Spatial Referring Expression Generation for HRI: Algorithms and Evaluation Framework

Lars Kunze
Oxford Robotics Institute
Dept. of Engineering Science
University of Oxford
United Kingdom
lars@robots.ox.ac.uk

Tom Williams
Human-Robot Interaction Lab
Dept. of Computer Science
Tufts University
United States of America
williams@cs.tufts.edu

Nick Hawes
Intelligent Robotics Lab
School of Computer Science
University of Birmingham
United Kingdom
n.a.hawes@cs.bham.ac.uk

Matthias Scheutz
Human-Robot Interaction Lab
Dept. of Computer Science
Tufts University
United States of America
mscheutz@cs.tufts.edu

Abstract

The ability to refer to entities such as objects, locations, and people is an important capability for robots designed to interact with humans. For example, a referring expression (RE) such as "Do you mean the box on the left?" might be used by a robot seeking to disambiguate between objects. In this paper, we present and evaluate algorithms for Referring Expression Generation (REG) in small-scale situated contexts. We first present data regarding how humans generate small-scale spatial referring expressions (REs). We then use this data to define five categories of observed small-scale spatial REs, and use these categories to create an ensemble of REG algorithms. Next, we evaluate REs generated by those algorithms and by humans both subjectively (by having participants rank REs), and objectively, (by assessing task performance when participants use REs) through a set of interrelated crowdsourced experiments. While our machine generated REs were subjectively rated lower than those generated by humans, they objectively significantly outperformed human REs. Finally, we discuss the main contributions of this work: (1) a dataset of images and REs, (2) a categorization of observed small-scale spatial REs, (3) an ensemble of REG algorithms, and (4) a crowdsourcing-based framework for subjectively and objectively evaluating REG.

1 Introduction

Many tasks in Human-Robot Interaction (HRI) require robots to use natural language (NL) to refer to objects, places, or people in their environment, a task known as *Referring Expression Generation* (REG). Hence, robots capable of automatically generating compact REs based on interpreted scenes can have a huge impact in HRI. For example, they could generate scene descriptions in security reports (Hawes et al. 2016), document scientific experiments[1], and assist people in domestic settings. Imagine, for example, a robot that is asked to fetch a tea box from a kitchen (Figure 1). If there are many boxes in the scene, the robot may need to ask which box was meant, using a referring expression (RE) such as:

1. Do you mean the box in the middle?
2. Do you mean the box that is to the right of the box that is close to the mug?

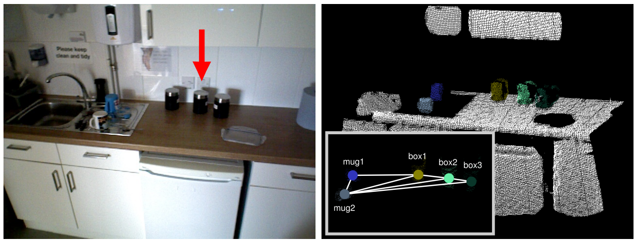


Figure 1: Example scenario. Left: real-world kitchen scene perceived by a robot (red arrow points to the target object). Right: scene segmentation (based on RGBD) and interpretation used for generating a referring expression: "Do you mean the box in the middle?"

While both REs describe the same object, the second requires more cognitive effort to understand. The REG task can thus be formulated as the process of finding a RE that uniquely identifies the target object while minimizing the expected cognitive effort needed to process that RE.

An RE may use various relations and properties such as type, size, color, or texture. In this work, we focus on REs that use *qualitative spatial relations* (QSRs), such as *Left* and *Close*, that hold between the *target* and other *landmark objects*. By using QSRs, the world is abstracted into a smaller qualitative state space in which REs hold. Moreover, these qualitative relations can be easier mapped to expressions in Natural Language as metric details are abstracted away. However, *qualitative spatial REs* may assume diverse forms, and there is no one form that is maximally appropriate in all situations. For example, in some situations it may be best to use a spatial RE which describes the target with a relation to some landmark, such as "the mug next to the sink", whereas if there are two mugs on opposite sides of the sink it may be better to say "the mug between the sink and the box". This suggests the need for an REG solution in which different *classes* of spatial relations may be selected based on context. We address this problem with a solution in which *context* is used to select between candidate REs generated by an ensemble of REG experts, each of which is capable of generating a single class of spatial RE.

The rest of this paper proceeds as follows. After discussing related work in Section 2, we discuss in Section 3 how we generated artificial tabletop scenes in order to acquire a corpus

