approaches require large amounts of training data. To this end, some authors use very large databases available on the internet, e.g., for 2D images (Fergus et al. 2005; Fergus, Weiss, and Torralba 2009) or 3D models (Wohlkinger et al. 2012). These approaches are semi-supervised. Only labels (search queries) are given, and care has to be taken when taking the often noisy search results for training. Moving from batch to incremental learning schemes (Fei-Fei, Fergus, and Perona 2007) makes such approaches more suitable for robotics applications. Also, by using the robot as an active observer, able to obtain its own training examples, automatic object discovery approaches can search the environment for recurring object-like structures (Kang, Hebert, and Kanade 2011; Herbst, Ren, and Fox 2011; Karpathy, Miller, and Fei-Fei 2013). Interactions then individuates single objects (Fitzpatrick et al. 2003) and in-hand rotation allows to obtain a large set of different object views (Krainin et al. 2011).

But what if a robot has to learn a new object quickly during task performance? This one-shot learning is comparatively easy for specific object instances (Özuysal et al. 2009; Collet et al. 2009), but considerably more challenging for categories (Bart and Ullman 2005; Fei-Fei, Fergus, and Perona 2006; Salakhutdinov, Tenenbaum, and Torralba 2012). User interaction and dialogue are used in (Skocaj et al. 2011) to learn about objects, but the setting is limited to table tops.

Describing objects explicitly in terms of structural features and their relations requires identifying and modeling object parts and their attributes. The approach by (Varadarajan and Vincze 2012) uses Geons to model object parts such as handles, and relations between them to describe common affordances, such as support or graspability. Approaches like (Ückermann, Haschke, and Ritter 2012; Katz et al. 2013; Richtsfeld et al. 2014) perform scene segmentation from RGB-D data at a slightly lower level, leading to objects defined in terms of groups of parametric surfaces, and allow a broader range of objects to be modeled. What makes these structural approaches appealing is that they abstract visual data to a level where meaningful attributes in terms of language ("the *top side*") as well as robot tasks and capabilities ("grasp the *handle*") can be defined.

Requirements for Language-Guided Visual One-Shot Learning

Visually perceivable objects can be described in natural language in a variety of ways that capture *object categories*, *object parts*, *surface patterns and symbols*, *object characteristics*, *spatial and mereological relations*, and others. For example, using adjectives and nouns to refer to objects and object parts, and using prepositions to refer to spatial and mereological relations among objects and object parts, a mug might be described as "cylindrical container with a round handle attached on one side". "Cylindrical" and "round" are adjectives that refer to visually perceivable properties of objects while "container" and "handle" are nouns referring to object parts (that also have certain properties such as "being able to contain items" and "affording to be grasped for pick-up"). Moreover, "attached" indicates that the parts are connected and "on" refers to a spatial relationship further specifying the attachment. Another description using verbs might get at the same content: "cylindrical container that can be picked up by its handle" where the handle's attachment to the object is implied via the action specification "can be picked up". Clearly, the latter example provides richer information about mugs, but also requires more reasoning capabilities for explicating the implied properties and is much more complex to realize computationally. Thus, to make the project feasible, we restrict the natural language side to simple noun phrase descriptions of objects that do not involve complex embedded clauses or stretch over multiple sentences (proving incremental descriptions), even though

the ultimate goal is to allow for that kind of variety.

To be able make sense of object descriptions, the vision system must be able to effect a general mapping from structured linguistic descriptions to hierarchical object descriptions that capture the types and relationships of object parts referred to in the natural language description. The vision system can then utilize this mapping to build new internal representations of previously unknown objects and will be able to recognize them as long as it has the capabilities to recognize object parts and their mereological relationships. I.e., whatever hierarchical representation the vision system developed based on the natural language description, it eventually has to "ground out" in perceptual categories for which the vision system has *detectors* and in *spatial and mereological relationships* among those categories which the system can determine. These perceptual categories can either be already learned complex objects, or they might be "primitive objects" that are recognized in their entirety (e.g., either using built-in geometric models as is common for basic shapes like "cylinders" or using statistical models trained on a large data set as could be done for "handle"). Note that often even basic shapes do not fit exactly into a primitive shape category (e.g., an orange is roughly spherical, cups are roughly cylindrical), it is critical for vision processes to be flexible enough to account for "approximate fitting" of primitive objects (e.g., object shapes).

