Grounding Language in Multi-Perspective Referential Communication

Anonymous ACL submission

Abstract

We introduce a task and dataset for referring expression generation and comprehension in multi-agent embodied environments. In this task, two agents in a shared scene must take into account one another's visual perspective, which may be different from their own, to both produce and understand references to objects in a scene and the spatial relations between them. We collect a dataset of 2,970 human-written referring expressions, each paired with human comprehension judgments, and evaluate the performance of automated models as speakers and listeners paired with human partners, finding that model performance in both reference generation and comprehension lags behind that of pairs of human agents. Finally, we experiment training an open-weight speaker model with evidence of communicative success when paired with a listener, resulting in an improvement from 59.7 to 69.2% in communicative success and even outperforming the strongest proprietary model.

1 Introduction

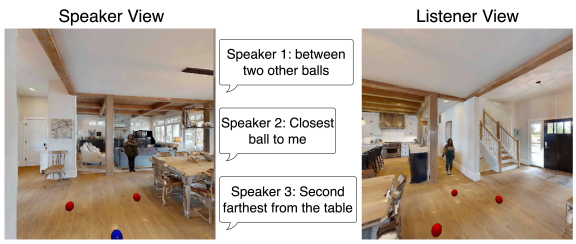

Figure 1: Example scene from our environment and dataset. On the left, the speaker refers to the target object, distinguished by its blue color. On the right, the listener selects the candidate referent they believe is described by the speaker's description, without access to its distinct color.

Language agents embodied in situated interactions alongside human users must be able to reason jointly about the space they occupy, the language they encounter, and their human partner's perception. For example, in Figure 1, one agent describes the location of an object to another agent, whose view differs from their own. To correctly resolve and generate references to the surrounding environment, both the speaker and listener must take into account the physical relationship between objects, its own view of the environment, and an estimate of the user's perspective in the environment. In contrast to most prior work on referring expression generation and comprehension, we focus on the setting where both agents are physically embodied in a scene with different perspectives of the scene. Prior work in this setting has focused on human dyads that are literally physically situated in an environment (Schober, 1993; Taylor and Tversky, 1996), or in synthetically-generated abstract environments (Udagawa and Aizawa, 2019).We study human-human and human-agent referential communication in photorealistic 3D environments, introducing a platform that supports generating task instances with varying levels of difficulty.

We collect a dataset of 2,970 human-written referring expressions grounded in 1,485 generated scenes. We evaluate several recent vision-and-language models on the tasks of referring expression generation and comprehension, including general instruction-tuned vision-language models, models designed for fine-grained vision-language processing, and a modular vision-and-language reasoning system. When interpreting human-written referring expressions, the fine-grained Ferret model (You et al., 2023) performs the best, successfully identifying 69.2% of intended referents. Using human listeners, we find that the proprietary GPT-4o produces referring expressions that correctly identify the intended target referent for 64.9% of scenes, while the open-weight LLaVA-1.5 (Liu et al., 2024) is only successful for 55.7% of scenes. Compared to the average human-human success rate of 87.6%, all models lag far behind humans when both generating and comprehending referring expressions. Analyzing the language used by both automated and human speakers reveals significant differences in referential strategies; for example, human speakers use themselves or the listener agent as reference points much more frequently than automated models, which mostly rely on other objects in the scene.

Our scene-generation platform supports controlling two levels of task difficulty. First, it supports modifying the relative orientation of the agents. Second, we train a referent placement policy to minimize communicative success between two automated agents. For scenes generated using this policy, we see a significant decrease in communicative success across nearly all agent combinations.

Finally, we experiment with improving our weaker speaker model, LLaVA-1.5, by fine-tuning it with data collected in deployment with both human and automated listeners. During learning, we first sample referring expressions from the speaker model, convert empirical observations of language interpretation by a listener into training examples (Kojima et al., 2021), then apply proximal policy optimisation to update model parameters on this data. With a single round of training, we see significant improvements in LLaVA-1.5's ability to generate accurate referring expressions, with rates of communicative success with a human listener improving from 59.7 to up to 69.2, outperforming even the stronger GPT-4o speaker. Our code, models, and dataset will be released under an open-source license upon publication.

2 Task and Environment

We study the task of embodied referential communication, where two agents coordinate their attention in a shared scene using referring expressions. To this end, we design a platform that for generating photorealistic 3D scenes that support this task at varying levels of difficulty.

2.1 Embodied Referential Communication

We study referential communication via a reference game (Clark and Wilkes-Gibbs, 1986), where a speaker describes a target referent, and a listener attempts to identify the target using the speaker's description. In our task, two agents are physically embodied in the same shared 3D scene, but with different perspectives, and thus different observations of the scene. Each scene includes candidate referent objects, one of which is a target object that the speaker needs to communicate to the listener. Communicative success is achieved if the listener is able to identify the speaker's intended target.

Formally, let $\mathcal{O}$ be the set of possible agent observations, each represented as a 2D image; $\mathcal{R}$ be the set of candidate referents in an scene, and $\mathcal{X}$ be the set of possible referring expressions. Formally, a speaker model $p_s : \mathcal{O} \times \mathcal{R}^N \times \{1 \dots N\} \rightarrow \Delta^{\mathcal{X}}$ maps from an observation of the shared scene, a set of referents, and the index of the target referent r_t to a distribution over possible referring expressions. A listener model $p_l : \mathcal{O} \times \mathcal{R}^N \times \mathcal{X} \rightarrow \Delta^{\{1 \dots N\}}$ maps from its observation of the scene, the set of all candidate referents, and the referring expression generated by the speaker to a distribution over possible referent indices. Given a scene with speaker observation $o_s \in \mathcal{O}$, listener observation $o_l \in \mathcal{O}$, a set of N candidate referents $\mathcal{R}$, and a target referent index t, communicative success is achieved when the listener selects the intended target:

$$x = \arg\max_{x' \in \mathcal{X}} p_s(x' \mid o_s, \mathcal{R}, t)$$

$$\hat{t} = \arg\max_{1 \leq i \leq N} p_l(i \mid o_l, \mathcal{R}, x)$$

$$\text{Success}(p_s, p_l, o_s, o_l, \mathcal{R}, t) = \mathbb{1}_{t=\hat{t}}$$

2.2 Scene Generation

Formally, we denote a scene $\mathcal{S} = (e, \rho_s, \rho_l, \mathcal{R}, t)$ as an environment $e \in \mathcal{E}$ populated with two agents ρ_s and ρ_l and N referents $\mathcal{R}$, as well as the index of the target referent r_t. To generate a scene, we first sample a base environment, then place the two agents, then the candidate referents. Finally, we render each agent's observation of the scene.[1]

