INGRESS: Interactive Visual Grounding of Referring Expressions

The International Journal of Robotics Research
XX(X):1–15

DOI: 10.1177/ToBeAssigned
www.sagepub.com/
SAGE

Mohit Shridhar[1], Dixant Mittal[2] and David Hsu[2]

Abstract

This paper presents INGRESS, a robot system that follows human natural language instructions to pick and place everyday objects. The key question here is to ground referring expressions: understand expressions about objects and their relationships from image and natural language inputs. INGRESS allows unconstrained object categories and rich language expressions. Further, it asks questions to clarify ambiguous referring expressions interactively. To achieve these, we take the approach of *grounding by generation* and propose a two-stage neural-network model for grounding. The first stage uses a neural network to generate visual descriptions of objects, compares them with the input language expressions, and identifies a set of candidate objects. The second stage uses another neural network to examine all pairwise relations between the candidates and infers the most likely referred objects. The same neural networks are used for both grounding and question generation for disambiguation. Experiments show that INGRESS outperformed a state-of-the-art method on the RefCOCO dataset and in robot experiments with humans. The INGRESS source code is available at https://github.com/MohitShridhar/ingress.

Keywords

Natural language grounding, disambiguation, human-robot interaction

1 Introduction

The human language provides a powerful natural interface for interaction between humans and robots. In this work, we aim to develop a robot system that follows natural language instructions to pick and place everyday objects. To do so, the robot and the human must have a shared understanding of language expressions and the environment.

The core issue here is to *ground* natural language referring expressions: locate specific objects from input language expressions and images of the environment (Figure 1). To focus on this main issue, we assume for simplicity that the scene is uncluttered and the objects are clearly visible. While prior work on object retrieval typically assumes predefined object categories, we want to allow unconstrained object categories so that the robot can handle a wide variety of everyday objects not seen before (Figure 1). Further, we want to allow rich human language expressions in free form, with no artificial constraints (Figure 1). Finally, despite the richness of human language, referring expressions may be ambiguous. The robot should ask the human questions interactively in order to disambiguate (Figure 1 *c*).

To tackle these challenges, we take the approach of *grounding by generation*, analogous to that of analysis by synthesis (Neisser 2014). We propose a neural-network grounding model, consisting of two networks trained on large datasets, to generate language expressions from the input image and compare them with the input referring expression. If the referring expression is ambiguous, the same networks are used to generate questions interactively for disambiguation. We call this approach INGRESS, for INteractive visual Grounding of Referring ExpreSSions.

A referring expression may contain self-referential and relational sub-expressions. Self-referential expressions describe an object in terms of its own attributes, e.g., name, color, or shape. Relational expressions describe an object in relation to other objects, e.g., spatial relations. By exploiting the compositionality principle of natural language (Werning et al. 2012), INGRESS decomposes the grounding process into two stages (Figure 2). The first stage uses a neural network to ground the self-referential sub-expressions and identify a set of candidate objects. The second stage uses another neural network to ground the relational sub-expressions by examining all pairwise relations between the candidate objects. Following the earlier works of Bisk et al. (2016); Nagaraja et al. (2016); Tellex et al. (2010), we focus on binary relations here, in particular, visual binary relations.

When the referred object cannot be uniquely identified from the initial language and image inputs, INGRESS asks disambiguation questions and asks as few questions as possible. We present two methods, a simple heuristic and a more sophisticated partially observable Markov decision process (POMDP). Through probabilistic language modeling

[1]Paul G. Allen School of Computer Science and Engineering, University of Washington, Seattle, WA, USA. The work was completed while the author was at the National University of Singapore.
[2]School of Computing, National University of Singapore, Singapore.

Corresponding author:
Mohit Shridhar, Paul G. Allen School of Computer Science and Engineering, University of Washington, 185 E Stevens Way NE, Seattle, WA 98195, USA
Email: mshr@cs.washington.edu

(a) (b) (c)

Figure 1. Interactive visual grounding of referring expressions. (*a*) Ground self-referential expressions. (*b*) Ground relational expressions. (*c*) Ask questions to resolve ambiguity. Red boxes indicate referred objects. Blue dashed boxes indicate other candidate objects. See also the accompanying video at `http://bit.ly/INGRESSvid`.

and reasoning under uncertainty, the POMDP method shows better performance over the heuristic method in disambiguation by asking fewer questions, resulting in more natural and fluid user interaction.

We implemented INGRESS on a Kinova Mico robot manipulator, with voice input and RGB-D sensing. Experiments show that INGRESS outperformed a state-of-the-art method on the RefCOCO test dataset (Kazemzadeh et al. 2014) and in robot experiments with humans.

2 Related Work

Grounding referring expressions is a classic question widely studied in natural language processing, computer vision, and robotics (Clark et al. 1991; Pateras et al. 1995). A recent study by Li et al. (2016b) identifies four key issues in grounding for human-robot collaborative manipulation: visual search, spatial reference, ambiguity, and perspectives (e.g., "on my left"). Our work addresses the first three issues and briefly touches on the last one.

Visual grounding of referring expressions is closely related to object recognition. In robotics, object recognition is often treated as a classification task, with a predefined set of object category labels (Eppner et al. 2016; Pangercic et al. 2012; Krishnamurthy and Kollar 2013). These methods restrict themselves to tasks covered by predefined visual concepts and simple language expression templates. Other methods such as the works of FitzGerald et al. (2013) and Matuszek et al. (2012) relax the restriction on language by developing a joint model of language and perception, but they have difficulty in scaling up to many different object categories.

Relations play a critical role in grounding referring expressions for human-robot interaction, as objects are often described in relation to others. Again, some earlier work treats relational grounding as a classification task with predefined relation templates e.g., Golland et al. (2010); Guadarrama et al. (2013); Huo and Skubic (2016). A recent state-of-the-art method by Paul et al. (2016) performs sophisticated spatial inference on probabilistic models, but it assumes an explicit semantic map of the world and relies on formal language representation generated by a syntactic parser, which does not account for the visual context and is sensitive to grammatical variations.

Our approach to visual grounding is inspired by recent advances in image caption generation and understanding (Hu et al. 2016; Johnson et al. 2016; Mao et al. 2016; Nagaraja et al. 2016; Yu et al. 2017; Karpathy and Fei-Fei 2015; Vinyals et al. 2015). By replacing traditional handcrafted visual feature extractors with convolutional neural networks (CNNs) and replacing language parsers with recurrent neural networks (RNNs), these methods learn to generate and comprehend sophisticated human-like object descriptions for unconstrained object categories. In essence, the networks automatically connect visual concepts and language concepts by embedding them jointly in an abstract space. Along this line, Nagaraja et al. (2016) propose a network specifically for grounding relational expressions. Similarly, Hu et al. (2017) propose a modular neural network and train it for grounding end-to-end. In contrast, we train separate neural networks for self-referential and relational expressions and use them in a generative manner. This allows us to generate questions for disambiguation, an important issue not addressed in the the earlier work.

Our approach of grounding by generation is broadly related to inverse semantics (Tellex et al. 2014), in which probabilistic grounding graphs are used to generate natural language requests for help during robot failures. The recent work of Arkin and Howard (2018) similarly uses proactive grounding of likely utterances to improve the efficiency of dialogue interactions. These model-based Bayesian methods quantify the uncertainty of grounding, but compared with data-driven neural network methods, they face difficulty in scaling up to large domains with many object categories.

Ambiguity is an important issue for grounding in practice, but scarcely explored. The recent work of Hatori et al. (2018) detects ambiguities, but relies on fixed generic questions, such as "which one?", to acquire additional information for disambiguation. The work of Whitney et al. (2017) forms a POMDP model for disambiguation, but it again relies on fixed generic questions, such as "do you mean this one?", together with a pointing gesture. To process verbal responses, it builds a probability model over words from a predefined vocabulary of known objects. INGRESS generates *object-specific* questions, e.g., "do you mean this blue plastic bottle?", resulting in improved disambiguation performance. It also allows unrestricted object categories.

Interactive visual grounding is also related to the broader question of visual question answering (VQA), e.g., (De Vries et al. 2017a; Das et al. 2017; De Vries et al. 2017b). It is also related to slot-filling dialog systems (Williams and Young 2007; Doshi and Roy 2008), which do not take advantage of visual inputs, and language-based navigation tasks Mei et al. (2016); Fried et al. (2018), which focus on

grounding actions and spatial relations, sometimes through dialog (Hemachandra and Walter 2015).

Our work integrates language grounding, visual information processing, and robot actions. It evaluates the system on a real robot with humans. This paper expands our earlier work (Shridhar and Hsu 2018) by introducing a POMDP model, INGRESS-POMDP, for disambiguation and integrating it with visual grounding for improved performance. INGRESS-POMDP provides a principled probabilistic decision framework to choose *what* questions to ask and *when* to stop. We also provide additional details on the neural-network model for visual grounding and the overall system design.

3 Overview

INGRESS breaks the grounding process into two stages sequentially and trains two separate LSTM networks, S-LSTM and R-LSTM, for grounding self-referential expressions and relational expressions, respectively (Figure 2). The two-stage neural network grounding takes advantage of the compositionality principle of natural language (Werning et al. 2012). In particular, relations, such as "left of", are independent of entities, such as "blue cups" and "stuffed animals". They can be composed to form an expression, e.g., "a blue cup on the left of stuffed animals". This *neural modular* approach to grounding exploits the compositional structure of language in the neural network architecture and has been gaining increasing attention because of its empirical success. See, e.g., (Andreas et al. 2016; Johnson et al. 2017b; Hu et al. 2017) . Further, the first stage acts as a 'filter', which reduces the number of candidate objects for relational grounding in the second stage, and improves computational efficiency as a result.