Beyond the 3-D nature of objects, a variety of 2-D *surface patterns* and textures can also be described in natural language. These range from simple patterns such as crosses, to complex patterns like product labels (e.g., the label on a Coke can) or special symbols (such as the symbol for biohazards), and also include textures such as checkered or plaid. Surface pattern descriptors can be used in combination with object categories and object parts to further specify the details of an object (from a visual processing perspective, a Coke can is a can with the Coke label texture on it).

Notice that many object descriptions are also true of textures. The expression "red cross", for example, could equally refer to a 3-D object and a 2-D surface pattern. Hence, the intended meaning (object vs. surface pattern) has to be determined either through explicit instruction or from instruction together with perceivable context; and if it cannot be determined, the robot might have to use additional strategies to disambiguate the meaning (e.g., through discourse with the interlocutor). Conversely, it is also the case that multiple natural language descriptions can be used to refer to a single object property. Hence, the natural language and vision systems need to be able to handle this many-to-one mapping from descriptions to visual representations.

In addition to nouns referring to atomic or complex object types, descriptions of object characteristics are used to refer to visually perceivable properties such as color, size, symmetry, height, and rigidity, just to name a few. Such descriptions are often expressed via adjectives in noun phrases to further specify object categories, parts, and surface patterns. Moreover, *spatial relations* referring to constituent parts of an object are often used in noun phrases to describe the relationship among parts (including mereological relationships among parts and subparts), as in the example "cylindrical container with a round handle attached on one side". Being able to use spatial relations among object parts allows great flexibility in object descriptions; in particular, it enables the linguistic description of a decomposition of the object which can be directly used in the vision system for hierarchical object representations. These spatial relationships can hold between object categories, object parts, and surface patterns.

Descriptions of objects can also contain modifiers that are not inherent descriptions of the object itself, but serve to disambiguate a particular object from similar ones in the scene (e.g., "the cup on the left"), or provide attention cues to speed up detection (e.g., "the book on the top shelf"). The vision system must ignore these additional descriptions as they are not constitutive of the object, but are used to single out an object in the current context. Since the general problem of distinguishing what parts of a referential expression belong to the *object description proper* vs. serve the purpose of *object identification* is extremely difficult (as it might require significant background knowledge about object types, uses, context, etc.), we restrict our investigation to object descriptions without additional identifying information.

Natural Language and Vision Integration

To enable one-shot natural language-guided object learning, the natural language and vision systems of the robotic architecture have to be deeply integrated, allowing for the exchange of information about objects, objects parts, and spatial relations that may not have an established meaning yet. For example, the natural language understanding (NLU) system might attempt to resolve a noun phrase (e.g., "a red cross") to an object in the visual scene whose meaning is ambiguous from the vision system's perspective, because it could denote a 3-D object and a 2-D texture, both of which are present in the visual scene. The vision system, in turn, might provide the candidate referents back to the NLU system, which can then attempt to interpret subsequent expressions as further specifications of the referent ("on it"). If reference resolution fails, then the dialogue system can attempt to ask for further specification. It is this kind of internal dialogue between the natural language and vision systems that enable one-shot visual object learning. We will, thus, briefly describe those systems next.

The Natural Language System

Several components are needed for spoken natural language-dialogue interactions, including a *speech recognizer*, a *syntactic and semantic parser*, a *pragmatic reasoner*, a *dialogue manager*, *a goal manager*, a *sentence generator*, and a *speech synthesizer*. After the speech recognizer has recognized a word, the recognized word is passed on to the syntactic and semantic parsers for integration into the current parse. The goal of the parser is to utilize syntactic structure to form meaning representations that the robot can further process. For example, the query "Do you see a medkit?" will be represented as an interrogatory utterance with semantics represented in predicate form as $see(self, medkit)$. Note that asking whether the robot can see an object is not only a request for information, it

is an instruction to look around for the object in question (represented as $goal(self, found(self, medkit))$), as the robot must first try to *find* the object in question. The dialogue manager uses the pragmatic reasoner to carry out the pragmatic analysis necessary to generate these deeper semantics from the sentence's surface semantics. Pragmatic analysis is accomplished by comparing the incoming utterance (which is received in a form that contains utterance type, interlocutor information and adverbial modifiers in addition to surface semantics) against sets of pragmatic rules which map utterances (e.g., $AskYN(Speaker, Listener, see(Listener, medkit))$) to deeper semantics under certain contexts (e.g., $goal(Listener, found(Speaker, medkit))$).