Base environments. We load indoor 3D environments from ScanNet++ (Yeshwanth et al., 2023) as 3D meshes into habitat-sim (Savva et al., 2019), which supports basic object physics and ray casting for identifying fields of view visible to each agent.

Agent placement. Both the speaker and listener agents are associated with a camera pose $\rho = (\langle x, y, z \rangle, \langle \theta, \phi, \psi \rangle)$, where $\langle x, y, z \rangle$ denote the position in 3D space and $\langle \theta, \phi, \psi \rangle$ represent the pitch, roll, and yaw angles respectively. To ensure observations are reasonable, we sample the camera height z from a range of typical adult human height, and fix pitch θ and roll ϕ at 0°. We enforce a maximum distance between the agent cameras, and a

[1] Appendix A.1 contains additional details about scene generation, including object placement and observation rendering.

non-empty overlap of their respective fields of view. We randomly assign speaker and listener roles to the two cameras, except in the case that only one agent's camera is in the other's field of view, but not vice versa. In this case, the former camera represents the speaker.

Candidate referent placement. Each scene contains a set of $N = 3$ candidate referents $\mathcal{R} = \{r_1, \ldots, r_N\}$, where $r_i = \langle x_i, y_i, z_i \rangle$ denotes the location of each referent. A target index $1 \leq t \leq N$ denotes the referent that the speaker aims to communicate to the listener. For each referent, we first sample a position from the set of all empty coordinates $\mathcal{C}$ in the scene. We use a gravitational physics simulation to drop the each referent from this position until it comes to rest on a solid horizontal surface. We use rejection sampling to ensure all referents are visible to both agents, and referents are not too close together.

Agent observations. Each agent's observation is represented as a 2D image $o \in \mathbb{R}^{3\times H\times W}$ rendered from its camera pose ρ. The speaker's observation $o_s = \text{proj}_s(e, \mathcal{R}, t, \rho_s)$ is a projection of the speaker's view of the environment, and $o_l = \text{proj}_l(e, \mathcal{R}, \rho_l)$ is a projection of the listener's view. While proj_l renders each referent with the same color (red), proj_s renders the target r_t in a different color (blue) from the distractor objects, allowing the speaker to easily distinguish the target when writing their referring expression. Both projections also render the other agent's camera as a 3D model of a human, which are sampled from 2K2K (Han et al., 2023).

2.3 Controlled Difficulty

We implement two ways to control the difficulty of referential communication via scene generation: by manipulating the relative orientation of speaker and listener, and by adversarially placing referents. Figure 2 shows examples of four scenes generated from different relative orientations, and with and without adversarial referent placement.

Speaker-listener orientation. The relative orientation of the speaker ρ_s and listener ρ_l is the absolute difference $\psi' = \min(|\psi_s - \psi_l|, 360° - |\psi_s - \psi_l|)$ of their horizontal rotations (yaw). We experiment with the influence of ψ' on interaction dynamics. When ψ' is close to $0°$, the two agents are facing the same direction, and their observations are likely to be similar to one another. When ψ' is close to $180°$, the agents are facing each other and thus have completely different views of the same scene. Following Schober (1993), we hypothesize that differences in relative angles of speakers and listeners may influence language use. Our environment supports uniformly sampling agent placements with fixed relative orientation.

Adversarial placement of referents. We design a referent placement policy model $R : \mathcal{C}^* \times \mathcal{O}_s \times \mathrm{P}_s \times \mathrm{P}_l \rightarrow \Delta^{\mathcal{R}^N \times \{1\ldots N\}}$, which takes as input a set of empty coordinates $\mathcal{C}$, the speaker's observation prior to referent placement, and both agent poses. It generates a distribution over referent locations prior to the physics simulation, and over referent indices representing the target. The policy model is implemented as a vision transformer (Dosovitskiy et al., 2020), and is trained to maximize the communicative failure rate between two fixed agent models, $\hat{p}_s$ and $\hat{p}_l$, by optimizing

$$\max_R \mathbb{E}_{(\mathcal{R}',t')\sim R(\cdot)} \left[1 - \text{Success}(\hat{p}_s, \hat{p}_l, o_s, o_l, \mathcal{R}', t')\right] ,$$

where o_s and o_l are the agents' observations after referents $\mathcal{R}$ are placed. During scene generation, we use the trained policy to sample initial positions of referents, then apply gravitational physics to find the resting position of each referent.

3 Experimental Setup

We use our scene generation platform to evaluate embodied, multi-perspective referential communication with pairs of agents including humans and automated models.

3.1 Data

We generate a set of 27,504 scenes for training and evaluating automated agents. We recruit crowdworkers to participate in the task both as listeners and speakers, collecting a dataset of 2,970 human-written referring expressions paired with human listener selections in 1,485 of these scenes.