Each stage follows the grounding-by-generation approach and uses the LSTM network to generate a textual description of an input image region or a pair of image regions. It then compares the generated expression with the input expression to determine the most likely referred object. An alternative is to train the networks directly for grounding instead of generation, but it is then difficult to use them for generating questions in case of ambiguity. We describe the grounding model in more detail in Section 4.

To resolve ambiguities, INGRESS uses S-LSTM or R-LSTM to generate the textual description of a candidate object and fits it to a question template to generate an object-specific question. The user then may provide a correcting response based on the question asked. We describe two methods for choosing the disambiguation questions in Section 5.

While INGRESS handles a wide variety of language expressions as well as object categories and relations, its performance is ultimately limited by training data. We examine these limitations in Sections 7 and 8.

4 Visual Grounding

4.1 Grounding Self-Referential Expressions

Given an input image I and an expression E, the first stage of INGRESS aims to identify candidate objects from I and self-referential sub-expressions of E. More formally, let R be a rectangular image region that contains an object. We want to find image regions with high probability $p(R \mid E, I)$. Applying the Bayes' rule, we have

$$\begin{aligned}\arg\max_{R\in\mathcal{R}} \; p(R \mid E, I) &= \arg\max_{R\in\mathcal{R}} \; \frac{p(E \mid R, I)\, p(R \mid I)}{p(E \mid I)} \\ &= \arg\max_{R\in\mathcal{R}} \; p(E \mid R, I)\, p(R \mid I),\end{aligned} \tag{1}$$

where $\mathcal{R}$ is the set of all rectangular image regions in I. Assuming a uniform prior over the image regions, our objective is then to maximize $p(E \mid R, I)$, in other words, to find an image region R that most likely generates the expression E.

To do so, we apply the approach of DenseCap (Johnson et al. 2016), which directly connects image regions that represent object proposals with natural expressions, thus avoiding the need for predefined object categories. See Figure 2 for an overview. First, we use a Faster R-CNN based localization module (Johnson et al. 2016) to process the input image I and find a set of image regions $R_i, i = 1, 2, \ldots$, each representing an object proposal. We use a fully connected layer to process each region R_i further and produce a 4096-dimensional feature vector f_i. Next, we feed each feature vector f_i into S-LSTM, an LSTM network, and predict a sequence S_i of word probability vectors. The sequence S_i represents the predicted expression describing R_i. The jth vector in S_i represents the jth word in the predicted expression, and each element of a vector in S_i gives the probability of a word.

The input sequence E is padded to have the same length as S. The full expressions, such as the ones highlighted in Figure 3, are generated with an *argmax* over each word probability vector S_i in the sequence. We then calculate the average cross entropy loss (CEL) between E and S_i, or equivalently $p(E \mid R_i, I) = p(E \mid S_i)$, which compares the input and generated word probability distributions. Effectively, the S-LSTM output allows us to estimate the probability of each word in an expression. The average cross entropy loss over all words in the expression indicates how well it describes an image region. For more details regarding the expression generation, see DenseCap (Johnson et al. 2016).

Our implementation uses a pre-trained captioning network provided by DenseCap (Johnson et al. 2016). The network was trained on the Visual Genome dataset (Krishna et al. 2016), which contains around $100,000$ images and $4,300,000$ expressions, making the model applicable to a diverse range of real-world scenarios. On average, each image has 43.5 region annotation expressions, e.g., "cats play with toys hanging from a perch" and "woman pouring wine into a glass".

4.2 Relevancy Clustering

While CEL measures how well the input expression matches the generated sequence of word probability vectors, it is subjected to visual ambiguity as a result of lighting condition variations, sensor noise, object detection failures, etc. Consider the Pringles chip can example in Figure 3. The

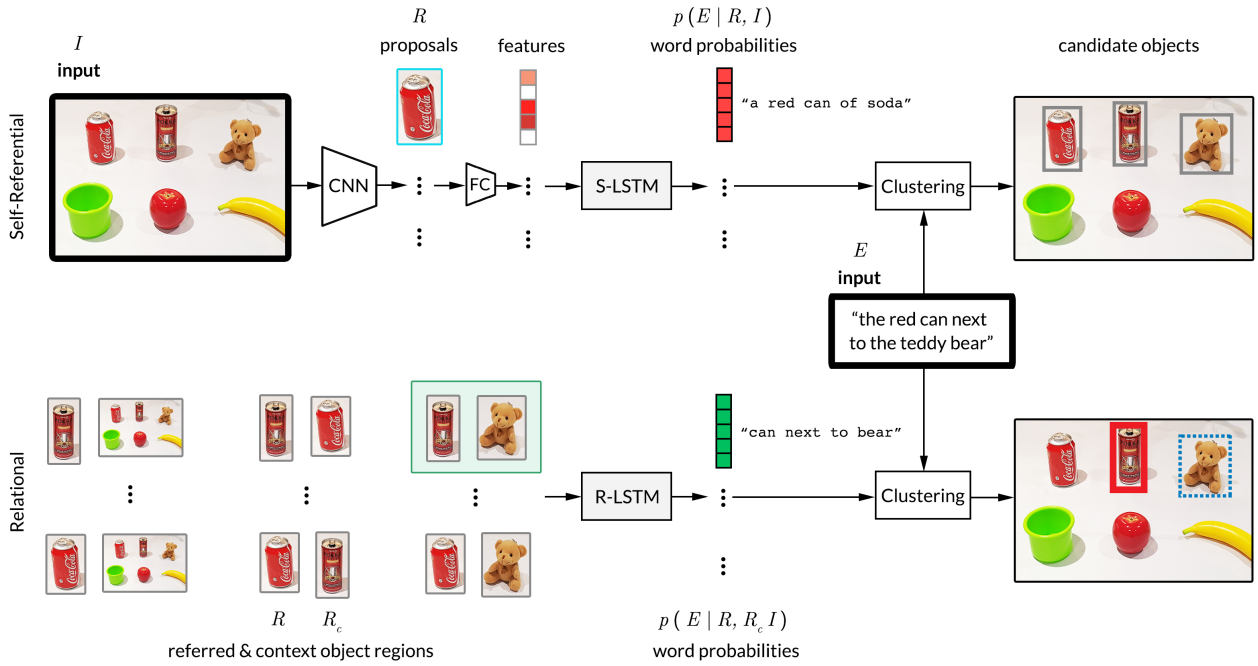

Figure 2. INGRESS overview. The first stage grounds self-referential expressions and outputs a set of candidate referred objects (top row). The input image goes into a Faster R-CNN based localization module (Johnson et al. 2016) to generate image regions representing object proposals. Each image region goes into a fully connected network to extract a feature vector, which in turn goes into an LSTM network to generate a word probability sequence that represents an expression distribution describing the image region. The generated expression and the input expression are compared to find candidates for the referred object. The second stage grounds relational expressions by examining all pairs of candidate image regions (bottom row). Each pair goes into another LSTM network, which generates a word probability sequence describing the relation between the pair of image regions. Again, the generated expression and the input expression are compared to find the referred object.

image region contains only part of the can, and it is visually quite similar to a red cup. CEL is thus low, indicating a good fit, unfortunately. Further, the word probability vectors might not consider paraphrases and synonyms, unless explicitly trained with specific examples.

To deal with these issues, we consider an additional measure, METEOR (Banerjee and Lavie 2005). METEOR is a standard machine translation metric that calculates the normalized semantic similarity score between two sentences. For example, the METEOR score between "the green glass" and "the green cup" is 0.83, and that between "the green glass" and "the blue book" is 0.06. The score is based on explicit word-to-word matches and word alignment between sentences. The word matches consider synonyms, and alignments can handle simple morphological variants or paraphrases. We calculate the METEOR measure by generating the most likely expression E_i from S_i and comparing E_i with the input expression E. METEOR, however, has its own limitation. It does not account for the visual context and treats all words in an expression with equal importance. For example, the METEOR score between "a blue cup on the table" and "a red cup on the table" is high, because most words in the expressions match exactly (Figure 3).

For robustness, we calculate both CEL and METEOR between S_i and E, for $i = 1, 2, \ldots$. We then perform K-means clustering with normalized CEL & METEOR values and choose $K = 2$ to form two clusters of relevant and irrelevant candidate image regions for the referred object (Figure 3). Finally, the relevant cluster $\mathcal{R}'$ is sent to the second stage of the grounding model, if $\mathcal{R}'$ contains multiple

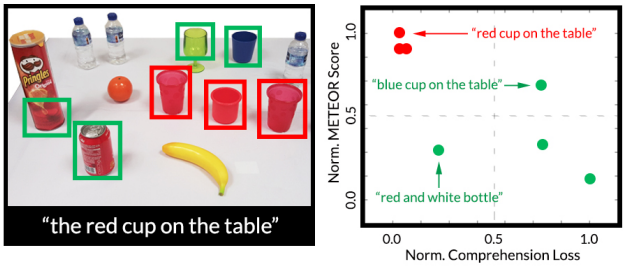

Figure 3. Relevancy clustering. Red boxes (left) and red dots (right) indicate relevant objects. Green boxes and dots indicate irrelevant objects. The labels pointing to the dots are generated self-referential expressions.

candidates. This handcrafted clustering step is somewhat ad hoc, but it improves the performance in our experiments, and future work could consider learning to cluster in an end-to-end manner or integrating more sophisticated clustering methods (Cherouvim and Papadopoulos 2005).