[1] The Smart Wet Lab Assistant: `https://istc-pc.washington.edu/?page_id=303`

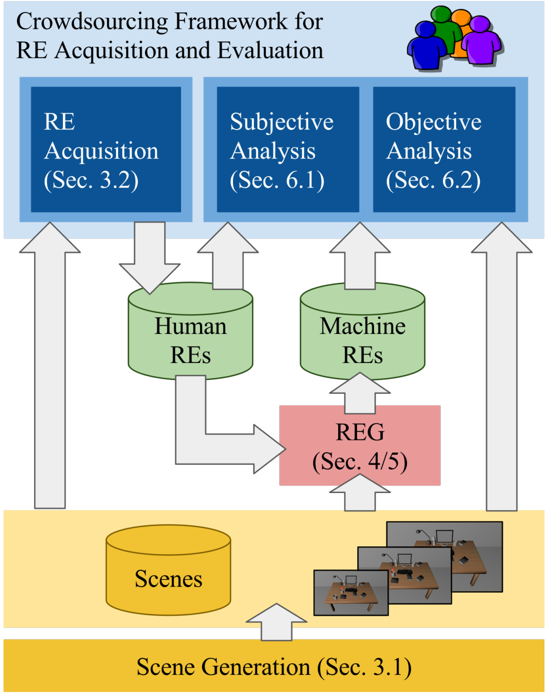

Crowdsourcing Framework for
RE Acquisition and Evaluation
RE Acquisition (Sec. 3.2)
Subjective Analysis (Sec. 6.1)
Objective Analysis (Sec. 6.2)
Human REs
Machine REs
REG (Sec. 4/5)
Scenes
Scene Generation (Sec. 3.1)

Figure 2: Overview of crowdsourcing framework for acquiring and evaluating REs. Given a set of scenes (Section 3.1) the framework can be used to both crowdsource natural REs from humans (Section 3.2) and generate REs based on the geometric and qualitative relations between objects within the scenes (Section 4). The resulting human and machine generated REs can be evaluated using a subjective and/or an objective analysis (Section 6).

of human-generated REs, and then present a categorization of those REs. In Section 4, we present a set of algorithms and heuristics (uniformly referred to as "algorithms") that generate different types of REs from this categorization, based on qualitative scene descriptions. We then present two strategies for contextually selecting a class of RE in Section 5. In Section 6, we introduce a novel crowdsourcing framework for the evaluation of REG algorithms, and present an empirical evaluation of the task effectiveness (an objective measure) and user preference (a subjective measure) of the presented algorithms. The presented framework (Figure 2) can be used as a common evaluation paradigm for future REG algorithms. Finally, we conclude with a discussion in Section 7, and of future work in Section 8.

2 Related Work

The problem of REG dates back to early work on Natural Language Processing (Winograd 1972). A review of REG algorithms and evaluation methods can be found in (Krahmer and van Deemter 2012). In (Dale and Reiter 1995), classic REG algorithms are discussed, including the *Full Brevity*, *Greedy Heuristic*, and *Incremental Algorithm* (IA), which differ primarily in how *distinguishing* properties are selected. The IA, for example, selects properties using a *preference ordering*, e.g. $type < color < size$. As we will describe, one algorithm within our ensemble is based on the version of the IA that was extended to handle binary *relations* (e.g., "left") (Kelleher and Kruijff 2005; 2006). Two of our other algorithms go beyond this to handle *n-ary* relations as well (e.g., *in-the-middle*). In general, all of our presented algorithms are based on a graph-based scene representation (Krahmer, van Erk, and Verleg 2003).

In addition to our ensemble of experts, we present an REG *evaluation* framework. Most approaches to evaluating REG algorithms (Viethen and Dale 2006) compare REs generated by algorithms with those found in text corpora (Krahmer and van Deemter 2012). These evaluation methods are problematic, however, because NL is flexible, and a RE may be realized in many equally acceptable forms. Our evaluation thus follows the path taken by (Viethen and Dale 2006), who argue in favor of a *task-based* evaluation. We thus *objectively* evaluate our approach with a task-based evaluation in which participants must identify referenced objects in scenes generated using qualitative spatial relationships (Fisher et al. 2012; Merrell et al. 2011), similar to that in (Belz 2008) and in the spirit of the *GIVE Challenge* (Koller et al. 2010). Unlike the GIVE Challenge, however, we (1) use a multi-stage evaluation, (2) evaluate machine-generated REs with respect to human-generated ones, and (3) supplement our task-based evaluation with a ranking task that subjectively evaluates the perceived naturalness of our generated REs (cf. (Gatt, Belz, and Kow 2009)). This evaluation is realized via crowdsourcing. Recently, crowdsourcing platforms have been used for a variety of tasks, including image labeling (Russell et al. 2008), commonsense knowledge acquisition (Gupta and Kochenderfer 2004), and language understanding (Kazemzadeh et al. 2014a; Tellex et al. 2011). In (Fang et al. 2013), REG was evaluated using Amazon's Mechanical Turk (AMT); but while they used AMT only to evaluate the effectiveness of generated REs, we also use it for the wider variety of REG evaluation and benchmarking tasks that comprise our framework.

3 Initial Data Collection

The initial motivation for this work was a desire to understand and generate REs in small-scale HRI scenarios. The REG algorithms and evaluation we present build on an initial study where users were asked to provide commands to pick up objects in tightly controlled machine-generated scenes.

3.1 Generation of Desktop Scenes

We generated a set of artificial desktop scenes using the MORSE simulator (Lemaignan et al. 2014). To obtain realistic scenes, we used object statistics of real-world office desks (Kunze, Burbridge, and Hawes 2014). The statistics provided information about the presence of an object, its location on a desk, and its qualitative spatial relations to other objects.

First, we sampled a number of objects (e.g., keyboards, monitors, laptops, lamps, cups, books, and bottles) to appear[2], at least one of which (a keyboard, monitor, or laptop)

[2] As we are mainly interested in spatial REs, we only used monochromatic objects to prevent subjects from using color- and texture-based features.

functions as a *landmark*. The *landmark* is placed on the desk according to its object-specific spatial distribution. Second, we sampled a set of qualitative spatial relations (QSRs) such as *Left* and *Close* between the landmark and all other objects, which were transformed into metric object poses using a generative model of the Ternary Point Calculus (TPC) (Moratz, Nebel, and Freksa 2003). A physics engine is used to ensure that generated scene configurations are physically possible.