Scene generation. We use ScanNet++ (Yeshwanth et al., 2023) (non-commercial license), which contains 450 high-quality 3D indoor environments, as the basis of our task instances. We generate scenes using both forms of controlled difficulty (Section 2.3). First, we train our adversarial referent placement policy, implemented as ViT-s/16 (Dosovitskiy et al., 2020), using GPT-4o as both a speaker and listener in 27,600 generated

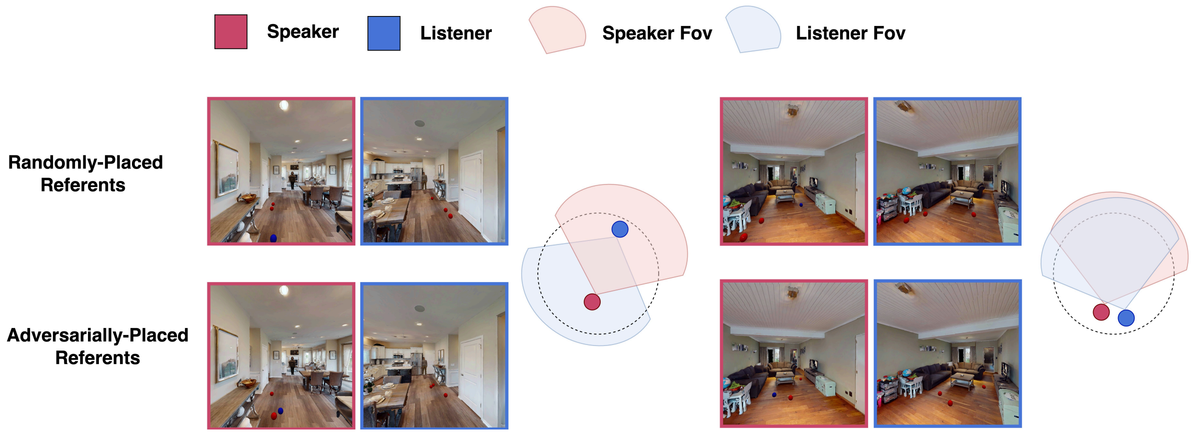

Figure 2: Example scenes generated with different relative orientations ($\approx 180°$ on left, $\approx 30°$ on right) and with randomly- (top) or adversarially- (bottom) placed referents. Adversarially-generated referent configurations often space referents more evenly, with the target referent not easily uniquely identifiable.

scenes comprising 60 samples per base environment.[2] To generate our final dataset of scenes, we first sample 300 agent placements for each relative angle in $\{0, \ldots, 180\}$ distributed uniformly across the 450 base environments. For each of these agent placements, we sample two referent placements, resulting in two complete scenes: one where referent locations are randomly sampled, and another where referents are placed using the adversarial referent placement policy.

We use GPT-4o to perform rejection sampling on low-quality scenes, removing examples with visible artifacts and those that make the task impossible, e.g., where all referents are not visible to both agents. The final dataset includes 27,504 scenes, which we split into train (80%), validation (10%) and test (10%) splits. Base environments may appear in multiple splits.

Crowdsourcing. We recruit 194 crowdworkers on Prolific[3]. Qualified workers are fluent English speakers, reside in the United States, and pass a qualification task by writing referring expressions for 15 scenes, with successful listener selection from two or more of three other workers for at least 10 of these referring expressions. On average, we pay $18 USD per hour.[4]

Speaker task. Speakers are presented with a prompt that asks them to describe the location of the blue ball to another person who may or may not be visible to them in the scene, and who cannot distinguish the colors of the balls. Speakers first click a button that reveals their view of the scene. They write a referring expression, then submit their work. We record both the referring expression and the time taken between revealing the scene and submitting the task.

Listener task. Listeners first click a button that reveals their view of the scene and a referring expression. They click on the referent they believe to be the target in the image, then submit their work. We record both the click position and the time taken between revealing the view and submitting the task. A listener's selection is the sphere which is rendered closest to their click position.

Dataset statistics. For a randomly-sampled subset of 1,485 scenes from the validation set, we collect a referring expression from at least one worker, resulting in a total of 2,970 referring expressions, paired with judgments from three separate listeners. Each referring expression is labeled with the majority-class referent selection. The median time spent per speaker and listener task are 33.0s and 10.5s respectively.

3.2 Evaluated Models

We experiment with four instruction-tuned vision-language models.[5] Two of these models are designed for more general use: **GPT-4o**[6], a proprietary model developed by OpenAI that supports real-time joint processing of audio, vision, and text; and **LLaVA-1.5** (Liu et al., 2024), a large

[2] Appendix A.2 contains more details on the adversary.

[3] https://www.prolific.com

[4] Details on data collection, including task templates, are available in Appendix A.3.

[5] Additional details, including prompts, are available in Appendix B.1.

[6] https://openai.com/index/hello-gpt-4o/

open-weight instruction-tuned multimodal model. We also experiment with two instruction-tuned open-weight models designed specifically to refer to regions of and ground references in images at any granularity: **Ferret** (You et al., 2023) and **Groma** (Ma et al., 2024). Ferret employs a hybrid region representation that combines discrete coordinates and continuous features to represent regions in an image, while Groma utilizes a localized visual tokenization mechanism, where an image is decomposed into regions of interest and encoded into region tokens. We use these models as listeners only as preliminary experiments showed poor performance on reference generation.

We also experiment with modular vision-language reasoning systems, which decompose the problems of language understanding and perception by first mapping language to some executable code, which is then executed on an image (Subramanian et al., 2023; Gupta and Kembhavi, 2023). In this work, we use **ViperGPT** (Surís et al., 2023), using GPT-4 to generate intermediate Python programs. We use ViperGPT as a listener agent only.

For both speaker models, we provide as input the speaker's observation o_s and a prompt to describe the location of the blue sphere. For listeners, we provide as input a referring expression x and the listener's observation o_l, as well as a list of each candidate referent's bounding box, and prompt the model to select the bounding box corresponding to the described target. We sample from all models using a temperature of 0.