4.3 Grounding Relational Expressions

In the second stage, we aim to identify the referred object by analyzing its relations with other objects. We make the usual assumption of binary relations (Bisk et al. 2016; Kollar et al. 2010; Nagaraja et al. 2016). While this may appear restrictive, binary relations are among the most common in everyday expressions. Further, some expressions, such as "the leftmost cup", seem to involve complex relations with multiple objects, but it can be, in fact, treated as a binary relation between the referred object and all other objects treated as a single set. Akin to the grounding of

self-referential expressions, we seek a pair of image regions, referred-object region R and context-object region R_c, with high probability $p(R, R_c \mid E, I)$:

$$\underset{\substack{R \in \mathcal{R}', R_c \in \mathcal{R}' \cup \{I\} \\ R \neq R_c}}{\arg\max} \; p(R, R_c \mid E, I) = \underset{\substack{R \in \mathcal{R}', R_c \in \mathcal{R}' \cup \{I\} \\ R \neq R_c}}{\arg\max} \; p(E \mid R, R_c, I). \tag{2}$$

Our approach for grounding relational expressions parallels that for grounding self-referential expressions. See Figure 2 for an overview. We form all pair-wise permutations of candidate image regions, including the special one corresponding to the whole image similar to Mao et al. (2016). Each input pair is composed of a concatenated vector $[f, b, f_c, b_c]$, where f and f_c are the referred and context feature vectors, b and b_c are the referred and context bounding boxes respectively. The bounding boxes are encoded in a normalized format: $[\frac{x_{tl}}{W}, \frac{y_{tl}}{H}, \frac{x_{br}}{W}, \frac{y_{br}}{H}, \frac{w \cdot h}{W \cdot H}]$, where (x_{tl}, y_{tl}) is the top-left spatial coordinate, (x_{br}, y_{br}) is the bottom-right spatial coordinate, w and h are the box width and height, W and H are the image width and height respectively. Most of the spatial information in the scene is captured by these 2D bounding box encodings. We feed the concatenated vectors into R-LSTM, another LSTM, trained to predict relational expressions. By directly connecting image region pairs with relational expressions, we avoid the need for predefined relation templates. For each image-region pair (R, R_c), we generate the relational expression E'. We compute CEL and METEOR between E' and the input expression E over all generated expressions and again perform K-means clustering with $K = 2$. If all pairs in the top-scoring cluster contain the same referred object, then it is uniquely identified. Otherwise, we take all candidate objects to the final disambiguation stage. Figure 2 illustrates the entire pipeline for grounding the sentence "the red can next to the teddy bear". The first clustering stage filters out self-referential expressions with high CEL and METEOR scores e.g. "a red can of soda", "red can" etc. from low-scoring ones e.g. "green cup on the table", "a teddy bear on the table" etc. The second clustering stage filters out a single high-scoring pair "can next to bear" from other low-scoring pairs "can above red ball", "banana on the right" etc.

Following the approach of UMD RefExp (Nagaraja et al. 2016), we trained R-LSTM on the RefCOCO training set (Kazemzadeh et al. 2014), which contains around $19,000$ images and $85,000$ referring expressions that describe visual relations between images regions, e.g., "bottle on the left". Specifically, we used UMD RefExp's Multi-Instance Learning Negative Bag Margin loss function for training. We used stochastic gradient decent for optimization, with a learning rate of 0.01 and a batch size of 16. The training converged after $70,000$ iterations and took about a day to train on an Nvidia Titan X GPU.

5 Resolving Ambiguities

For robot pick-and-place operations, we assume that the user intends for a single object to be picked up. If the object cannot be uniquely identified by grounding the self-referential and relational sub-expressions, the final disambiguation stage of INGRESS processes the remaining

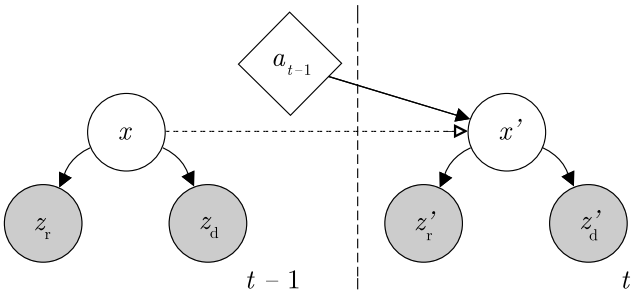

Figure 4. The graphical model of INGRESS-POMDP. x is the hidden variable representing the referred object. z_r and z_d are observations corresponding to the response utterance and the description utterance, respectively. a is the robot action that asks a disambiguation question. The dashed arrow from x to x' indicates that the value of x does not change over time.

candidate objects interactively. It asks the user "Do you mean ...?" to solicit additional information. Generating object-specific questions is straightforward for INGRESS, because of its grounding-by-generation design. To ask a question about an object, we either use S-LSTM or R-LSTM to generate an expression E and then fit it to the question template "Do you mean E?".

After detecting an ambiguous scenario, the robot must still decide what questions to ask, either self-referential or relational, and decide when to stop. The choice of questions is crucial for gathering the required information for disambiguation. For example, asking a self-referential question in a scene with visually similar objects does not help in resolving ambiguity. Further, the robot must balance the benefit of additional information against the risk of annoying the user, and stop with a reasonable amount of information to resolve the ambiguity. We present two methods below: a greedy heuristic method that goes through the candidate objects one by one and a principled POMDP model.

5.1 Heuristic Disambiguation

The heuristic method examines all relevant candidate image regions sequentially and uses a handcrafted heuristic to decide between the types of questions. In the order of descending likelihood (Equation 2), the robot physically points at an object and asks a question generated by either S-LSTM or R-LSTM. We initially choose S-LSTM, as most referring expressions primarily rely on visual information (Li et al. 2016b). We generate a self-referential expression for each candidate and check if it is informative. In this case, an expression E is *informative* if the average METEOR score between E and all other generated expressions is small, in other words, it is sufficiently different from all other expressions. If the most informative expression has an average METEOR score less than 0.25, we proceed to ask a question using E. Otherwise, we use R-LSTM to generate a relational question.

After asking the question, the user can respond "yes" to choose the referred object, "no" to continue iterating through other possible objects, or provide a specific corrrective response to the question, e.g., "no, the cup on the left". To process the correcting response, we re-run INGRESS with the new description expression after "no".

The heuristic method treats each question-answer interaction independently and does not maintain an explicit history of responses. For example, if the answer is "no, the other cup", the method cannot infer the meaning of "other cup" with respect to earlier interactions.

5.2 POMDP Disambiguation

The POMDP is a natural choice for the disambiguation task. To choose a disambiguation question, it systematically reasons about the entire history of interactions by maintaining a *belief*, i.e., a probability distribution over the referred object. Specifically, given the current belief, the robot searches a tree that encodes sequences of future question-answer interactions and chooses the best question in expectation. The robot asks the chosen question, receives a response from the user, and uses the information acquired to update the belief. The process then repeats. For disambiguation, a key feature of POMDP planning is the principled trade-off between gaining additional information by asking questions and the cost of doing so.

5.2.1 POMDP Model Our disambiguation POMDP model, INGRESS-POMDP, is defined formally as a tuple (X, A, Z, T, O, R):

- The state space X consists of a set of N candidate objects. The referred object $x_{\text{R}} \in X$ is unknown in advance.
- The action space A contains three types of actions: $\textsc{AskSelf}(x)$, $\textsc{AskRel}(x)$, and $\textsc{Pick}(x)$ for $x \in X$. $\textsc{AskSelf}(x)$ and $\textsc{AskRel}(x)$ are information-gathering actions that ask a self-referential question (e.g. "do you mean the blue cup?") or a relational question (e.g. "do you mean the cup on the left?") about the image regions corresponding to x, respectively. Unlike similar actions for heuristic disambiguation (Section 5.1), $\textsc{AskSelf}(x)$ and $\textsc{AskRel}(x)$ are purely verbal, with no accompanying physical pointing gestures. They are thus much faster to execute. $\textsc{Pick}(x)$ is a physical action that commands the robot to finalize x as the desired object. For N candidate objects, there are a total of $3N$ actions.
- The observation space Z contains all possible user answers in response to disambiguation questions. Since there is no restriction on language expressions, the observation space size $|Z|$ is unbounded in the worst case. For illustration, $|Z|$ is on the order of 10^{60} for a vocabulary of 10,000 words and a maximum sentence length of 15.
- $T(x, a, x')$: We assume that the user does not change their mind about the referred object throughout the entire interaction. So x_{R} remains constant, and the probabilistic state-transition function $T(x, a, x') = p(x, a, x')$ is 1 if $x' = x$ and 0 otherwise.

a	x	$R(x, a)$
$\textsc{Pick}(x)$	$x = x_{\text{R}}$	10
$\textsc{Pick}(x)$	$x \neq x_{\text{R}}$	-10
$\textsc{AskSelf}(x)$	*	-1
$\textsc{AskRel}(x)$	*	-1

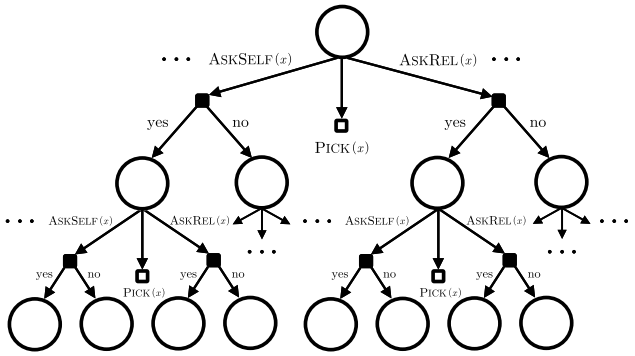

Figure 5. A disambiguation belief tree of depth $d = 2$.