Finally, in each scene an object was selected as target object and an image was then generated. In the image, the selected object was denoted with a red arrow (cf. Figure 5). Overall we generated 20 scenes for the experiments.

3.2 Collection of Commands and Categorization of Small Scale Spatial REs

Twenty participants (9 male, 11 female) were tasked with generating the best possible command to pick up the selected object in each scene. Participants ranged in age from 20 to 59 (M=35.3, SD=9.44). Of these, ten completed the survey and made an attempt to disambiguate the target objects, resulting in a total of 200 REs. As we will describe in the next section, participants' REs generally fell into five categories:

(1) *Type*: REs such as "Pick up the cup." These refer only to the target, and comprised 6.6% of all REs.

(2) *Relative*: REs such as "Get the bottle that is in front of the keyboard." These REs refer to the target with respect to the direction relative to some other object, and comprised 21.9% of all REs.

(3) *Set-Relative*: REs such as "Pick up the cup in between the keyboard and the computer screen" or "Pick up the second bottle from the right". These refer to the target relative to some set of other objects, and comprised 19.2% of all REs.

(4) *Proximal*: REs such as "Pick up the mug closest to the book." These refer to the target based on its proximity to *some other object*, and comprised 20.4% of all REs.

(5) *Distal*: REs such as "Pick up the furthest bottle." These refer to the target as being furthest from some object or other entity (such as the scene viewer) *in a given direction*, and comprised 31.8% of all REs.

While we saw high variance of class usage, certain classes saw higher context-specific usage. For example, *Type* REs were often seen when there were no objects of the same type as the target in the scene, and *Set-Relative* REs were often seen in cluttered environments. This presents an opportunity for a new direction in REG algorithms for HRI: instead of using a single algorithm to generate a RE for a given object, it may be more prudent to first determine which *class* of RE is most appropriate, and then use a RE of that class generated by a class-specific REG algorithm. In the next section, we propose a collection of such REG algorithms.

4 Algorithms for RE Generation

In this section we describe the generation of different classes of REs. As the majority of the proposed REG algorithms makes use of qualitative spatial relations (QSRs), we first explain how different QSRs are generated.[3]

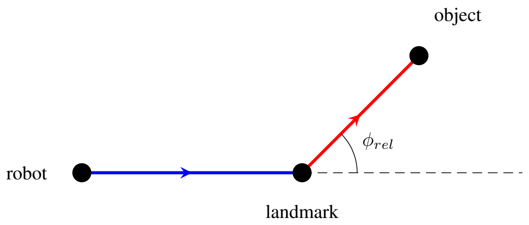


Figure 3: The relative angle is defined by the reference axis specified by *robot* and *landmark*, and the *object*. The example above illustrates a situation where the *object* is left and behind of the *landmark*.

4.1 Qualitative Spatial Relations

The REG algorithms presented in this work build on easily expressible QSRs such as *Left* and *Close*, with direction and distance between objects calculated using the *Ternary Point Calculus* (TPC) (Moratz, Nebel, and Freksa 2003). The TPC is so named for its use of three points: the *origin* (e.g., the position of a robot), the *referent* (e.g., a target to be described), and the *relatum* (e.g., a landmark with respect to which to describe the referent). The origin and relatum (hereafter *robot* and *landmark*) define a reference axis which partitions the surrounding space, allowing a spatial relationship to be defined by the partition in which the referent (hereafter *object*) lies. To determine this partition, the relative angle ϕ_{rel} between the robot-landmark axis and the object (Figure 3) is calculated as:

$$\phi_{rel} = \tan^{-1}\frac{y_{obj} - y_{land}}{x_{robj} - x_{land}} - \tan^{-1}\frac{y_{land} - y_{robot}}{x_{land} - x_{robot}} \quad (1)$$

When calculating a robot-landmark axis, we assume a robot is standing about two meters before a scene. According to the *relative angle* and the *relative radius*, we label the relations between all objects and the landmark as *Left*, *Right*, *InFront*, *Behind*, *Close*, and *Distant*. Note, when generating a RE, we are not considering the intrinsic reference frames of objects (e.g. the front of an object such as the screen of a monitor), but rather their extrinsic relations to other objects when viewed from a certain point of view. We leave the problem of different reference frames ((Tenbrink 2011)) for future consideration. Here are a subset of QSRs that hold for the target (Cup1) in Figure 5 (scene 16):

Close(Keyboard, Cup1) ∧ *Right(Keyboard, Cup1)* ∧
Right(Monitor, Cup1) ∧ *Behind(Monitor, Cup1)* ∧
Close(Book2, Cup1) ∧ *InFront(Book2, Cup1)* ∧
... ∧ *Behind(Lamp, Cup1).*

This approach captures abstractions used by humans, as it generalizes across quantitatively different yet qualitatively similar scenes. As such, it applies to any indoor environment in which entities are arranged on a 2D plane, without needing a massive corpora to attempt to learn those abstractions. In previous work, we exploited such abstractions for the interpretation of desktop scenes (Kunze et al. 2014).