3.3 Evaluation and Analysis

We evaluate models both as speakers and listeners, partnered both with human and automated agents. Our main metric is communicative success: for each scene, did the pair of agents successfully coordinate on the target referent? Pairing automated listeners with human speakers evaluates a model's ability to comprehend a human-written referring expression, and pairing automated speakers with human listeners evaluates a model's ability to precisely refer to a region of the scene. Both sides of this communicative task require understanding spatial language and taking into account the other agent's perspective of the shared scene. For each setting, we analyze the influence of task difficulty on communicative success.

4 Results

We experiment with four configurations of agent dyads, combining humans and automated speakers and listeners. Table 1 includes results for the 1,485 validation scenes we use for collecting human-human data, split across scenes with random and adversarial referent placement.

Human speakers and listeners. Using the referring expressions collected in Section 3.1, we find that human-human pairs achieve an average communicative success rate of 87.6.[7]

Human speakers, automated listeners. We evaluate model performance in comprehending human-written referring expressions. For each human-written referring expression in our collected dataset, we select the most likely referent according to the model. We observe substantially lower accuracy in referent selection compared to human listeners. Ferret, which was designed for fine-grained vision-and-language processing, outperforms the other models at an average selection accuracy of 69.2, but still lags far behind human performance.

Automated speakers, human listeners. We acquire a single referring expression from each instruction-tuned model for each evaluation scene. For each referring expression, we acquire three human listener selections and compare the majority class referent to the intended target. Both GPT-4o and LLaVA-1.5 are significantly less successful in describing target referents when compared to human speakers; GPT-4o's references lead to correct human listener selection in 64.9% of scenes, while the LLaVA-1.5 speaker is successful for 55.7%.

Automated speakers and listeners. We evaluate settings where both agents are automated systems. Using the referring expressions acquired from both speaker agents, we use all five listener models to perform referent selection. In nearly all cases, performance with pairs of automated listeners is lower than dyads containing at least one human. However, both Ferret and Groma perform on par with human listeners on referring expressions generated by both GPT-4o and LLaVA-1.5, for both random and adversarial referent configurations. In fact, both models actually outperform human listeners

[7] For fair comparison to settings where only one referring expression is produced per scene, we report the macro-average over scenes. The micro-average over all referring expressions in this experiment is 88.4.

		Listeners											
		Human		**GPT-4o**		**LLaVA-1.5**		**Ferret**		**Groma**		**ViperGPT**	
		Ran.	Adv.	Ran.	Adv.	Ran.	Adv.	Ran.	Adv.	Ran.	Adv.	Ran.	Adv.
Speakers	**Human**	**90.2**	**84.9**	67.6	66.0	63.3	63.2	70.1	68.2	64.3	65.7	57.8	56.0
	GPT-4o	67.8	62.0	61.1	57.2	60.4	57.8	67.8	62.1	66.5	64.8	55.6	53.3
	LLaVA-1.5	55.2	56.1	50.9	49.8	44.7	42.2	59.1	52.8	61.9	55.4	48.9	48.7

Table 1: Rates of communicative success for all four combinations of human and automated speakers and listeners, across 1,485 scenes, split by scenes with random (Ran.) and adversarial (Adv.) referent placement. Results for human-human pairs are bolded and in **blue**; results for human speakers and automated listeners are in orange; results for human listeners and automated speakers are in green; and results for fully-automated pairs are in black.

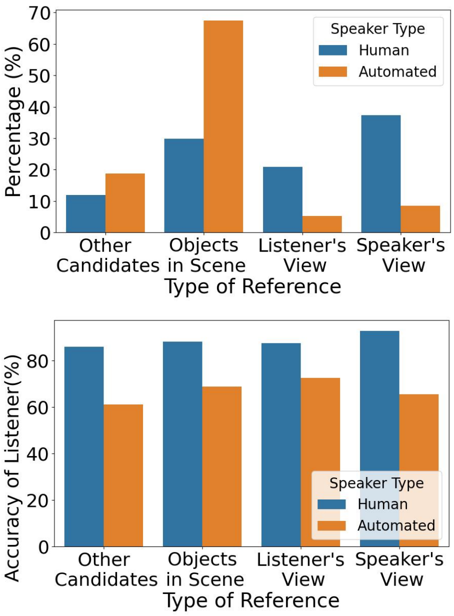

Figure 3: Distributions of speakers' referential strategies and human listeners' corresponding performance for both human and automated speakers.

for referring expressions generated by LLaVA-1.5 for random referent configurations.

4.1 Adversarial Referent Placement

Our adversarial referent placement policy was trained to minimize communicative success between a GPT-4o speaker and listener. Table 1 shows that scenes generated with this policy indeed reduce rates of communicative success in this setting by 2.4%. The learned policy also reduces the success rate for nearly all other combinations of agents, including for human-human pairs, where we see rates of communicative success drops from 91.6 to 85.1 when adversarially placing candidate referents.

4.2 Language Analysis

We manually annotate 200 randomly-sampled referring expressions written by crowdworkers and GPT-4o with respect to referential strategies used by the speaker. We consider four core referential strategies: reference to other candidate referents (e.g., *in front of the other two red balls*), reference to fixed objects in the scene (*in front of the kitchen entryway*), and reference to the listener (*on your left*) or speaker's perspective (*closest to me*). Figure 3 (left) shows the prevalence of each referential strategy for both speakers across this sample.

Both automated and human speakers typically use reference points to describe the position of the target referent. However, automated speakers rely much more heavily on reference to fixed objects, using this strategy in 67.5% of descriptions, compared to 29.5% by human speakers. In contrast, human speakers are much more likely to use themselves or the listener as reference points.