Goals generated through pragmatic analysis are sent to the goal manager, which attempts to find appropriate actions to achieve those goals. In the case of $goal(self, found(self, medkit))$, the goal manager searches for and identifies the "find" action, which attempts to find the desired object in the robot's immediate environment. This action first tries to find an acceptable means to search for the object in question by requesting a *typeID* from the vision system (e.g., for the type "medkit") to ensure that the type of object in question is already known to the vision system. If the vision system knows of the object type, the goal manager instructs the vision system to perform a visual search using the identifier to locate objects of that type, and the results of the search will be reported back to the natural language system. If an acceptable visual search cannot be initiated because the object type is unknown, the goal manager will submit a natural language generation request to the dialogue manager to indicate that the robot does not know what the object looks like.

The dialogue manager can then keep track of the dialogue context and interpret subsequent utterances (based on their dialogue context and syntactic form) to be explanations of the object type in question. The explanation, once parsed, is then sent back to the vision system which can use it to form a new object representation together with a configuration of its subsystems that will allow it to recognize the object. Details of the employed natural language system are described in (Cantrell et al. 2010; 2012; Briggs and Scheutz 2013).

The Vision System

The vision system is a highly parallelized and modular set of *Saliency Operators*, *Detectors*, *Validators*, and *Trackers*, responsible for detecting, identifying, and tracking objects that can be described in natural language (see Figure 1). It supports multiple asynchronous *visual searches* that can be initiated dynamically and incrementally through interactions with the natural language system. Each visual search is controlled by a *Search Manager* which is instantiated by a natural language description and responsible for assembling and managing the requisite processes that are necessary for performing the particular visual search. Search managers can range in complexity from simple searches consisting of only one *Detector* and one *Tracker* (with no *Saliency Operators* and no *Validators*), to hierarchical searches consisting of several layered *Search Managers*.

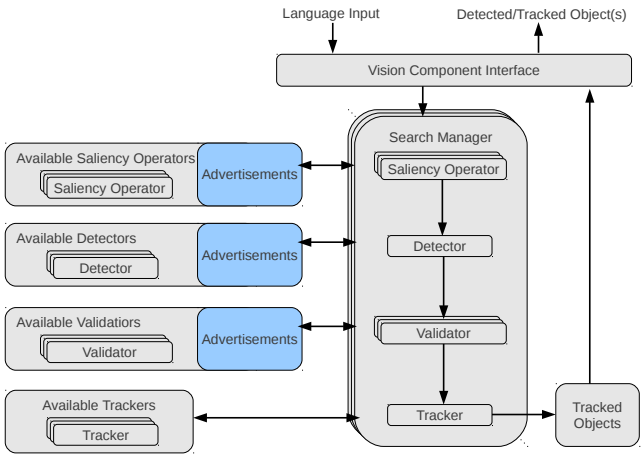

Figure 1: An overview of the vision system.

Saliency Operators are attentional mechanisms that operate on entire images to locate the most salient regions of an image with respect their associated property (e.g., color, orientation, size, etc.). Each operator produces a probability map where each pixel of the map is assigned a value in the range [0, 1]. The resulting maps from multiple saliency operators are fused together by a special purpose "Master Saliency Operator" in order to locate the overall most salient image region (cp. to (Itti and Koch 2001)).

Detectors are responsible for segmenting target object candidates and can be specialized detectors (e.g., face detector) or generic segmentation processes that attempt to cluster the scene into constituent parts. Generic segmenters can consume saliency maps to guide the segmentation process to particular parts of the scene (Krause et al. 2013).

Validators check segmented objects for a particular visual property (e.g., color, shape, etc.), operating in parallel or chained together sequentially. If an object candidate is found to have the required visual property, it is tagged as such and a confidence level between [0, 1] is assigned for that property. Successfully validated objects are passed on through the chain of Validators, being tagged by each Validator along the way. If a candidate object fails to meet the requirements of any Validator, it is dropped from further consideration.

Trackers receive fully validated objects which they then track from image to image. Various tracking options are available, and can range from simple matching based trackers to more heavy-weight trackers such as OpenTLD (Kalal, Mikolajczyk, and Matas 2012). Tracked objects are then available to other components in the system (e.g., the dialogue system, the manipulation system, etc.).