- $O(x, a, z)$: The observation function $O(x, a, z) = p(z \mid x, a)$ captures the probability of a user's verbal response z to a question $\textsc{AskSelf}(x)$ or $\textsc{AskRel}(x)$.
- $R(x, a)$: The reward function $R(x, a)$ encourages the robot to identify the referred object as fast as possible. Each question, $\textsc{AskSelf}(x)$ or $\textsc{AskRel}(x)$, incurs a small penalty to encourage asking for information but not too many times. Picking the correct object results in a large reward, and picking the wrong one results in a large penalty. The reward values are empirically chosen to reflect this behavior.

See Figure 4 for the graphical model.

One main challenge of constructing the INGRESS-POMDP is observation modeling, as we allow unrestricted language expressions. We decompose an utterance $z \in Z$ into two parts: response utterance z_{r} and description utterance z_{d}. The response utterance corresponds to affirmatives represented by a set of positive words $Z_{\text{p}} = \{$'yes', 'yeah', 'yep', 'sure'$\}$ or non-affirmatives represented by a set of negative words $Z_{\text{n}} = \{$'no', 'nope', 'nah'$\}$. The description utterance z_{d} corresponds to any rich visual object description. Following earlier work (Whitney et al. 2017), we assume conditional independence and factor the observation model:

$$p(z \mid x, a) = p(z_{\text{r}} \mid x, a)\ \ p(z_{\text{d}} \mid x) \tag{3}$$

The probability of observing the response utterance, $p(z_{\text{r}} \mid x, a)$, is conditioned on the question asked. If the question mentions the attributes of the referred object, the user's response likely contains positive words. Otherwise, the response likely contains negative words. We assume that the user is mostly truthful in answering the questions. For simplicity, we specify a conditional probability table for $p(z_{\text{r}} \mid x, a)$ manually:

	$z_{\text{r}} \in Z_{\text{p}}$	$z_{\text{r}} \in Z_{\text{n}}$
$x = x_{\text{R}}$	0.99	0.01
$x \neq x_{\text{R}}$	0.01	0.99

These values assume the user is cooperative 99% of the time, but they can be easily learned from data as well.

The probability of observing the description utterance $p(z_{\text{d}} \mid x)$ is simply the probability of generating the description z_{d} given the object x. This corresponds to

(Equation 1) for self-referential expressions or (Equation 2) for relational expressions. It is independent of the question asked.

The enormous observation space poses significant computational challenges. To achieve real-time user interaction, we in fact use a simpified observation model for POMDP planning (Section 5.2.2). However, we use the full observation model in (Equation 3) in order to retain the accuracy in belief tracking (Section 5.2.3). This allows INGRESS to balance computational efficiency and accuracy in reasoning.

5.2.2 POMDP Planning To solve the POMDP, we search a *belief tree* (Figure 5). Each tree node represents a belief over the referred object. The root node of the tree corresponds to the current belief. A parent node and a child node, with associated belief b and b', are connected by an action-observation pair. For example, if the robot has initial belief b, takes the action of asking a question, and receives the user response as an observation, the robot's belief then becomes b'. The tree search produces an action with the highest expected total reward for the current belief.

We make two approximations to make the tree search fast. The first deals with the enormous observation space. Typically, fast online POMDP planning leverages sparsely sampled observations during the forward search (see, e.g., (Somani et al. 2013; Silver and Veness 2010)). In our case, language expressions sampled in a word-by-word fashion would be incoherent and irrelevant to the context. So instead, we use an approximate observation model which groups the observations according to the response utterance and effectively, ignores the description utterance:

$$p(z \mid s, a) \approx p(z_{\mathrm{r}} \mid x, a). \tag{4}$$

This approximation dramatically reduces the observation space size, while still allowing us to choose intelligently between information-gathering actions. Consider a simple scenario with two identical blue cups. The robot is instructed to "pick up the cup". Asking a self-referential question, e.g., "do you mean this blue cup?" would not change the belief and resolve the ambiguity. In contrast, asking a relational question, "do you mean this left cup?", and receiving a "yes" or "no" response helps in identifying the referred object.

The second approximation makes a reasonable assumption on the length of disambiguation dialogs. In everyday life, disambiguation typically takes no more than a few questions, as more questions may simply be annoying to most people. So we construct a belief tree of maximum depth $d = 4$.

The belief tree contains $\mathcal{O}(|A|^d|Z|^d)$ nodes, where $|A|$ is the size of the action space and $|Z|$ is the size of the observation space. In our case, $|A| = 3N$, for N candidate objects. With the approximations, $|Z| = 2$ and $d = 4$. The belief tree size is sufficiently small to allow for a full tree search to choose the best action at each time step in real time.

5.2.3 Belief Update After the robot asks a disambiguation question and receives an answer as the observation, it updates the belief, using the full observation model in (Equation 3):

$$b_t(x) = \frac{1}{\mathbb{Z}} \; p(z \mid x, a) \; b_{t-1}(x), \tag{5}$$

where $\mathbb{Z} = \sum_{x=1}^{N} p(z \mid x, a) \; b_{t-1}(x)$. The full observation model accounts for both response and description utterances

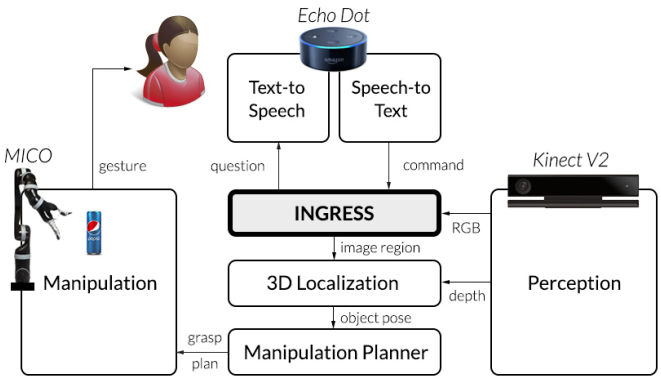

Figure 6. An overview of the system architecture.

and handles correcting descriptions, such as "no, the blue one". It uses the full information in the user's answer and tracks the belief more accurately. The computation cost is nevertheless modest, as belief update handles a single observation received, unlike planning, which reasons about many possible future observations.

6 System Implementation

To evaluate our approach, we implemented INGRESS on a robot manipulator, with voice input and RGB-D sensing. Below we briefly describe the system setup (Figure 6).

6.1 Visual Perception and Speech Recognition

Our grounding model takes in as input an RGB image and a textual referring expression, and outputs a 2D bounding box containing the referred object in the image (Figure 6). Our system uses a Kinect2 RGB-D camera for visual perception and an Amazon Echo Dot device to synthesize the referring expression from voice input.

6.2 Grounding Networks

The localization module for object detection uses a non-maximum suppression threshold of 0.7 and a final output threshold of 0.05 for minimal overlap between bounding boxes in uncluttered scenes.

S-LSTM and R-LSTM have a vocabulary size of $10,497$ and $2,020$, respectively. The maximum sequence length for both is 15 words.

6.3 Object Manipulation

Our system uses a 6-DOF Kinova MICO arm for object manipulation. It is currently capable of two high-level actions, PICKUP and PUTIT. For PICKUP, the system first uses the Kinect2 depth data corresponding to the selected 2D bounding box and localizes the referred object in 3D space. It then plans a top-down or a side grasp pose based on the object size, as well as a path to reach the pose. For PUTIT, the system similarly identifies the placement location. It moves the end-effector to position it directly above the desired location and then simply opens up the gripper. This simple set up is sufficient for our experiments. However, future works could integrate state-of-the-art methods for grasping and manipulating novel objects (Mahler et al. 2017).

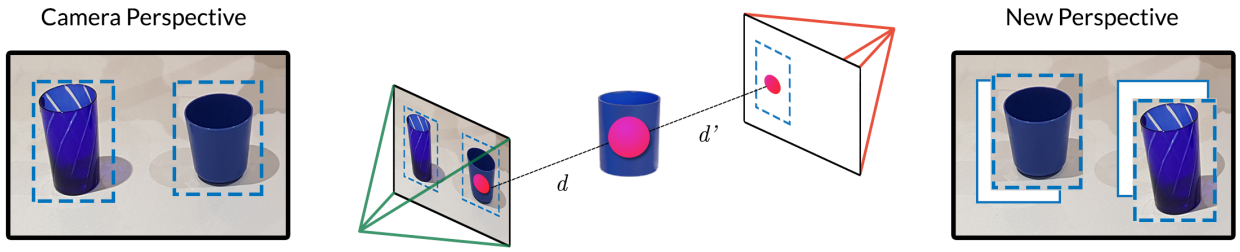

Figure 7. Perspective reprojection. For each bounding box localized in the camera perspective (left), a corresponding 3D centroid is computed from depth data. This 3D centroid is reprojected onto the image plane of the new perspective (right) using the camera's projection matrix and 6-DOF pose. The aspect ratio of the box is maintained, while the size of the box is scaled proportionally according to the ratio between distances d and d'. The reprojected bounding boxes are used to ground relationships with perspective constraints.