[3] Note, in this work we generate all QSRs for all pairs of objects in the scene. However, in general, the generation of QSRs could be subject to constraints such as the distance between two objects.

Table 1: NL sentence templates for rendering REs.

RE Cat.	NL Sentence Template
Type	*Pick up the* ***?obj****.*
Relative	*Pick up the* ***?obj*** *that is (close to/far from/left of/right of/in front of/behind of) [and (close to/far from/left of/right of/in front of/behind of)] to the* ***?landmark*** *[that is . . .].*
Set-Relative	*Pick up the* ***?obj*** *in the middle.* *Pick up the* ***?nth ?obj*** *from the left/right.*
Proximal	*Pick up the* ***?obj*** *that is next to the* ***?landmark****.*
Distal	*Pick up the (leftmost—rightmost)* ***?obj****.* *Pick up the* ***?obj*** *(furthest from/closest to) you.*

Variables *?obj* and *?landmark* refer to the type of the respective objects; *?nth* is a positive number and denotes a particular object when counting from a given direction.

4.2 Class-Specific RE Generation

In this section, we present an algorithm that generates REs for each category using QSRs or metric positions of objects within a scene. All REs are first generated as logical formulae then rendered in NL using sentence templates shown in Table 1. We will refer to all objects of the same type as the target (T) as *distractors* (D) and all other objects as *landmarks* (L).

Algorithm 1 (*Type*) This strategy refers to the target object by its type alone. This is useful when there are no distractors and when the other algorithms cannot generate a RE (cf. Section 5). Example: *"Pick up the bottle"*.

Algorithm 1: Adapted locative incremental algorithm for the generation of *relative* REs.

```
1  Function REGRelative ($T, D, L, Q$)
   Input  : Target object $T$; set of distractors $D$; set of landmarks
            $L$; qualitative scene description $Q$
   Output : Referring expression $RE$
2  begin
3      $RE \leftarrow NIL$
4      /* Rank $L$ by size, distance to $T$, No. of distr.*/
5      $LR \leftarrow Ranked(L)$
6      for $l \in LR$ do
7          $R \leftarrow \{r \mid r(T, l) \in Q\}$
8          /* Order R according to predefined order: $Close$,
             $Distant, InFront, Behind, Left, Right$*/
9          $RO \leftarrow Ordered(R)$
10         for $r_{set} \in PowerSet(RO)$ do
11             if $r_{set}$ is distinguishing for $T$ w.r.t. $D$ then
12                 /* Craft RE using relations between $T$ and $l$*/
13                 $RE_T \leftarrow REG(T, l, r_{set})$
14                 $LD \leftarrow GetDistractors(l, L)$
15                 if $r_{set}$ is distinguishing for $l$ w.r.t. $LD$ then
16                     return $RE_T$
17                 else
18                     $T' \leftarrow l$
19                     $D' \leftarrow LD$
20                     $L' \leftarrow \{l | l \in L \setminus LD\}$
21                     $RE_l \leftarrow REGRelative(T', D', L', Q)$
22                     if $RE_l \neq NIL$ then
23                         return $RE_T + RE_l$
24     return $RE$
```

Algorithm 2 (*Relative*) This strategy is an adaptation of the algorithm presented in (Kelleher and Kruijff 2005; 2006). If there are no distractors in a scene, the type of the target object is discriminative and is thus used to describe the object; otherwise, Algorithm 1 is called, with the target, distractors, landmarks, and set of QSRs that hold in the scene given as input (Line 1).

First, landmarks are ranked according 'suitability' (Lines 4-5), such that large landmarks close to the target that have fewer distractors are preferred. For each candidate landmark l (Line 6), the algorithm determines the set of QSRs that hold between it and the target T (Line 7), and orders them: (*Close*/*Distant* < *InFront*/*Behind* < *Left*/*Right*) (Lines 8-9) with the aim of minimizing the interlocutor's cognitive effort (Kelleher and Kruijff 2005; 2006). The algorithm then iterates over all elements of the power set of QSRs (Line 10). If an element r_{set} discriminates a target from its distractors (Line 11), then, like (Kelleher and Kruijff 2005; 2006), the algorithm generates a RE (Lines 12-13). Unlike that work, however, we then additionally verify that r_{set} also discriminates the candidate landmark (Line 15). If so, the RE is returned (Line 16). Otherwise, if the RE discriminates the target but not the landmark, the algorithm recurses with the landmark as the new target (Lines 18-21). If this recursive call is successful, the chain of REs is combined and returned (Lines 22-23). Example: *"Pick up the bottle that is close to and in front of the keyboard"*.

Algorithm 3 (*Set-Relative*) This class has two strategies. The first is used if the target object is *between two distractor objects*. For all distractors we check whether any two are *Left* and *Right*, *InFront* and *Behind*, *Left/InFront* and *Right/Behind*, or *Left/Behind* and *Right/InFront* with respect to the target:

$$\begin{aligned}&Middle\ (T, x, y) \Leftrightarrow \exists x, y\\ &\quad((Left(x, T) \wedge Right(y, T))\vee\\ &\quad(InFront(x, T) \wedge Behind(y, T))\vee\\ &\quad(Left(x, T) \wedge InFront(x, T)\wedge Right(x, T) \wedge Behind(y, T))\vee\\ &\quad(Left(x, T) \wedge Behind(x, T) \wedge Right(x, T) \wedge InFront(y, T)))\end{aligned}$$

where $x, y \in D$. If any of these hold for the target (but not for any distractor) we use them to generate a RE. Example: *"Pick up the bottle in the middle"*.