Figure 3 (right) shows the average accuracy of human listeners for references employing each referential strategy. Regardless of whether the speaker is automated or human, using other candidate referents as reference points (e.g., *in front of the other two red balls*) is most likely to mislead the listener, likely because these can introduce ambiguity in frame of reference. Conversely, using fixed objects in the scene as reference points generally performs better, but sometimes the object chosen by the speaker might not be visible to the listener, and descriptions of relative positions can change with shifts in viewing angle. This suggests estimating the listener's perspective of the scene is nontrivial, even for human speakers. While using oneself or the listener as a reference point is the most effective referential strategy, speakers sometimes fail to explicitly state whose perspective is referred to, leading to ambiguity.

5 Learning from Communicative Success

We propose to further train our speaker model from learning signals acquired during referential communication. The basic premise that motivates this approach is that empirical observations of language interpretation provides evidence of utterance meaning, regardless of speaker intent (Kojima et al., 2021). For instance, if the listener selects a different referent than the intended target, this indicates the speaker's referring expression describes (or at the very least, better describes) the chosen referent, even if the generated expression fails to describe the intended referent. In contrast to prior work that proposes methods that learn from communicative success (or failure) (Kojima et al., 2021; Liu et al., 2023), we additionally explore the use of preference-based learning signals that explicitly pair the intended and chosen targets in case of communicative failure.

Learning. During training, we collect a dataset of M examples $\mathcal{D} = \left\{(\mathcal{S}^{(i)}, x^{(i)}, \hat{t}^{(i)})\right\}_{i=1}^{M}$, each consisting of a generated scene $\mathcal{S}$ (including the target referent index t), referring expression $x \sim p_s(o_s, \mathcal{R}, t; \theta)$ sampled from a pre-trained speaker and the referent $\hat{t} \sim p_l(o_l, \mathcal{R}, x; \phi)$ selected by a listener.

We use offline proximal policy optimization (PPO; Schulman et al., 2017) to fine-tune speaker parameters θ using our collected dataset of examples $\mathcal{D}$. We experiment with two methods for using the collected data: (a) learning from successes only (LSO) and (b) pairwise preference learning (PPL). When learning from successes only, examples receive a reward of +1 when $t = \hat{t}$ and 0 otherwise. In pairwise preference learning, we take advantage of the fact that, especially in light of communicative failure, we can assume that the referring expression better describes the listener's guess than it describes the speaker's target referent. We formalize this by designing a reward function that maximizes the difference between the likelihoods of the speaker's referring expression x describing the listener's chosen target $\hat{t}$ versus the intended target t:

$$p_s(x \mid o_s, \mathcal{R}, t; \theta') - p_s(x \mid o_s, \mathcal{R}, \hat{t}; \theta') \, .$$

In cases where $t = \hat{t}$, the assigned reward is +1.

Experimental setup. We use the initial speaker model, pre-trained LLaVA-1.5 (Liu et al., 2024), to generate referring expressions for 200 scenes sampled from the training split. We experiment with learning from both human and automated listener agents. We hypothesize that human listeners will provide higher-quality feedback in the form of referent selections than the automated listener model, given a human listener's superior language-understanding capability. However, using an automated listener is less costly, as it requires collecting no additional human data. For our automated listener, we also use pre-trained LLaVA-1.5. We collect a single guess per referring expression from our automated listener, and three human listener guesses. This results in two datasets: $\mathcal{D}_a$ containing 200 examples of automated listener selections, and $\mathcal{D}_h$ containing 600 examples of human selections. Training results in four models: optimizing with learning from successes only and pairwise preference learning, and learning from $\mathcal{D}_a$ and $\mathcal{D}_h$. We acquire three human listener selections generated referring expressions in a randomly-sampled but representative subset 195 scenes from the validation set.

Speaker	Listener Accuracy	Avg. Reference Length	Vocab. Size
Pre-trained θ	59.7	61.1	410
+ LSO ($\mathcal{D}_a$)	61.5	41.7	521
+ LSO ($\mathcal{D}_h$)	65.6	54.6	462
+ PPL ($\mathcal{D}_a$)	66.7	19.8	496
+ PPL ($\mathcal{D}_h$)	69.2	15.6	547
Human	91.3	15.8	566
GPT-4o	66.3	78.9	684

Table 2: Performance of the LLaVA-1.5 speaker before and after training on data collected in 195 scenes with human listeners. We also report the average number of tokens per reference and vocabulary size for each speaker. For reference, we include statistics with human and GPT-4o speakers on the same set of scenes.

Results. Table 2 shows that learning from communicative success significantly improves the quality of an initially-weak speaker agent. Overall, learning from human listeners ($\mathcal{D}_h$) is more effective than learning from an automated listener, though this is still beneficial. We also find that preference learning significantly improves over training only on examples exhibiting correct target selection. After fine-tuning on only 200 sampled referring expressions with human judgments and preference-based reward, LLaVA-1.5 actually outperforms GPT-4o as a speaker, with a communicative success rate of 69.2 when paired with human listeners.

Manual analysis reveals that after training, the model generates fewer genuinely ambiguous descriptions (43.6 to 36.0% of analyzed descriptions), and shifts from a referential strategy that refers to other candidates to one that refers to fixed objects in the scene. We also analyze how training influences sentence length and vocabulary size on references generated for 195 scenes (Table 2): prior to training, LLaVA-1.5 produces lengthy descriptions at an average length of 61.1 tokens. After training with LSO, reference lengths decrease slightly. However, after training with PPL, reference lengths decrease significantly, matching lengths of human-written descriptions. We also find that in our setting, learning from communicative success actually increases the model's vocabulary size, in contrast to earlier work (Kojima et al., 2021).

6 Related Work

The meanings of relative spatial terms are highly dependent on the situated environment: the items participating in the relation and their intrinsic parts and affordances (Clark, 1973; Landau, 2018); the relative perspectives of participants in an embodied scene (Taylor and Tversky, 1996; Goschler et al., 2008); and within-interaction conventions formed during multi-turn embodied dialogue (Schober, 1993), among other factors. In this work, we focus on the influence of relative perspective between multiple on the use of spatial language.