The particular generic *detector* used in the experiments below segments RGB-D point clouds of possibly cluttered scenes into a set of object candidates (Richtsfeld et al. 2014), by grouping locally smooth surface patches using learned 3D perceptual grouping principles. Each object candidate, in turn, consists of a set of surface patches and their neighborhood relations. Using a mixture of planar patches and NURBS patches allows for accurate modeling of a wide va-

riety of typical indoor objects.

Furthermore, the segmented surfaces are inspected by *validators* for color, texture, and shape properties. The shape validator checks planarity and relative orientation of patches. And the texture validator checks individual surfaces for known textures based on 2D shape contexts (Belongie, Malik, and Puzicha 2002). Shape contexts are particularly suited to one shot learning of textures in dialogue contexts. Symbols and patterns marking objects for specific uses, such as a red cross or a biohazard sign, are more likely to be explicitly referred to in a dialogue than, for example, the texture of a cereal box (note that people will refer to the cereal box itself, not its texture). Such "simple" textures are better described by an edge based descriptor (such as shape contexts) than by distinctive feature descriptors (such as SIFT), which work best with rich, dense textures. Shape contexts are at the same time very descriptive and robust against various distortions (we want to recognize any cross, not just a specific instance). A shape context descriptor is learned from a single object view, adding several artificially warped versions of the training image for improved robustness.

To facilitate the building of visual searches from natural language, the vision system takes simple quantifier-free first-order predicate expressions (i.e., the meaning expressions of noun phrases) and maps them onto various vision processes. Specifically, the language input is first decomposed into constituent parts (predicate expressions), and then an attempt is made to satisfy each descriptor with an existing vision capability. Each vision process mentioned above (with the exception of Trackers) provides a description of its capabilities so that a Search Manager can easily assemble the necessary vision processes based on the predicate description. These existing vision processes can be traditional bottom-up vision algorithms that have been trained off-line, a composition of low-level vision algorithms that have already been learned through one-shot learning, or logic-guided specifications of how vision processes should be assembled (as is the case with spatial relations).

Proof-of-Concept Evaluations

The vision and natural language systems have been fully implemented in our DIARC architecture (Scheutz et al. 2007; 2013) and evaluated on a Willow Garage PR2 robot in real-time spoken dialogue interactions. The particular scenario we report here is one in which the robot is located in front of a table with several objects unknown to the robot (see the bottom in Figure 2). A human instructor then teaches the robot what a "medkit" is through a simple natural language dialogue (see the top of Figure 2):

```
H: Do you see a medkit?
R: What does a medkit looks like?
```

Here the request for type identifier for "medkit" failed, hence no visual search could be performed, and the dialogue manager was tasked with providing the appropriate feedback to the human instructor.

```
H: It is a white box with a red cross on it.
R: Ok.
```

After receiving the definition of "medkit" and having been able to configure a visual search, the robot now knows how

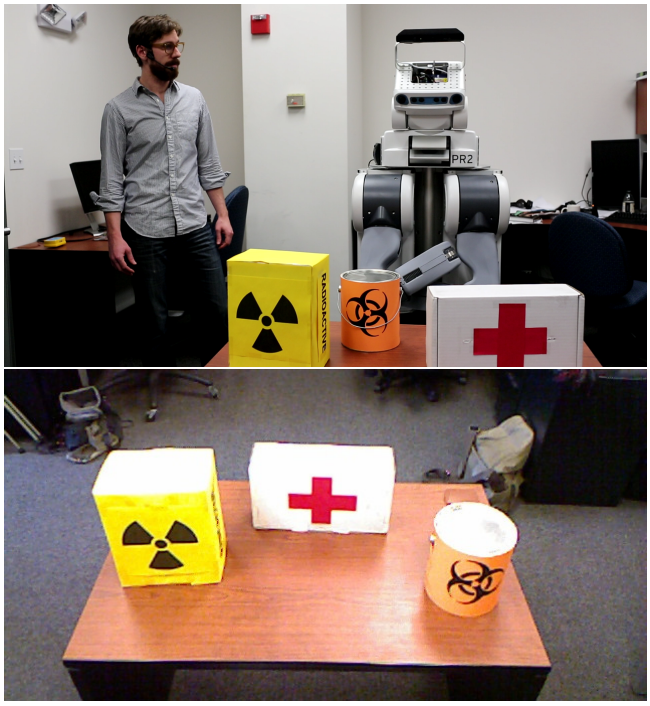

Figure 2: A human instructor teaching the robot what a "medkit" is (top); a typical instruction scene viewed from robot's mounted Kinect sensor (bottom).