When this first strategy fails, we instead compute an ordering in two directions (left to right and front to back) over the target object and all distractor objects using metric object positions. The position of the target object within the ordered list is then computed in all four directions, and the direction with the lowest position is chosen. Example: *"Pick up the second bottle from the right"*.

Algorithm 4 (*Proximal*) This strategy uses the same algorithm as *Relative*, but only generates an RE if the *Close* relation is distinctive. Example: *"Pick up the cup that is next to the keyboard"*.

Algorithm 5 (*Distal*) This strategy generates an RE if the target is the furthest of its type in a certain direction:

Leftmost(T,D) $\Leftrightarrow \neg\exists x Left(x, T)$,
Rightmost(T,D) $\Leftrightarrow \neg\exists x Right(x, T)$,
Closest(T,D) $\Leftrightarrow \neg\exists x InFront(x, T)$,
Furthest(T,D) $\Leftrightarrow \neg\exists x Behind(x, T)$,

where $x \in D$. For this strategy to generate an RE, the target must be above a distance threshold (15cm in this work, but in principle learnable from data) from the next distractor if those objects are qualitatively in the same position. Example: *"Pick up the bottle furthest from you"*.

5 RE Class Selection Strategies

The algorithms described in Section 4.2 generate one RE for each category. In this section we define two methods for choosing which class of RE a robot should use when interacting with a human. First, we present a classifier-based strategy that learns which class of RE would be used by humans in a particular scene. Second, we present a fixed-ordering strategy, which uses the fact that only some classes of RE can be generated in a given scene.

Classifier For the first selection strategy, we trained a set of classifiers on annotated human data. Based on object type information and human-used QSRs (Section 4.1) we identified twenty-six relevant features (in future work we aim to learn such features using unsupervised methods):

- Two scene-wide parameters: the number of distractors and non-distractors in the scene.
- Six distance parameters: the number of distractors and non-distractors within a certain distance from the target object: *Close* (R1), *Medium* (R2), and *Distant* (R3). We parameterized the qualitative relations with R1 and R2 for which we tried several combinations of values (in meters): .3, .35, .4, .45 for R1, and .5, .6, .7 for R2, and where R3 was set to 5.0; a number sufficiently high so as to extend to the edge of the scene. However, such parameterizations of QSRs can be learned as we have shown in previous work (Young and Hawes 2015).
- Twelve directional parameters: the number of distractors and non-distractors, within a distance *Close (R1), Medium (R2), Distant (R3)* of the target in each direction (i.e., *InFront*, *Behind*, *Left*, or *Right*).
- Eight binary existential parameters: each of which is 1 iff there is a distractor within either 15cm or 5m of a particular table edge (top, bottom, left, right).

Several classifiers were trained on a corpus of human-generated REs (Section 6) annotated with RE category information. Specifically, we examined Naive Bayes, Logistic Regression (LR), Decision Trees, and Linear SVM classifiers, each under the range of parameterizations listed above. Ten-fold cross evaluation for each classifier-parameterization pair showed best performance results with LR under parameterization R1=.45,R2=.7 (46.73% accuracy).

Dependent on the scene, it might be the case that not all of the five algorithms will be able to produce a RE. Therefore, we have determined an ordering of classifiers/algorithms in which we use them when available. We used the confusion matrix resulting from training the LR classifier on the full data set to create a set of classification orderings. For example, when a scene is classified as *Relative*, the next-best classifications are, in decreasing order, *Proximal*, *Set-Relative*, *Distal*, and *Type*. Here, the "next-best classifications" are those classes that are most often misclassified as the target class; because *Proximal* is the second most likely actual class of REs classified as *Relative*, we view it as the second best choice of class for a RE classified as *Relative*; the fact that a RE of class *Relative* is not available suggests that perhaps *Relative* was not actually the correct choice, so *Proximal* represents a "second chance" for the approach. If this is not possible, a RE of class *Set-Relative* is used, and so on.

Fixed Ordering The second RE class selection strategy used an experimentally-determined fixed ordering: instead of using a selection ordering based on the LR classifier, this strategy always uses the ordering *Distal*, *Proximal*, *Relative*, *Set-Relative*, *Type*, moving roughly in descending order of typical discriminability, at least for the scenes we examined. While this ordering does not *explicitly* use environmental features, it does *implicitly*, as those environmental features affect which of the five algorithms will be able to produce results in the first place, and thus guide the RE class selection mechanism through failure of algorithm preconditions.

6 Crowdsourced Evaluation

In this section we describe the crowdsourced evaluations used to evaluate the presented REG algorithms[4]. The two criteria we have evaluated will show how successfully our approach could ultimately be when used by a robot. We wished to evaluate: (C1) how well our machine-generated REs *subjectively* compared to those generated by humans, as assessed through *preference ordering*; and (C2) how well our machine-generated REs *objectively* compared to those generated by humans, as assessed by *task completion*.