Production and comprehension of referring expressions has been studied in human-human dialogue (Clark and Wilkes-Gibbs, 1986; Taylor and Tversky, 1996; van der Sluis and Luz, 2011; Udagawa et al., 2020, *inter alia*), and in interactions between human and automated language users (Janarthanam and Lemon, 2010; Fang et al., 2014, 2015; Huang et al., 2020, *inter alia*). However, most of this work has focused on disembodied referential communication, where agents tasked with communicating about sets of stimuli (Hawkins et al., 2017; Haber et al., 2019), or where agents are not physically situated within an environment (Kazemzadeh et al., 2014; Achlioptas et al., 2020). The problem of situated language grounding in multi-agent settings reflects an increasingly popular real-world scenario of embodied agents. In studies where interaction participants are both embodied with different visual perspectives on the same scene, they must either be literally physically embodied in a single scene (Schober, 1993), or are placed in synthetic environments (Udagawa and Aizawa, 2019).

A small number of existing works have trained language-generation models using evidence of communicative success in interaction with another agent. For example, Kojima et al. (2021) train an instruction-generating agent by observing humans follow generated instructions, and Liu et al. (2023) use signals from reference games with automated listeners to improve a speaker model. Our work takes inspiration from the latter to improve our speaker model using referent selections from an automated listener; however, we explore a preference-based objective that explicitly pairs the intended and empirically chosen referents.

7 Conclusion

We study multi-agent referential communication in situated interactions. In this setting, a speaker and a listener are both embodied in a shared scene, but are placed in different locations, with different views of the scene. We design a platform that supports generation of photorealistic 3D scenes, with control for difficulty of the referential task. We evaluate both humans and automated agents as speakers and listeners in this task. While human-human dyads are successful at coordinating on a referent around 88.4% of the time, automated models fall far behind when used both as speakers and as listeners. However, we can substantially improve the performance of an open-weight speaker model by training it with evidence of communicative success in referential communication with both automated and human listeners. Our findings suggest that despite the increasing relevance of multi-agent situated interactions between humans and automated agents, there is significant headroom for applying models that jointly process language and visual perception in this setting. However, they also show the promise of training such agents in interaction with people.

Limitations

Our task currently focuses on single-shot reference, where a speaker creates a single referring expression, and the listener cannot ask for clarification or engage in interactive reference resolution (Clark and Wilkes-Gibbs, 1986; Udagawa and Aizawa, 2019). Evaluating how models participate in an interactive version of our task is a compelling direction for future work. Additionally, while our experiments are currently conducted exclusively in English, the language of space and motion has enormous variation across language communities (Levinson and Wilkins, 2006). Core spatial concepts studied in English, like *on* or *in*, do not have universally uniform meanings, with different languages dividing the conceptual space of spatial language in vastly different ways (Landau, 2017). Future work should explore how spatial Finally, our experiments on learning from communicative success perform only a single round of speaker deployment and training. Future work could perform further rounds of speaker deployment and listener judgments (i.e., as in Kojima et al., 2021; Suhr and Artzi, 2023), and analyze dynamics of language change in a continual learning setting.

References

Panos Achlioptas, Ahmed Abdelreheem, Fei Xia, Mohamed Elhoseiny, and Leonidas Guibas. 2020. Referit3d: Neural listeners for fine-grained 3d object identification in real-world scenes. In *Computer Vision – ECCV 2020*, pages 422–440, Cham. Springer International Publishing.

Herbert H. Clark. 1973. Space, time semantics and the child. In Timothy E. Moore, editor, *Cognitive Development and Acquisition of Language*, pages 27–63. Academic Press, San Diego.

Herbert H. Clark and Deanna Wilkes-Gibbs. 1986. Referring as a collaborative process. *Cognition*, 22(1):1–39.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Rui Fang, Malcolm Doering, and Joyce Chai. 2014. Collaborative models for referring expression generation in situated dialogue. *AAAI*.

Rui Fang, Malcolm Doering, and Joyce Y. Chai. 2015. Embodied collaborative referring expression generation in situated human-robot interaction. In *HRI*.

Juliana Goschler, Elena Andonova, and Robert J. Ross. 2008. Perspective use and perspective shift in spatial dialogue. In *Spatial Cognition VI. Learning, Reasoning, and Talking about Space*, pages 250–265, Berlin, Heidelberg. Springer Berlin Heidelberg.

Tanmay Gupta and Aniruddha Kembhavi. 2023. Visual programming: Compositional visual reasoning without training. In *CVPR*.

Janosch Haber, Tim Baumgärtner, Ece Takmaz, Lieke Gelderloos, Elia Bruni, and Raquel Fernández. 2019. The PhotoBook dataset: Building common ground through visually-grounded dialogue. In *ACL*.

Sang-Hun Han, Min-Gyu Park, Ju Hong Yoon, Ju-Mi Kang, Young-Jae Park, and Hae-Gon Jeon. 2023. High-fidelity 3d human digitization from single 2k resolution images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Robert D. Hawkins, Mike Frank, and Noah D. Goodman. 2017. Convention-formation in iterated reference games. *Cognitive Science*.

Jiani Huang, Calvin Smith, Osbert Bastani, Rishabh Singh, Aws Albarghouthi, and Mayur Naik. 2020. Generating programmatic referring expressions via program synthesis. In *ICML*.

Srinivasan Janarthanam and Oliver Lemon. 2010. Adaptive referring expression generation in spoken dialogue systems: Evaluation with real users. In *SIGDIAL*.

Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. 2014. ReferItGame: Referring to objects in photographs of natural scenes. In *EMNLP*.

Noriyuki Kojima, Alane Suhr, and Yoav Artzi. 2021. Continual learning for grounded instruction generation by observing human following behavior. *Transactions of the Association for Computational Linguistics*, 9:1303–1319.