to recognize medkits and complete the search (despite various distractor objects). Specifically, the utterance "It is a white box with a red cross on it" is translated by the natural language processing system into the quantifier-free logical form (with free variables ranging over perceivable objects)

$$\text{color}(X,\text{white}) \wedge \text{type}(X,\text{box}) \wedge$$
$$\text{color}(Y,\text{red}) \wedge \text{type}(Y,\text{cross}) \wedge \text{on}(Y,X)$$

which then gets passed on to the vision system. Search constraints are grouped based on their free variables, and each collection is used to instantiate a visual search. The fragment "color(X,white)$\wedge$type(X,box)" triggers a visual search for white boxes ($VS1$) while the fragment "color(Y,red)$\wedge$type(Y,cross)" triggers a separate visual search for red crosses ($VS2$). This leaves the relation "on(Y,X)" which initiates a third search for the "on"-relation between two objects as given by searches that found objects for the free variables X and Y. Note that there is an interesting complication in this example in that the expression "red cross" is not used to indicate an object but a texture and that the "on"-relation really can hold between two objects, two textures, and an object and a texture. Since the vision system in this configuration does not have a validator for cross objects, only one for cross textures, it searches for a texture on the surface of segmented objects (had it had a validator for cross objects, it would have had to check the other "on" relation too).

When asked, the robot confirms that it sees a medkit:

```
H: Now do you see one?
R: Yes.
```

And when the medkit is moved to a different location in a different pose, the robot can still recognize and point to it:

```
H: How about now?
R: Yes.
H: Where is it?
R: It's over there.
```

A video is available at `http://tinyurl.com/kjkwtos`.

We also verified that the vision system could learn the remaining two objects depicted in Figure 2: "yellow box with a radioactive symbol on it" and "orange cylinder with a biohazard symbol on it", which are formally translated as

color(X,yellow)∧type(X,box)∧type(Y,RA)∧on(Y,X)
color(X,orange)∧type(X,cylinder)∧type(Y,BH)∧on(Y,X)

Note that these descriptions differ from that of the medkit in that no color description was given for the texture (i.e., the colors of the radioactive and biohazard symbol were not specified). Hence, different colors are possible for those markers, while for the medkit the red color of the cross became an essential part of the "medkit" object description.

Discussion

The above example demonstrates how the robot can learn to recognize new objects based on a single natural language description *as long as* it has a way to recognize the component parts and their spatial relations as detailed in linguistic expression (i.e., the concepts of 3-D box and 2-D cross as well as colors and spatial relationships in the above example). Note that the main goal was to provide a proof-of-concept demonstration on a fully autonomous robot to show that the proposed one-shot learning method is viable and feasible, not not to provide a thorough evaluation of the one-shot learning capabilities enabled by the proposed interactions between the natural language and vision systems. For one, because the current system does not have a large number of basic detection and natural language capabilities as the focus is on the depth of the integration, not the breath of coverage. But more importantly, because it is not even clear how to best evaluate such integrated capabilities (e.g., how many instruction-object pairs would it take to convincingly demonstrate that the algorithms generalize?). We believe that a thorough evaluation of one-shot learning will ultimately have to include a careful selection of different types of interaction scenarios and contexts that each can each highlight important aspects of one-shot learning. For example, it would be possible for the robot to learn how to recognize an object even without seeing it as long as the linguistic expression is unambiguous and it already knows all constituent concepts (in the above case, the ambiguity between "red cross" denoting an object vs. a texture would have to be resolved linguistically, for example, by saying "red cross symbol"). Moreover, the robot could try to maximally exploit the visual information provided in the instruction context to learn as many aspects as possible, only requiring minimal knowledge to begin with. For example, the robot could also learn what a cross was based on its knowledge of red and a capability to form new shape descriptors, or it could learn what "on" meant based on the spatial relation between the box and the red cross. Moreover, it could exploit linguistic knowledge that "red" and "white" are color terms to even learn those colors based on its perceptions. These would clearly be interesting directions to pursue in the future.