6.4 Software and Hardware Platform

The entire system (Fig. 6), with components for RGB-D visual perception, grounding, and manipulation planning, is implemented under the Robot Operating System (ROS) framework and runs on a PC workstation with an Intel i7 Quad Core CPU and an NVIDIA Titan X GPU. The grounding model runs on the GPU.

6.5 Perspective Correction

Referring expressions are conditioned on perspectives (Li et al. 2016b): object-centric (e.g., "the bottle next to the teddy bear"), user-centric (e.g., "the bottle on my left"), or robot-centric (e.g., "the bottle on your right"). Object-centric expressions are handled directly by the grounding model. User-centric and robot-centric expressions require special treatment. Handling perspectives reliably is a complex issue. Here we provide a solution dealing with the simple, common cases in a limited way. Given two detected viewpoints for the user and the robot perspective, the system associates a set of possessive keywords such as "my", "your", etc. with each viewpoint. It then matches the input expression against the keyword list to select a viewpoint and performs a corresponding geometric transformation of detected 2D image bounding boxes to the specified viewpoint frame (Figure 7).

In our experiments, "my" and "your" viewpoints are manually specified using an interactive 6-DOF marker tool in ROS (Robot-Operating System) RViz*. Future work could integrate Kinect-based people detection and tracking to allow dynamic updates of viewpoints.

7 Experiments

We evaluated our system under three settings. First, we evaluated for grounding accuracy and generalization to a wide variety of objects and relations on the RefCOCO dataset (Kazemzadeh et al. 2014). Next, we evaluated for generalization to rich language expressions in robot experiments with humans. In both cases, INGRESS outperformed UMD Refexp (Nagaraja et al. 2016). Finally, we evaluated for effectiveness of disambiguation and found that INGRESS-POMDP, through object-specific questions, sped up task completion by 1.9 times on average with respect to a generic-question baseline.

Dataset	HGT (%)		MCG (%)	
	UMD Refexp	INGRESS	UMD Refexp	INGRESS
Val	75.5	**77.0**	56.5	**58.3**
TestA	74.1	**76.7**	57.9	**60.3**
TestB	76.8	**77.7**	**55.3**	55.0

Table 1. Grounding accuracy of UMD Refexp and INGRESS on the RefCOCO dataset, with human-annotated ground-truth (HGT) object proposals and automatically generated MCG object proposals.

In uncluttered scenes with 10–20 objects, the overall voice-to-action cycle takes 2–5 seconds for voice-to-text synthesis, retrieving the synthesized text from Amazon's service, grounding, visual perception processing, and manipulation planning for picking or putting actions by the 6-DOF robot arm. In particular, grounding takes approximately 0.15 seconds.

7.1 RefCOCO Benchmark

The RefCOCO dataset contains images and corresponding referring expressions, which use both self-referential and relational information to uniquely identify objects in images.

*RViz Markers: http://wiki.ros.org/interactive_markers

Figure 8. Experimental setup for robot experiments.

The dataset covers a wide variety of different objects and is well suited for evaluating generalization to unconstrained object categories. Our evaluation measures the accuracy at which a model can locate an image region, represented as an image bounding box, given an expression describing it unambiguously.

We compared INGRESS with UMD Refexp (Nagaraja et al. 2016) on the RefCOCO dataset. UMD Refexp's approach to relational grounding is similar to that of INGRESS (see Section 4.3), but there are two key differences. First, UMD Refexp uses feature vectors from an image-net pre-trained VGG-16 network, whereas INGRESS uses captioning-trained feature vectors from the self-referential grounding stage. Second, for images with more than 10 object proposals, UMD Refexp randomly samples 9 candidates for relational grounding, while INGRESS only examines the pairs of objects proposals chosen by the self-referential grounding stage. Although INGRESS is trained with data from both Visual Genome and RefCOCO, we only evaluate on RefCOCO's test set, because Visual Genome does not contain relational expressions and UMD Refexp makes a strong-assumption on pair-wise annotations.

7.1.1 Procedure The RefCOCO dataset consists of a training set, a validation set (Val), and two test sets (TestA and Test B). TestA contains images with multiple people. TestB contains images with multiple instances of all other objects. TestA contains 750 images with 5657 expressions. TestB contains 750 images with 5095 expressions. Val contains 1500 images with 10834 expressions. Following UMD Refexp, we use both human-annotated ground-truth object proposals and automatically generated MCG proposals (Arbeláez et al. 2014) in our evaluation.

7.1.2 Results The results are reported in Table 1. The correctness of a grounding result is based on the overlap between the output and the ground-truth image regions. The grounding is deemed correct if the intersection-over-union (IoU) measure between the two region is greater than 0.5. Table 1 shows that INGRESS outperforms UMD Refexp in most cases, but the improvements are small. INGRESS adopts a two-stage grounding process in order to reduce the number of relevant object proposals processed in complex scenes. On average, the validation and test sets contain 10.2 ground-truth object proposals and 7.4 MCG object proposals per image. As the number of object proposals per image is small, the two-stage grounding process does not offer significant benefits.

We also observed that images containing people have greater improvement in accuracy than those containing only objects. This likely results from the large bias in the number of images containing people in the Visual Genome dataset (Krishna et al. 2016). Future work may build a more balanced dataset with a greater variety of common objects for training the grounding model.

7.2 Robot Experiments

We also assessed the performance of our grounding model in a realistic human-robot collaboration context and particularly, to study its ability in handling rich language expressions. In our experiments, a group of participants provided natural language instructions to a 6-DOF manipulator to pick and place objects (Figure 8).

Again, we compared INGRESS with UMD Refexp (Nagaraja et al. 2016). We also conducted an ablation study, which compared pure self-referential grounding (S-INGRESS) and the complete model with both self-referential and relational grounding. For S-INGRESS, we directly used the image region with the lowest cross-entropy loss from the self-referential stage. For INGRESS, we used the region chosen by the full model. Further, both S-INGRESS and INGRESS, used the object proposals generated by the self-referential stage, whereas UMD Refexp used MCG proposals (Arbeláez et al. 2014). All methods used a large number of object proposals. So the probability of randomly picking the referred object was very low.

7.2.1 Procedure Our study involved 16 participants (6 female, 10 male) recruited from a university community. All subjects were competent in spoken English. Each participant was shown 15 different scenarios with various household objects arranged in an uncluttered manner.

In each scenario, the experimenter asked the participant to describe a specific object to the robot. The experimenter gestured at the object without any language descriptions. Before instructing the robot, the subjects were given general guidelines to provide descriptions that are simple and unambiguous for this set of experiments aimed at assessing grounding accuracy. The only hard restriction was that perspectives e.g. 'my left', should be stated explicitly, but otherwise the experimenter made no attempt to intervene or alter the utterance if the user did not follow the guidelines. Upon receiving an expression, all 3 models (S-INGRESS, INGRESS, UMD Refexp) received the same image and expression as input, and 3 trials were run back-to-back on the robot in a fixed order. A trial was considered successful if the robot located the specified object on its first attempt. The success was assessed by the experimenter, without any feedback from the participant.

The average number of objects per scenario was 8. And the maximum number of identical objects was 3. The scenarios were carefully designed such that 66% required relational cues, 33% involved perspective taking, and 100% required self-referential information. For assessing perspectives, the participant was positioned at one of the four positions around the robot: front, behind, left, right. Also, since the models were trained on public datasets, all objects used in the experiments were 'unseen'. However, generic objects like apples and oranges had minimal visual differences to the training examples.

7.2.2 Results The results (Fig. 9) show that overall INGRESS (76.7%) significantly outperforms both S-INGRESS ($53.3\%, p < 0.001$ by t-test) and UMD Refexp ($31.3\%, p < 0.001$ by t-test). S-INGRESS is effective in locating objects based on self-referential information. However, it fails to infer relationships, as each image region is processed in isolation. While UMD Refexp in principle makes use of both self-referential and relational information, it performs poorly in real-robot experiments, particularly, in grounding self-referential expressions. UMD Refexp is trained on a relatively small dataset, RefCOCO, with mostly relational expressions. Its ability in grounding self-referential

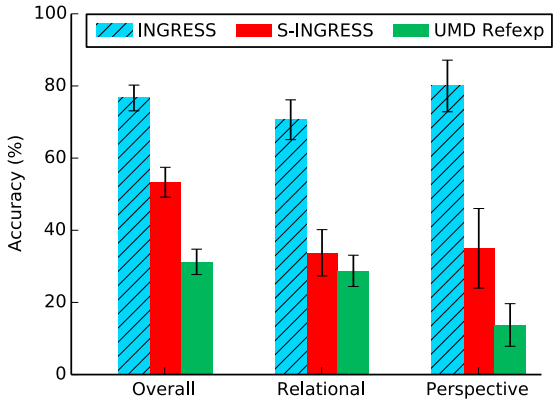

Figure 9. Grounding accuracy in robot experiments with humans. Error bars indicate 95% confidence intervals.

expressions is inferior to that of INGRESS and S-INGRESS. Further, INGRESS uses relevancy clustering to narrow down a set of object proposals for relationship grounding, whereas UMD Refexp examines a randomly sampled subset of object proposal pairs, resulting in increased errors. Finally, UMD Refexp is incapable of handling perspectives, as it is trained on single images without viewpoint information.

During the experiments, we observed that referring expressions varied significantly across participants. Even a simple object such as an apple was described in many different ways as "the red object", "the round object", "apple in middle", "fruit" etc. Likewise, relations were also described in many different ways. $41/240$ expressions from participants used non-binary relations, e.g., "the can in the middle", "the second can", etc.

Occasionally, participants used complex ordinality constraints, e.g., "the second can from the right on the top row". None of the models examined here, including INGRESS, can handle ordinality constraints. Other common failures include text labels and brand names on objects, e.g., "Pepsi".

7.3 Disambiguation

We conducted a user study to examine the effectiveness of INGRESS in asking disambiguating questions. Specifically, we examined two issues:

- Does our approach of asking object-specific questions improve grounding in terms of the time required to resolve ambiguities?
- Are the generated questions effective in obtaining the required additional information from the user?

The study compared the effectiveness of three methods: generic-question iteration (baseline), INGRESS-Heuristic, and INGRESS-POMDP. The baseline method generic-question iteration, is similar to the work of Hatori et al. (2018). There, the robot exhaustively points at objects while asking a generic question "do you mean this object?", and expects a yes/no answer from the user. In contrast, INGRESS-Heuristic asks object-specific questions (e.g., "do you mean this red cup?") based on the heuristic described in Section 5.1, and the user may provide a correcting response (e.g., "no, the red cup on the right"). Similarly, INGRESS-POMDP also asks object-specific questions, except the question asking is based on the decision-theoretic model described in Section 5.2.

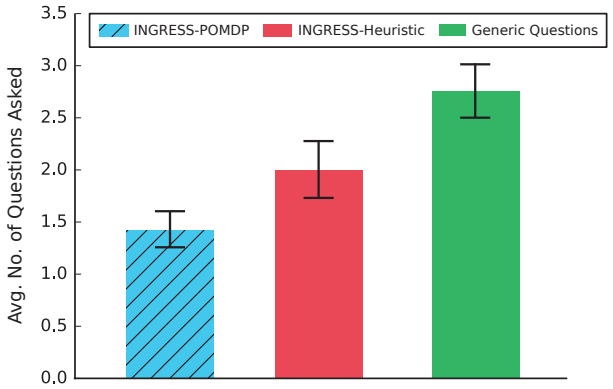

Figure 10. Disambiguation performance measured by the average number of questions asked (lower is better). Error bars indicate 95% confidence intervals.

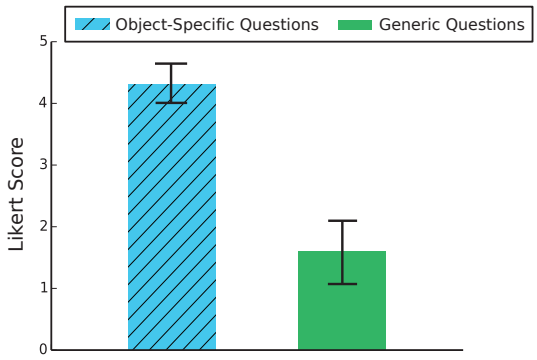

Figure 11. User survey on the robot's effectiveness in communicating the additional information required for disambiguation (higher is better). Error bars indicate 95% confidence intervals.

7.3.1 Procedure The study was conducted with 24 participants. 8 participants for the baseline condition, 8 participants for INGRESS-Heuristic, and 8 participants for INGRESS-POMDP. Each subject was shown 10 different scenarios with various household object. For each scenario, the experimenter initiated the trial by giving the robot an ambiguous instruction e.g., "pick up the cup" in scene with a red cup, blue cup, green cup and yellow cup. For INGRESS-Heuristic, the robot pointed to one of the candidate objects, and asked a question. For INGRESS-POMDP, the robot directly asked a question since it maintains an explicit belief over objects without having to point at them (see Section 5.2.1). Then the participant was instructed to correct the robot to pick another object of the same category. Again, the experimenter gestured at the desired object without any verbal communication. For the baseline, the participants could only use yes/no corrections. For INGRESS-Heuristic and INGRESS-POMDP, they could correct the ambiguous expression with additional information e.g., "no, the red cup" or "no, the cup on the left".

The average number of objects per scenario was 4.6. These scenarios were different to that in Shridhar and Hsu (2018) in that they were more ambiguous and typically required asking 1-2 more questions. We conducted a total of 240 trials. In all successful trials, the participants were eventually able to correct the robot to find the object specified by the experimenter. If the robot stopped asking questions and picked the wrong object, the trial was recorded as a failure.

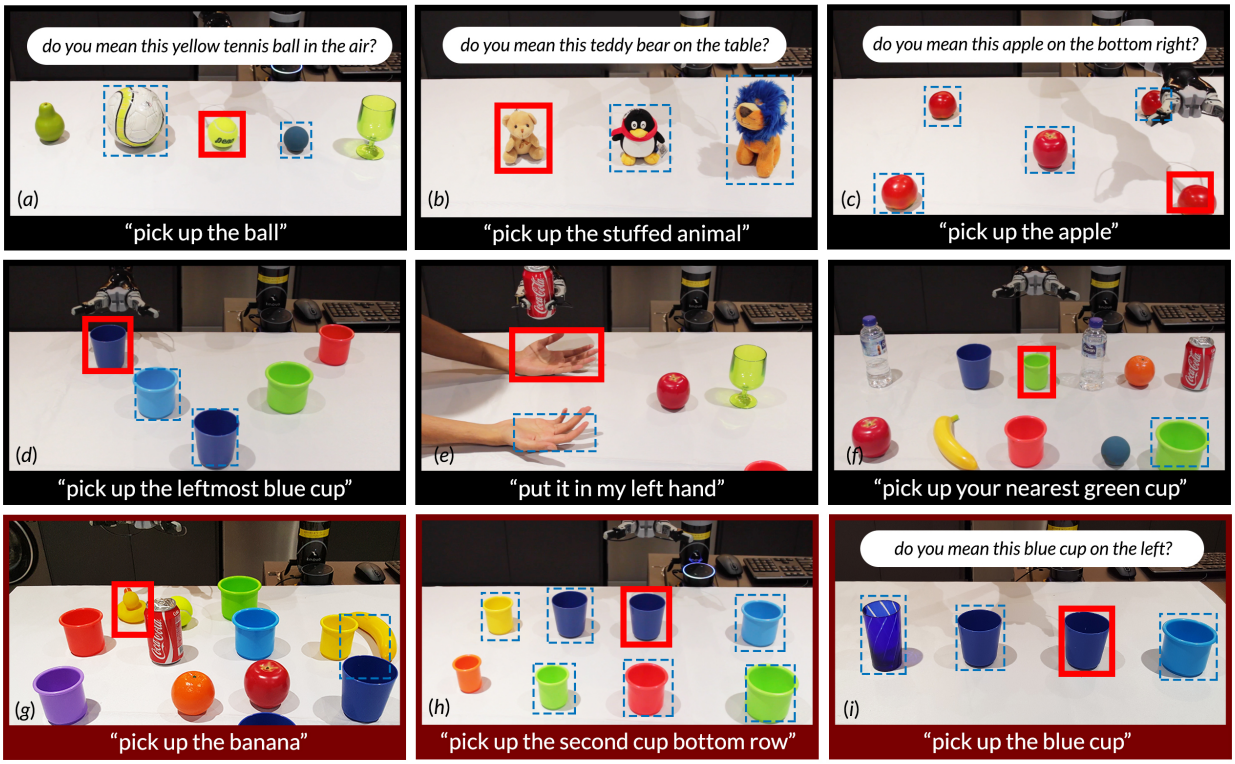

Figure 12. A sample of interactive grounding results. Red boxes indicate the objects chosen by INGRESS. Blue dashed boxes indicate candidate objects. The first two rows show successful results and disambiguation questions. The last row shows some failure cases.

7.3.2 Results Figure 10 shows that INGRESS-POMDP (average 1.45 questions) is more efficient in disambiguation than the baseline method (average 2.75 questions), with $p < 0.001$ by the t-test. INGRESS-POMDP is also faster than INGRESS-Heuristic (average 1.99 questions), with $p = 0.01$ by the t-test. While the differences appear small, they are statistically significant. More importantly, the difference is likely practically significant. The results indicate that INGRESS-POMDP usually asks 1 to 2 disambiguation questions, which are common in our daily life. However, INGRESS-Heuristic asks 2 questions on the average and sometimes 3 or more questions, which would be annoying. While the difference of one question appears small, it is important for fluid user interaction. The success rates of INGRESS-POMDP (89%) and INGRESS-Heuristic (88%) were similar. So INGRESS-POMDP reduced the number of questions asked without reducing the overall accuracy. In effect, INGRESS-POMDP's questions are more effective in reducing the robot's uncertainty in the referred object and provide potentially more natural user interaction. Further, in terms of raw execution time, INGRESS-POMDP was significantly faster than the other two methods, since the question asking was purely verbal and did not involve a pointing gesture. The pointing gesture typically takes an additional 1–4 seconds for planning and execution.