Barbara Landau. 2017. Update on “what”' and “where”' in spatial language: A new division of labor for spatial terms. *Cognitive Science*, 41(S2):321–350.

Barbara Landau. 2018. Learning simple spatial terms: Core and more. *Topics in Cognitive Science*, 12(1):91–114.

S. C. Levinson and D. P. Wilkins. 2006. *Grammars of space: Explorations in cognitive diversity*. New York: Cambridge University Press.

Andy Liu, Hao Zhu, Emmy Liu, Yonatan Bisk, and Graham Neubig. 2023. Computational language acquisition with theory of mind. In *The Eleventh International Conference on Learning Representations*.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024. Improved baselines with visual instruction tuning. In *CVPR*.

Chuofan Ma, Yi Jiang, Jiannan Wu, Zehuan Yuan, and Xiaojuan Qi. 2024. Groma: Localized visual tokenization for grounding multimodal large language models. *Preprint*, arXiv:2404.13013.

Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. 2019. Habitat: A platform for embodied ai research. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9339–9347.

Michael F. Schober. 1993. Spatial perspective-taking in conversation. *Cognition*, 47(1):1–24.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *Preprint*, arXiv:1707.06347.

Sanjay Subramanian, Medhini Narasimhan, Kushal Khangaonkar, Kevin Yang, Arsha Nagrani, Cordelia Schmid, Andy Zeng, Trevor Darrell, and Dan Klein. 2023. Modular visual question answering via code generation. In *ACL*.

Alane Suhr and Yoav Artzi. 2023. Continual learning for instruction following from realtime feedback. In *Proceedings of the Conference on Neural Information Processing Systems*.

Dídac Surís, Sachit Menon, and Carl Vondrick. 2023. ViperGPT: Visual inference via python execution for reasoning. *CVPR*.

Holly A. Taylor and Barbara Tversky. 1996. Perspective in spatial descriptions. *Journal of Memory and Language*, 35(3):371–391.

Takuma Udagawa and Akiko Aizawa. 2019. A natural language corpus of common grounding under continuous and partially-observable context. *AAAI*.

Takuma Udagawa, Takato Yamazaki, and Akiko Aizawa. 2020. A linguistic analysis of visually grounded dialogues based on spatial expressions. In *Findings of EMNLP*.

Ielka van der Sluis and Saturnino Luz. 2011. A cross-linguistic study on the production of multimodal referring expressions in dialogue. In *European Workshop on Natural Language Generation*.

Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. 2023. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *Proceedings of the International Conference on Computer Vision (ICCV)*.

Haoxuan You, Haotian Zhang, Zhe Gan, Xianzhi Du, Bowen Zhang, Zirui Wang, Liangliang Cao, Shih-Fu Chang, and Yinfei Yang. 2023. Ferret: Refer and ground anything anywhere at any granularity. *arXiv preprint arXiv:2310.07704*.

A Data

A.1 Scene Generation

We include more details on scene generation in addition to in Sec.2.2.

Agent placement. We impose three constraints on agent placement to help a more efficient scene generation pipeline:

- Maximum distance between the agents: Let $d_{\max}$ be the maximum allowed distance between the speaker and the listener. Denoting the positions of the speaker and listener as ρ_s and ρ_l, respectively, we require that $|\rho_s - \rho_l| \leq d_{\max}$. We use $d_{\max} = 10$.

- Field of view overlap: Let Fov_s and Fov_l be the fields of view of the speaker and listener, respectively. We require that the intersection of their fields of view is non-empty, i.e., $\mathrm{Fov}_s \cap \mathrm{Fov}_l \neq \emptyset$.

- Relative viewing angle: Let ψ_s and ψ_l be the horizontal viewing angles of the speaker and listener, respectively, relative to a common reference direction. The relative viewing angle between the agents is given by $\psi' = \min(|\psi_s - \psi_l|, 360° - |\psi_s - \psi_l|)$. We can place the agents with a pre-set relative viewing angle by satisfying $C_0 \leq |\psi'_s - \psi'_l| \leq C_1$, where C_0, C_1 is the viewing angle difference bounds we set.

Referent placement. We impose three constraints on referents placement so they don't stack, become obstructed, or float in the air to meet real world physics standards:

- Visibility constraint: Let Vis_s and Vis_l be the sets of points visible from the speaker's and listener's cameras, respectively. For each referent r_i, we require that $r_i \in \mathrm{Vis}_s \cap \mathrm{Vis}_l$.

- Physically-based placement: Let $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$ be the sets of valid x, y, and z coordinates within the environment bounds. For each referent r_i, we randomly sample coordinates $(x_i, y_i, z_i) \in \mathcal{X} \times \mathcal{Y} \times \mathcal{Z}$ and drop the referent using gravitational physical simulation until it comes to rest on a solid horizontal surface.

- Minimum distance: Let $d_{\min}$ be the minimum required distance between any two referents. For all pairs of referents r_i and r_j, where $i \neq j$, we enforce $|r_i - r_j| \geq d_{\min}$. We use $d_{\min} = 0.3$.

Scene rendering. Our environment supports rendering observations at different resolutions; e.g., we use H = 720 and W = 1280 for HD resolution.

Scene rejection sampling. We use GPT-4 which is a Vision Language Model (VLM) to skip low quality images rendering during the dataset generation. We use the below prompt:

Please analyze the following image and provide a score from 0 to 10 based on these criteria:

- *The image must contain exactly 3 red spheres. If there are more or fewer than 3 red spheres, the score should be 0.*

- *The image should have high perceptual quality. Consider factors such as:*
  - ***Resolution**: The image should be clear and not pixelated or blurry.*
  - ***Lighting**: The image should have adequate lighting, without extreme darkness or overexposure.*
  - ***Focus**: The subject of the image (the red spheres) should be in focus.*
  - ***Contrast**: The image should have good contrast, allowing the red spheres to be easily distinguishable from the background.*

- *The image should not have any visible artifacts, such as:*
  - ***Compression artifacts**: There should be no visible compression artifacts, such as blocky patterns or color banding.*
  - ***Noise**: The image should not have excessive noise or graininess.*
  - ***Distortions**: The image should not have any distortions, such as warping or stretching.*

A.2 Adversarial Referent Placement

We present more details on training the adversarial placements Sec.2.3. For each training iteration, the vision transformer (ViT-s/16) will take in the speaker view and available object placement locations and speaker and listener locations processed as (x, y, z) coordinates flattened into a noramlized array. The model will be learned to output the hard location from the input object placement locations as a single-choice pipeline.

A.3 Crowdsourcing

For speakers and listeners we prompt the user to follow a description and a tutorial. When annotating, they still have access to the tutorial. We include description as below:

We engage participants in a virtual environment where they assume the roles of a Speaker and a Listener. The task involves communication and spatial reasoning, requiring the "Speaker" to describe the location of specific objects within the environment, which are visible to them but not to the Listener. The Listener then interprets these descriptions to identify the objects accurately. Data collected from these interactions helps us understand the effectiveness of communication strategies and spatial language in varied settings. This study aims to improve collaborative tasks between humans and AI agents, enhancing how they interact within digital and real-world environments.

We choose participants from USA, fluent in English. We tell the users the data will be used for research purpose. The study is determined exempt from ethics review.

We manually check human data for non-conforming text. This step includes excluding private user information or offensive content.

B Experiments

B.1 Experimental Setup

For environment generation, we use Quadro RTX 6000 for graphics rendering for a single process. We parallize data generation with Habitat-Sim with 4 Quadro RTX 6000.

We prompt the instruction-tuned vision and language models to output speaker and listener text. Except for the model-specific architecture input formatting. We use the following prompts:

Speaker Prompt:

Describe the location of the blue sphere relative to the environment features in contrast with other red spheres.

Listener Prompt:

Imagine an image filled with several identical red spheres and a blue sphere. Your task is to identify the specific red sphere of interest from among several possible candidates. To assist you, you will receive a detailed description highlighting unique characteristics or positions of the sphere.

Your objective is to determine the precise location of this sphere in the image and mark it with a bounding box. Consider factors such as lighting, reflections, shadows, relative position to other objects, and any unique attributes mentioned in the description. You should analyze how these details help to pinpoint the exact sphere among the identical ones.

Once you have identified the sphere, outline its position using a bounding box and provide its coordinates in the format:

x_0 (left), y_0 (top), x_1 (right), y_1 (bottom)

Additionally, explain your reasoning in detail for why you chose this specific location for the bounding box. For example:

"Based on the description, the sphere is near the window on the left side, and the distinct light reflection on its surface sets it apart from the others. This suggests its location as... , Bounding box coordinates: [0.23, 0.44, 0.30, 0.46]."

Be aware that the description might offer a different viewpoint of the scene, so be prepared to adjust your analysis accordingly.

Format for Response:

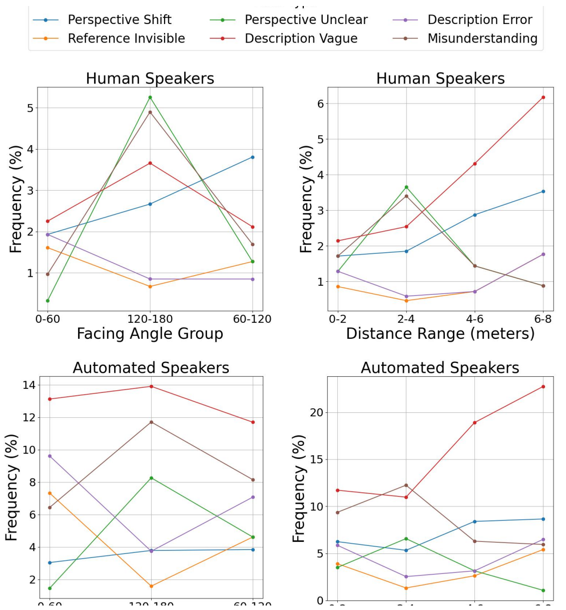

Figure 4: Impact of task difficulty on communication errors between speaker and listener.

Reasoning for location choice: *[Your detailed explanation here]*

Bounding box coordinates: *[x_0, y_0, x_1, y_1]*

Feel free to incorporate any nuanced observations or contrasting elements that helped you make the distinction.

B.2 Error Analysis

We analyze the frequency of several common communication errors in collaborative tasks involving both human and automated speakers interacting with human listeners, with varying degrees of task difficulty. For automated speakers, we utilize the LLaVA-1.5 model. The results are presented in Fig 4. It is evident that the error frequency in collaborations involving automated speakers is generally higher than that with human speakers, and these errors are predominantly vague descriptions. Conversely, human speakers more frequently encounter perspective shift issues, as they tend to use themselves or the listeners as reference points, whereas automated speakers prefer to reference fixed objects in the scene.

The impact of facing angles and distances on communication is also significant. We find that errors are most prevalent when the listener and speaker are facing each other at angles between 120-180 degrees. In these situations, directional terms such as "left" and "right" often become inverted, especially when speakers fail to clarify whose perspective is being used. Moreover, with the visibility of both parties, a speaker might use "human" as a reference point, but the listener typically assumes "human" refers to the speaker, leading to selections in the opposite direction. Additionally, as the distance between speaker and listener increases, the descriptions provided by speakers tend to become more vague, opting for broader reference points such as 'on the left side of the wall' rather than 'next to the table', further complicating accurate communication.

B.3 Ai Assistants Usage

When conducting the research, we use Ai to enchance our coding efficiency and quality. We use ChatGPT [8] and Claude,ai[9] to write codes for our dataset generation and human study websites server.

[8]https://chat.openai.com/
[9]https://claude.ai