Finally, it is important to point out that our proposed approach to one-shot learning should be viewed as a complement to existing data-driven approaches, not as a replacement. For one, not all objects can easily be described in natural language and thus learned from natural language descriptions. In particular, natural kinds such as plants, rock formations, and many other naturally occurring, visually perceivable phenomena are often difficult to break down into constituent parts for which natural language descriptions exist (e.g., due to their irregular shape and structure). In this case, a data-driven approach that was trained on many instances of a category (e.g., flower) would be able to provide a new primitive object category that could then be used in other object descriptions that are decomposable (e.g., "flower bed" as "multiple flowers in a row"). Also note that it is possible for both types of representations ("structured" vs. "atomic") to co-exist for the same object category, which has the advantage of recognizing objects faster and allowing for the recognition of potentially more instances depending on training data, while also having access to structured representations that allow for augmented learned models and for NL descriptions of objects (e.g., if the robot has to describe an unknown object to a human interactant). It might also be possible to use structured representations to bias data-driven methods to speed up learning and achieve better recognition performance (e.g., knowing that "handle" has to be part of "mug" no matter what the mug looks like, might provide an important bias to data-driven segmentation).

Conclusion

We argued that data-driven learning methods have to be complemented by one-shot learning methods to meet the demands of future human-robot interaction scenarios. We introduced a method for one-shot language-guided visual object learning that requires a deep integration between natural language and vision processing algorithms and demonstrated the viability of the proposed approach on a robot in a simple human-robot dialogue. Future work will extend the current system to allow the robot to maximally exploit the information present in both the linguistic and visual stimuli, which should enable additional one-shot learning of shapes, textures, and spatial relations.

Acknowledgments

This work was in part supported by US NSF grant IIS-1111323, ONR grants #N00014-11-1-0289 and #N00014-14-1-0149 to the last author and EU FP7 grants #600623 and #610532 and Austrian Science Foundation grant TRP 139-N23 to the second author.

References

Bart, E., and Ullman, S. 2005. Cross-generalization: learning novel classes from a single example by feature replacement. *CVPR'05*.

Belongie, S.; Malik, J.; and Puzicha, J. 2002. Shape Matching and Object Recognition Using Shape Contexts. *Pattern Analysis and Machine Intelligence* 24:509–522.

Briggs, G., and Scheutz, M. 2013. A hybrid architectural approach to understanding and appropriately generating indirect speech acts. In *Proceedings of Twenty-Seventh AAAI Conference on Artificial Intelligence*, 1213–1219.

Cantrell, R.; Scheutz, M.; Schermerhorn, P.; and Wu, X. 2010. Robust spoken instruction understanding for HRI. In *Proceedings of the 2010 Human-Robot Interaction Conference*, 275–282.

Cantrell, R.; Talamadupula, K.; Schermerhorn, P.; Benton, J.; Kambhampati, S.; and Scheutz, M. 2012. Tell me when and why to do it!: Run-time planner model updates via natural language instruction. In *Proceedings of the IEEE Human-Robot Interaction Conference*, 471–478.

Collet, A.; Berenson, D.; Srinivasa, S. S.; and Ferguson, D. 2009. Object Recognition and Full Pose Registration from a Single Image for Robotic Manipulation. In *ICRA'09*.

Farhadi, A.; Endres, I.; Hoiem, D.; and Forsyth, D. 2009. Describing objects by their attributes. In *CVPR'09*.

Fei-Fei, L.; Fergus, R.; and Perona, P. 2006. One-shot learning of object categories. *Pattern Analysis and Machine Intelligence* 28(4).

Fei-Fei, L.; Fergus, R.; and Perona, P. 2007. Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding* 106(1):59–70.

Fergus, R.; Fei-Fei, L.; Perona, P.; and Zisserman, A. 2005. Learning Object Categories from Google's Image Search. In *CVPR'05*.

Fergus, R.; Weiss, Y.; and Torralba, A. 2009. Semi-Supervised Learning in Gigantic Image Collections. In *NIPS'09*.

Ferrari, V., and Zisserman, A. 2007. Learning Visual Attributes. In *NIPS'07*.

Fitzpatrick, P.; Metta, G.; Natale, L.; Rao, S.; and Sandini, G. 2003. Learning about Objects through Action - Initial Steps Towards Artificial Cognition. In *ICRA'03*.

Herbst, E.; Ren, X.; and Fox, D. 2011. RGB-D Object Discovery Via Multi-Scene Analysis. In *IROS'11*.