6.1 Experiment One: Subjective Analysis

To evaluate C1, we devised a ranking task in which recruited participants (18 male, 11 female, ages 23 to 62 (M=31.28,SD=9.45)) were shown images generated in the initial data collection (Section 3.1), and asked to rank from "best" to "worst" eight randomly ordered commands for picking up the target in each image: six sampled human-generated commands and two machine-generated commands (generated using the Classifier- and Fixed-Ordering-based strategies).

Data was successfully collected for 19 of the 20 scenes. To analyze this data, we performed a repeated-measures ANOVA

[4] All participants were recruited using AMT and paid a small cash incentive.

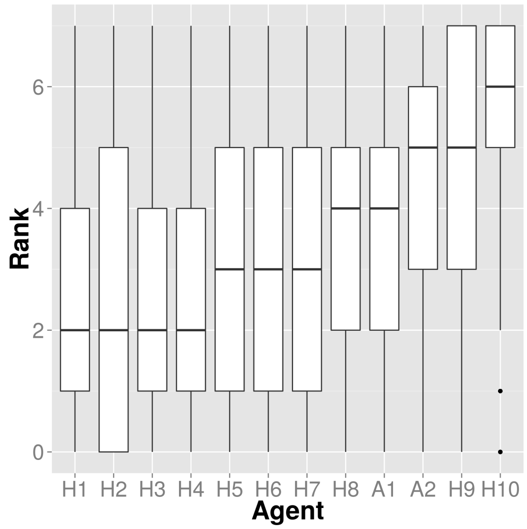


Figure 4: Subjective performance of REG algorithms (using different class selection strategies) vs. human subjects across all scenes. H1-10 represent human-generated REs; A1 represents the machine-generated REs using the fixed-ordering strategy; A2 represents the machine-generated REs using the classifier-based ordering strategy. Barred boxes denote first, second, and third quartile scores.

with participants' rankings as the dependent variable, and the RE generating agent (i.e., the human, or the algorithm using a specific class selection strategy) and presented scene as independent variables (Figure 4). Significant differences were found for agent ($F(11, 3952) = 35.059, p < .0001$) and for interaction between scene and agent ($F(198, 3952) = 2.660, p < .0001$)). The fixed-ordering strategy (A1 in Figure 4, M=3.54) performed better than the classifier based strategy (A2 in Figure 4, M=4.36), but both were in the bottom half overall.

This performance may be partly due to our simple sentence templates (Table 1). In this work, we have mainly focused on the content selection, i.e. the generation of the logical formulas, but not the linguistic realization, and as such our simple sentence templates (Table 1) likely harmed performance. It would be interesting to evaluate whether the results could be improved by using a different rendering approach. One reason why the fixed ordering may have performed better than the classifier is that the data provided by the classifier was only tagged with the most prominent category, for the sake of simplicity. However, human-generated REs were sometimes complex, making use of a series of REs of different classes, and sometimes narrowed the focus of the scene to a particular area as part of their RE (e.g., by calling attention to a particular group of objects or a particular part of the table).

6.2 Experiment Two: Objective Analysis

In the second experiment, we objectively evaluate the completion of tasks (C2) which is very essential for any successful interaction between humans and robots. To evaluate C2, we devised a task in which participants (15 male, 7 female, ages 23 to 62 (M=30.95, SD=8.99)) were shown the images generated in the initial data collection *without* the added arrow, paired with machine- or human-generated commands. For each scene-object pair, participants were asked to click on the object specified by the command. Overall, we presented to participants the twenty images four times each, in randomized 20-image blocks, totalling 80 images each. Each presented image was captioned with either the machine-generated RE (using the fixed-ordering strategy) or a RE from one of the three humans with the highest median subjective ratings. We only used REs of three humans to prevent participants from viewing scene-object pairs too often.

Table 2: Objective performance comparison.

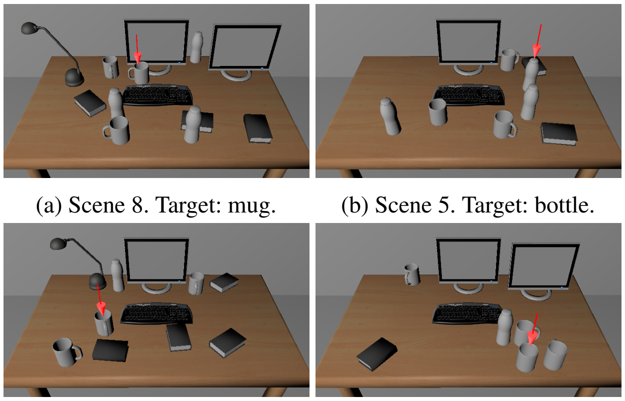
	Algorithm	**H1**	**H2**	**H3**
Success Rate	0.81	0.74	0.73	0.71

Results We performed a logistic regression analysis, with scene and RE-generating agent (i.e., the three humans, and the algorithm) as independent variables, and success of identification as the dependent variable. This revealed significant effects indicating that some scenes were more difficult to create REs for than others, and that some agents were more effective than others. As seen in Table 2, the proposed REG algorithms (using the fixed-ordering strategy) produced the objectively best results (81% vs. 74%, 73%, and 71%). Furthermore, a Pearson's chi-squared test of independence was performed to examine the difference in performance between human- and machine-generated REs; the results show a significant difference, X^2(1,N =1760)=12.4266,$p < .0005$. To determine why the presented algorithms were more effective than the top subjectively-performing humans, we examined some scenes in which the machine-generated REs greatly outperformed the human-generated REs.