We also conducted a post-experiment survey and asked participants to rate the agreement question "the robot is effective in communicating what additional information is required for disambiguation" on a 5-point Likert scale. Since INGRESS-POMDP and INGRESS-Heuristic are similar in that they both ask object-specific questions, we administered the survey only to INGRESS-POMDP and yes/no baseline participants. Again, our method of asking object-specifc questions with INGRESS-POMDP scores much higher than the baseline method, 4.25 versus 1.63 with a significance of $p < 0.001$ by the Kruskal-Wallis test (Figure 11).

During the experiment, we noted that participants used some back-referring anaphora, e.g., "no, the *other* cup", which occurred in $11/160$ expressions. Also, the participants quite often mimicked the language that the robot used. For example, when robot asks "do you mean this apple on the *bottom right*?", the user responds "no, the apple on the *top left*". A few participants also commented that they would not have used certain descriptions, e.g., "top left", if it were not for the robot's question. This is consistent with the psycholinguistic phenomenon of *linguistic accommodation* (Gallois and Giles 2015), in which participants in a conversation adjust their language style and converge to a common one. It is interesting to observe here that linguistic accommodation occurs not only between humans and humans, but also between humans and robots. Future works could study this in more detail.

7.4 Examples

Figure 12 shows a sample of interactive grounding results. Figure 12(*a–b*) highlight rich questions generated by INGRESS. The questions are generally clear and discriminative, though occasionally they contain artifacts, e.g., "ball in the air" due to biases in the training dataset. Further, although our system is restricted to binary relations, Figure 12(*c–d*) show some scenes that contain complex, seemingly non-binary relationships. The referred apple is at the bottom right corner of the entire image,

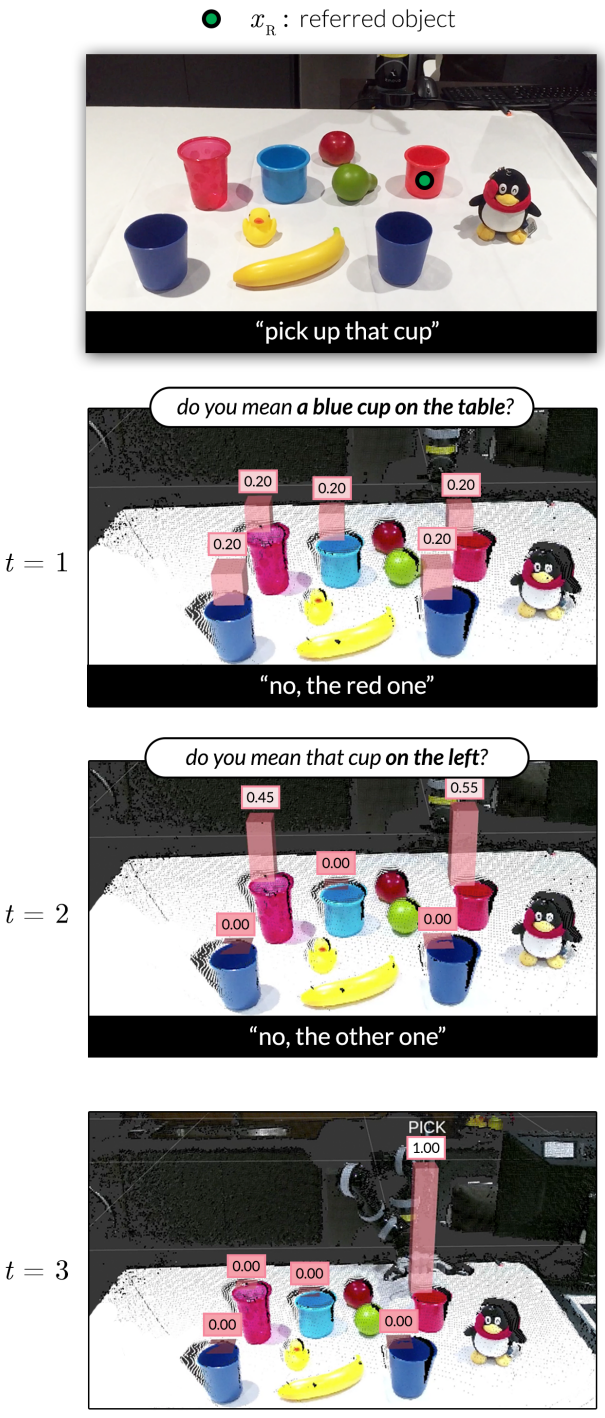

Figure 13. A sample scenarios for disambiguation with INGRESS-POMDP. The red bars above the objects plot the robot's belief on the referred object before asking the question.

treated as a single object. Likewise, the selected blue cup is the closest one to the left edge of the image. Figure 12(*e–f*) showcase user-centric and robot-centric perspective corrections, respectively. They enable users to adopt intuitive viewpoints such as "my left". Figure 12(*g–i*) show some common failures. INGRESS has difficulty with cluttered environments. Partially occluded objects, such as Figure 12(*g*), often result in false positives. It also cannot handle complex relationships, such as Figure 12(*h*), which requires counting ("third") or grouping objects ("row", "all four"). Figure 12(*i*) is an interesting case. The user's intended object is the second cup from the left, but the input expression is ambiguous. While the generated question is not discriminative, the robot arm's pointing gesture helps to identify the correct object after two questions.

Figure 13 shows a sample scenario for disambiguation with INGRESS-POMDP. The robot is instructed to "pick up that cup" in a scene with three blue cups, two red cups and a few other objects. Initially ($t = 1$), the robot has a uniform belief over the cups and decides to ask a self-referential question "do you mean a blue cup on the table?", as the question provides the maximum expected gain in information. When the user responds "no, the red one", the robot eliminates all "blue cups", and the belief concentrates on the two remaining red cups ($t = 2$). The robot then asks a relational question "do you mean the cup on the left?", as asking more about self-referential attributes does not provide additional information. The user responds "no, the other one". The robot eliminates the left red cup and directly picks right one ($t = 3$). INGRESS-POMDP generates all the questions automatically without any handcrafted rules. It is also important to note that the history of interactions is critical for grounding responses such as "no, the red one". The underlying belief representation maintained by INGRESS-POMDP enables the robot to infer that the user is referring to the "red cup" and not the "red apple" in the scene. For more examples, see the accompanying video at `http://bit.ly/INGRESSpomdp`

8 Discussion

While the experimental results are promising, INGRESS has several limitations (see Figure 12). First, it handles only binary relations between the referred and context objects. While this assumption is sufficient to cover the common situations, it is not easy to scale up the network to handle situations involving tertiary or n-nary relations. Recent work on relation networks (Santoro et al. 2017) grounds complex relationships by learning a joint embedding of all pairwise permutations of objects. Training such a network on complex relationship corpora (Johnson et al. 2017a) may help. Further, integrating non-verbal cues such as gestures and gaze (Palinko et al. 2016; Fischer and Demiris 2016) may reduce the need for interpreting complex instructions as some ambiguities can be resolved through body language cues. Second, INGRESS relies on keyword matching to understand perspectives. Sometimes these keywords might not be explicitly stated in the referring expression. So augmenting the training set with perspective-bearing expressions could allow the system to generalize better. Third, the clustering components of the grounding model are currently hard-coded. A more sophisticated method such as spectral clustering (Cherouvim and Papadopoulos 2005) may improve performance. Another possibility is to treat the clustering components as neural network modules, the grouping of relevant objects can be learned simultaneously with other components. Lastly, INGRESS cannot handle cluttered environments with partially occluded objects. Systematically moving away objects to reduce uncertainty (Li et al. 2016a) may help.

9 Conclusion

We have presented INGRESS, a neural network model for grounding unconstrained natural language referring expressions. By training the network on large datasets, INGRESS handles a wide variety of everyday objects. In case of ambiguity, INGRESS-POMDP asks object-specific disambiguating questions in a principled manner. The system a state-of-the-art method substantially in robot experiments with humans and generated interesting interactions for disambiguation of referring expressions. Even though we are far from achieving a perfect shared understanding of the world between humans and robots, we hope that our work is a step in this direction. It points to several important issues for further investigation (Section 8). An even more important, but different direction is to connect with the grounding of verbs (Kollar et al. 2010) to expand the repertoire of robot actions, as well as a range of other interesting language grounding problems (Paul et al. 2017; Nyga et al. 2018).

Acknowledgments

We thank members of the Adaptive Computing Lab at NUS for thoughtful discussions. We also thank the anonymous reviewers for their careful reading of the manuscript and many suggestions that have helped to improve the paper. This work was supported by the NUS School of Computing Strategic Initiatives.

References

Andreas J, Rohrbach M, Darrell T and Klein D (2016) Neural module networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. pp. 39–48.

Arbeláez P, Pont-Tuset J, Barron JT, Marques F and Malik J (2014) Multiscale combinatorial grouping. In: *Proceedings of Computer Vision and Pattern Recognition (CVPR)*.

Arkin J and Howard TM (2018) Experiments in proactive symbol grounding for efficient physically situated human-robot dialogue. *SIGDIAL Special Session on Physically Situated Dialogue (RoboDIAL-18)* .

Banerjee S and Lavie A (2005) Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In: *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*. pp. 65–72.

Bisk Y, Yuret D and Marcu D (2016) Natural language communication with robots. In: *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*. pp. 751–761.

Cherouvim N and Papadopoulos E (2005) Single actuator control analysis of a planar 3DOF hopping robot. In: *Proceedings of Robotics: Science and Systems*. Cambridge, USA. DOI: 10.15607/RSS.2005.I.020.