Hwang, S. J.; Sha, F.; and Grauman, K. 2011. Sharing features between objects and their attributes. In *CVPR'11*.

Itti, L., and Koch, C. 2001. Computational modeling of visual attention. In *Nature reviews, Neuroscience*, 194–203.

Kalal, Z.; Mikolajczyk, K.; and Matas, J. 2012. Tracking-Learning-Detection. *Pattern Analysis and Machine Intelligence* 34(7):1409–1422.

Kang, H.; Hebert, M.; and Kanade, T. 2011. Discovering Object Instances from Scenes of Daily Living. In *ICCV'11*.

Karpathy, A.; Miller, S.; and Fei-Fei, L. 2013. Object Discovery in 3D Scenes via Shape Analysis. In *ICRA'13*.

Katz, D.; Venkatraman, A.; Kazemi, M.; Bagnell, D.; and Stentz, A. 2013. Perceiving, Learning, and Exploiting Object Affordances for Autonomous Pile Manipulation. In *Robotics Science and Systems*.

Krainin, M.; Henry, P.; Ren, X.; and Fox, D. 2011. Manipulator and object tracking for in-hand 3D object modeling. *The International Journal of Robotics Research* 30(11):1311–1327.

Krause, E.; Cantrell, R.; Potapova, E.; Zillich, M.; and Scheutz, M. 2013. Incrementally Biasing Visual Search Using Natural Language Input. In *AAMAS'13*.

Lake, B. M.; Salakhutdinov, R.; and Tenenbaum, J. B. 2012. Concept learning as motor program induction: A large-scale empirical study. In *Cognitive Science Conference*.

Leibe, B.; Leonardis, A.; and Schiele, B. 2008. Robust Object Detection with Interleaved Categorization and Segmentation. *International Journal of Computer Vision* 77(1-3):259–289.

Logothetis, N., and Sheinberg, D. L. 1996. Visual object recognition. *Annual Review of Neuroscience* 19:577–621.

Miller, E.; Matsakis, N.; and Viola, P. 2000. Learning from one example through shared densities on transforms. In *CVPR'00*.

Mohan, S.; Mininger, A.; Kirk, J.; and Laird, J. E. 2012. Learning grounded language through situated interactive instruction. In *AAAI Fall Symposium Series*, 30–37.

Özuysal, M.; Calonder, M.; Lepetit, V.; and Fua, P. 2009. Fast Keypoint Recognition Using Random Ferns. *Pattern Analysis and Machine Intelligence* 32(3):448–461.

Richtsfeld, A.; Mörwald, T.; Prankl, J.; Zillich, M.; and Vincze, M. 2014. Learning of perceptual grouping for object segmentation on RGB-D data. *Journal of Visual Communication and Image Representation* 25(1):64–73.

Salakhutdinov, R.; Tenenbaum, J.; and Torralba, A. 2012. One-Shot Learning with a Hierarchical Nonparametric Bayesian Model. *JMLR Workshop on Unsupervised and Transfer Learning* 27:195–207.

Scheutz, M.; Schermerhorn, P.; Kramer, J.; and Anderson, D. 2007. First steps toward natural human-like HRI. *Autonomous Robots* 22(4):411–423.

Scheutz, M.; Briggs, G.; Cantrell, R.; Krause, E.; Williams, T.; and Veale, R. 2013. Novel mechanisms for natural human-robot interactions in the diarc architecture. In *Proceedings of AAAI Workshop on Intelligent Robotic Systems*.

Skocaj, D.; Kristan, M.; Vrecko, A.; Marko, M.; Miroslav, J.; Kruijff, G.-J. M.; Hanheide, M.; Hawes, N.; Keller, T.; Zillich, M.; and Zhou, K. 2011. A system for interactive learning in dialogue with a tutor. In *IROS'11*.

Ückermann, A.; Haschke, R.; and Ritter, H. 2012. Real-Time 3D Segmentation of Cluttered Scenes for Robot Grasping. In *IROS'12*.

Varadarajan, K. M., and Vincze, M. 2012. AfRob: The affordance network ontology for robots. In *IROS'12*.

Wohlkinger, W.; Buchaca, A. A.; Rusu, R.; and Vincze, M. 2012. 3DNet: Large-Scale Object Class Recognition from CAD Models. In *ICRA'12*.