Discussion Human REs generally showed a lack of attention. In the following we discuss four particular scenes of our data set (Figure 5). For example, in Scene 8, the presented algorithms generated "Pick up the cup that is next to the keyboard", whereas one user stated "Pick up the cup behind the left side of the keyboard and the monitor", which does not actually describe any object. Participants given this RE misidentified the target as the mug *next to* the keyboard. The human may simply have forgotten to add the words "next to" before "the monitor", in which case their utterance would have been unambiguous.

Similarly, there were scenes in which the human RE clearly referred to the wrong object. For example, in Scene 5, the target object was a bottle, but one human RE from the initial data collection referred to the book behind the bottle, which one might think the arrow was pointing to if they did not look closely. In this scene, participants in the second experiment may have actually clicked on the object that was described to them, it simply wasn't the object intended in the experiment.

In Scene 17, the proposed ensemble of algorithms gener-



(a) Scene 8. Target: mug.

(b) Scene 5. Target: bottle.

(c) Scene 16. Target: mug.

(d) Scene 17. Target: mug.

Figure 5: Example scenes of our data set. The corpus of images, metadata, and REs is available at `https://github.com/williamstome/SPARE-Corpus`.

ated a RE which was not perfect ("Pick up the cup that is in front of the bottle") but performed much better than the human REs. One human referred to the wrong cup, while two humans used side-to-side descriptions instead of front-to-back descriptions (i.e., "Pick up the coffee mug to the right of the bottle" and "Pick up the cup next to the bottle") which were either incorrect or ambiguous.

In Scene 16, two of the three humans also generated REs which referred to the wrong object. This seems to be due to inattention, again, as those humans seem not to have noticed a potential distractor object which made their REs inaccurate. The remaining human seems to have confused left and right, another sign of inattention. We thus see that the majority of the cases where the algorithm was *significantly* better were due to lack of attention on the part of human participants. However, this is only a problem when comparing human REs to machine-generated REs; not when comparing REG algorithms to each other.

7 General Discussion

The goal of this work was to develop REG algorithms for HRI based on empirical data, and to evaluate these algorithms through empirical experimentation, thus closing the loop on scientific discovery. In this work we sought to evaluate our algorithms using human data in part because in Human-Robot Interaction scenarios, humans will expect their teammates to communicate in a natural way. In this section, we will briefly discuss the four major contributions of this work.

Image and RE Corpus First, to study human generation of small-scale spatial REs, we generated a set of images of tabletop environments, and collected a corpus of human-generated REs referencing items found in those scenes. This corpus of images, metadata, and REs is available at `https://github.com/williamstome/SPARE-Corpus` for other researchers to use to both study human generation of REs and to test REG algorithms. Although smaller in size then other RE datasets (Kazemzadeh et al. 2014b; Mao et al. 2016), our approach allows us to generate arbitrary scenes for a range of different, situated tasks. Moreover, ground truth information (in 3D) can be immediately obtained from the simulator without the need for expensive labeling.

Categorization of Small-Scale Spatial Referring Expressions Second, examination of this corpus suggested five categories of small-scale spatial REs. Researchers will be able to refine this categorization, and possibly use it to inform their own REG algorithms.

REG Algorithms Third, we presented an ensemble of REG algorithms, and strategies for algorithm selection. In the future, it will be valuable to evaluate this ensemble relative to other REG algorithms. It is important to note that while our approach focused on spatial REG, there is no reason why other features could not be included in future work.

Evaluation Framework Finally, beyond the validation of the algorithms, the presented evaluation framework provides a benchmarking platform for comparing REG algorithms. This framework could easily use scenes and images of varying degrees of complexity beyond those presented here. The framework consisted of three phases, each of which presented participants with information gathered by other participants in the previous phase. This evaluation allowed us to both subjectively and objectively evaluate algorithms relative to each other and relative to human participants.

8 Conclusion and Future Work

In this paper, we have presented work with four primary contributions: (1) a corpus of machine-generated images and human-generated REs, (2) a categorization of small-scale spatial REs, (3) an ensemble of REG algorithms for HRI, and (4) a framework for the evaluation of REG algorithms.

The foremost direction for future work will be to integrate the presented algorithms with the Dialogue and Perception components of a robot architecture (e.g. DIARC (Scheutz et al. 2013)), in order to generate referring expressions in task-based scenarios based on perceptual data. In particular, we foresee integrating the presented algorithms with QSR-based robot perception systems such as that seen in (Kunze et al. 2014). Figure 1 shows a first step in this direction where a RE was generated based on QSRs that were abstracted from RGBD sensor data.

Acknowledgments

This work was funded in part by the European Union Seventh Framework Programme, (FP7/2007-2013) grant 600623, STRANDS, by EPSRC grant (EP/M015777/1), ALOOF, and ONR grant N00014-14-1-0149.

References

Belz, A. 2008. Intrinsic vs. extrinsic evaluation measures for referring expression generation. In *In Proc. 46th Annual Meeting of the Association for Computational Linguistics*.

Dale, R., and Reiter, E. 1995. Computational interpretations of the gricean maxims in the generation of referring expressions. *Cog.Sci.*

Fang, R.; Liu, C.; She, L.; and Chai, J. Y. 2013. Towards situated dialogue: Revisiting referring expression generation. In *Conference on Empirical Methods in Natural Language Processing*. ACL.