Clark H et al. (1991) Grounding in communication. *Perspectives on socially shared cognition* 13: 127–149.

Das A, Kottur S, Moura JM, Lee S and Batra D (2017) Learning cooperative visual dialog agents with deep reinforcement learning. *Proceedings of Computer Vision and Pattern Recognition (CVPR)* .

De Vries H, Strub F, Chandar S, Pietquin O, Larochelle H and Courville AC (2017a) Guesswhat?! visual object discovery through multi-modal dialogue. In: *Proceedings of Computer Vision and Pattern Recognition (CVPR)*.

De Vries H, Strub F, Mary J, Larochelle H, Pietquin O and Courville AC (2017b) Modulating early visual processing by language. In: *Advances in Neural Information Processing Systems (NIPS)*. pp. 6594–6604.

Doshi F and Roy N (2008) Spoken language interaction with model uncertainty: an adaptive human–robot interaction system. *Connection Science* 20(4): 299–318.

Eppner C, Höfer S, Jonschkowski R, Martin R, Sieverling A, Wall V and Brock O (2016) Lessons from the amazon picking challenge: Four aspects of building robotic systems. In: *Proceedings of Robotics: Science and Systems (RSS)*.

Fischer T and Demiris Y (2016) Markerless perspective taking for humanoid robots in unconstrained environments. In: *IEEE International Conference on Robotics and Automation (ICRA)*.

FitzGerald N, Artzi Y and Zettlemoyer L (2013) Learning distributions over logical forms for referring expression generation. In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. pp. 1914–1925.

Fried D, Hu R, Cirik V, Rohrbach A, Andreas J, Morency LP, Berg-Kirkpatrick T, Saenko K, Klein D and Darrell T (2018) Speaker-follower models for vision-and-language navigation. *Advances in Neural Information Processing Systems (NIPS)* .

Gallois C and Giles H (2015) Communication accommodation theory. *International Encyclopedia of Language and Social Interaction* .

Golland D, Liang P and Klein D (2010) A game-theoretic approach to generating spatial descriptions. In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP). Stroudsburg, PA, USA: Association for Computational Linguistics.

Guadarrama S, Riano L, Golland D, Go D, Jia Y, Klein D, Abbeel P, Darrell T et al. (2013) Grounding spatial relations for human-robot interaction. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

Hatori J, Kikuchi Y, Kobayashi S, Takahashi K, Tsuboi Y, Unno Y, Ko W and Tan J (2018) Interactively picking real-world objects with unconstrained spoken language instructions. *IEEE International Conference on Robotics and Automation (ICRA)* .

Hemachandra S and Walter MR (2015) Information-theoretic dialog to improve spatial-semantic representations. In: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 5115–5121.

Hu R, Rohrbach M, Andreas J, Darrell T and Saenko K (2017) Modeling relationships in referential expressions with compositional modular networks. In: *Proceedings of Computer Vision and Pattern Recognition (CVPR)*.

Hu R, Xu H, Rohrbach M, Feng J, Saenko K and Darrell T (2016) Natural language object retrieval. In: *Proceedings of Computer Vision and Pattern Recognition (CVPR)*.

Huo Z and Skubic M (2016) Natural spatial description generation for human-robot interaction in indoor environments. In: *IEEE International Conference on Smart Computing*

(SMARTCOMP).

Johnson J, Hariharan B, van der Maaten L, Fei-Fei L, Zitnick CL and Girshick R (2017a) Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In: *Proceedings of Computer Vision and Pattern Recognition (CVPR)*.

Johnson J, Hariharan B, van der Maaten L, Hoffman J, Fei-Fei L, Zitnick CL and Girshick R (2017b) Inferring and executing programs for visual reasoning. In: *ICCV*.

Johnson J, Karpathy A and Fei-Fei L (2016) Densecap: Fully convolutional localization networks for dense captioning. In: *Proceedings of Computer Vision and Pattern Recognition (CVPR)*.

Karpathy A and Fei-Fei L (2015) Deep visual-semantic alignments for generating image descriptions. In: *Proceedings of Computer Vision and Pattern Recognition (CVPR)*. pp. 3128–3137.

Kazemzadeh S, Ordonez V, Matten M and Berg TL (2014) Referitgame: Referring to objects in photographs of natural scenes. In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Kollar T, Tellex S, Roy D and Roy N (2010) Toward understanding natural language directions. In: *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI)*.

Krishna R, Zhu Y, Groth O, Johnson J, Hata K, Kravitz J, Chen S, Kalantidis Y, Li LJ, Shamma DA, Bernstein M and Fei-Fei L (2016) Visual genome: Connecting language and vision using crowdsourced dense image annotations. In: *International Journal of Computer Vision (IJCV)*.

Krishnamurthy J and Kollar T (2013) Jointly learning to parse and perceive: Connecting natural language to the physical world. *Transactions of the Association for Computational Linguistics* 1: 193–206.

Li JK, Hsu D and Lee WS (2016a) Act to see and see to act: Pomdp planning for objects search in clutter. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

Li S, Scalise R, Admoni H, Rosenthal S and Srinivasa SS (2016b) Spatial references and perspective in natural language instructions for collaborative manipulation. In: *IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*.

Mahler J, Liang J, Niyaz S, Laskey M, Doan R, Liu X, Aparicio J and Goldberg K (2017) Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. In: *Proceedings of Robotics: Science and Systems (RSS)*.

Mao J, Huang J, Toshev A, Camburu O, Yuille AL and Murphy K (2016) Generation and comprehension of unambiguous object descriptions. In: *Proceedings of Computer Vision and Pattern Recognition (CVPR)*.

Matuszek C, FitzGerald N, Zettlemoyer L, Bo L and Fox D (2012) A Joint Model of Language and Perception for Grounded Attribute Learning. In: *Proceedings of the International Conference on Machine Learning (ICML)*.

Mei H, Bansal M and Walter MR (2016) Listen, attend, and walk: Neural mapping of navigational instructions to action sequences. In: *Thirtieth AAAI Conference on Artificial Intelligence*.

Nagaraja VK, Morariu VI and Davis LS (2016) Modeling context between objects for referring expression understanding. In: *European Conference on Computer Vision (ECCV)*.

Neisser U (2014) *Cognitive psychology*. Psychology Press.

Nyga D, Roy S, Paul R, Park D, Pomarlan M, Beetz M and Roy N (2018) Grounding robot plans from natural language instructions with incomplete world knowledge. In: *Proceedings of The 2nd Conference on Robot Learning (CoRL)*, volume 87. PMLR, pp. 714–723.

Palinko O, Rea F, Sandini G and Sciutti A (2016) Robot reading human gaze: Why eye tracking is better than head tracking for human-robot collaboration. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

Pangercic D, Pitzer B, Tenorth M and Beetz M (2012) Semantic object maps for robotic housework-representation, acquisition and use. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

Pateras C, Dudek G and De Mori R (1995) Understanding referring expressions in a person-machine spoken dialogue. In: *IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*.

Paul R, Arkin J, Roy N and Howard T (2016) Efficient grounding of abstract spatial concepts for natural language interaction with robot manipulators. In: *Proceedings of Robotics: Science and Systems (RSS)*.

Paul R, Barbu A, Felshin S, Katz B and Roy N (2017) Temporal grounding graphs for language understanding with accrued visual-linguistic context. In: *International Joint Conferences on Artificial Intelligence (IJCAI)*.

Santoro A, Raposo D, Barrett DG, Malinowski M, Pascanu R, Battaglia P and Lillicrap T (2017) A simple neural network module for relational reasoning. In: *Advances in Neural Information Processing Systems (NIPS)*.

Shridhar M and Hsu D (2018) Interactive visual grounding of referring expressions for human-robot interaction. In: *Proceedings of Robotics: Science and Systems (RSS)*.

Silver D and Veness J (2010) Monte-carlo planning in large pomdps. In: Lafferty JD, Williams CKI, Shawe-Taylor J, Zemel RS and Culotta A (eds.) *Advances in Neural Information Processing Systems (NIPS)*. Curran Associates, Inc., pp. 2164–2172.

Somani A, Ye N, Hsu D and Lee WS (2013) Despot: Online pomdp planning with regularization. In: *Advances in Neural Information Processing Systems (NIPS)*. pp. 1772–1780.

Tellex S, Knepper R, Li A, Rus D and Roy N (2014) Asking for help using inverse semantics. In: *Proceedings of Robotics: Science and Systems (RSS)*.

Tellex S, Kollar T, Shaw G, Roy N and Roy D (2010) Grounding spatial language for video search. In: *ICMI-MLMI*. ACM, p. 31.

Vinyals O, Toshev A, Bengio S and Erhan D (2015) Show and tell: A neural image caption generator. In: *Proceedings of Computer Vision and Pattern Recognition (CVPR)*. pp. 3156–3164.

Werning M, Hinzen W and Machery E (2012) *The Oxford handbook of compositionality*. Oxford University Press.

Whitney D, Rosen E, MacGlashan J, Wong LL and Tellex S (2017) Reducing errors in object-fetching interactions through social feedback. In: *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 1006–1013.

Williams JD and Young S (2007) Partially observable markov decision processes for spoken dialog systems. *Computer Speech & Language* 21(2): 393–422.

Yu L, Tan H, Bansal M and Berg TL (2017) A joint speaker-listener-reinforcer model for referring expressions. In: *Proceedings of Computer Vision and Pattern Recognition (CVPR)*.