Fisher, M.; Ritchie, D.; Savva, M.; Funkhouser, T.; and Hanrahan, P. 2012. Example-based synthesis of 3d object arrangements. *ACM Transactions on Graphics (TOG)* 31(6):135.

Gatt, A.; Belz, A.; and Kow, E. 2009. The tuna-reg challenge 2009: Overview and evaluation results. In *Proceedings of the 12th European Workshop on Natural Language Generation*, ENLG.

Gupta, R., and Kochenderfer, M. J. 2004. Common sense data acquisition for indoor mobile robots. In *Proceedings of the 19th national conference on Artifical intelligence*, 605–610.

Hawes, N.; Burbridge, C.; Jovan, F.; Kunze, L.; Lacerda, B.; Mudrová, L.; Young, J.; Wyatt, J. L.; Hebesberger, D.; Körtner, T.; Ambrus, R.; Bore, N.; Folkesson, J.; Jensfelt, P.; Beyer, L.; Hermans, A.; Leibe, B.; Aldoma, A.; Faulhammer, T.; Zillich, M.; Vincze, M.; Al-Omari, M.; Chinellato, E.; Duckworth, P.; Gatsoulis, Y.; Hogg, D. C.; Cohn, A. G.; Dondrup, C.; Fentanes, J. P.; Krajník, T.; Santos, J. M.; Duckett, T.; and Hanheide, M. 2016. The STRANDS project: Long-term autonomy in everyday environments. *IEEE Robotics Automation Magazine*. Accepted Conditionally.

Kazemzadeh, S.; Ordonez, V.; Matten, M.; and Berg, T. L. 2014a. ReferItGame: Referring to objects in photographs of natural scenes. In *Conference on Empirical Methods in Natural Language Processing*.

Kazemzadeh, S.; Ordonez, V.; Matten, M.; and Berg, T. L. 2014b. Referitgame: Referring to objects in photographs of natural scenes. In *EMNLP*.

Kelleher, J., and Kruijff, G.-J. 2005. A Context-dependent Algorithm for Generating Locative Expressions in Physically Situated Environments. In *European Workshop on Natural Language Generation*.

Kelleher, J. D., and Kruijff, G.-J. M. 2006. Incremental generation of spatial referring expressions in situated dialog. In *21st International Conference on Computational Linguistics*, 1041–1048.

Koller, A.; Striegnitz, K.; Gargett, A.; Byron, D.; Cassell, J.; Dale, R.; Moore, J.; and Oberlander, J. 2010. Report on the second NLG challenge on generating instructions in virtual environments (GIVE-2). In *6th international natural language generation conference*.

Krahmer, E., and van Deemter, K. 2012. Computational generation of referring expressions: A survey. *Comput. Linguist.* 38(1):173–218.

Krahmer, E.; van Erk, S.; and Verleg, A. 2003. Graph-based generation of referring expressions. *Comput. Linguist.* 29(1):53–72.

Kunze, L.; Burbridge, C.; Alberti, M.; Tippur, A.; Folkesson, J.; Jensfelt, P.; and Hawes, N. 2014. Combining top-down spatial reasoning and bottom-up object class recognition for scene understanding. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*.

Kunze, L.; Burbridge, C.; and Hawes, N. 2014. Bootstrapping probabilistic models of qualitative spatial relations for active visual object search. In *AAAI Spring Symposium 2014 on Qualitative Representations for Robots*.

Lemaignan, S.; Hanheide, M.; Karg, M.; Khambhaita, H.; Kunze, L.; Lier, F.; Lütkebohle, I.; and Milliez, G. 2014. Simulation and HRI recent perspectives with the MORSE simulator. In *Int'l Conf. on Simulation, Modeling, and Programming for Autonomous Robots*.

Mao, J.; Huang, J.; Toshev, A.; Camburu, O.; Yuille, A.; and Murphy, K. 2016. Generation and comprehension of unambiguous object descriptions. In *CVPR*.

Merrell, P.; Schkufza, E.; Li, Z.; Agrawala, M.; and Koltun, V. 2011. Interactive furniture layout using interior design guidelines. *ACM Transactions on Graphics (TOG)* 30(4):87.

Moratz, R.; Nebel, B.; and Freksa, C. 2003. Qualitative spatial reasoning about relative position. *Spatial cognition III* 1034–1034.

Russell, B. C.; Torralba, A.; Murphy, K. P.; and Freeman, W. T. 2008. Labelme: A database and web-based tool for image annotation. *International Journal of Computer Vision* 77(1-3):157–173.

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 *AAAI Workshop on Intelligent Robotic Systems*.

Tellex, S.; Kollar, T.; Dickerson, S.; Walter, M. R.; Banerjee, A. G.; Teller, S.; and Roy, N. 2011. Understanding natural language commands for robotic navigation and mobile manipulation. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*.

Tenbrink, T. 2011. Reference frames of space and time in language. *Journal of Pragmatics* 43(3):704–722.

Viethen, J., and Dale, R. 2006. Towards the evaluation of referring expression generation. In *4th Australasian Language Technology Workshop*.

Winograd, T. 1972. *Understanding Natural Language*.

Young, J., and Hawes, N. 2015. Learning by observation using qualitative spatial relations. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, AAMAS '15, 745–751. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems.