Lang2LTL-2: Grounding Spatiotemporal Navigation Commands Using Large Language and Vision-Language Models

Jason Xinyu Liu, Ankit Shah, George Konidaris, Stefanie Tellex, David Paulius

Brown University, USA

Abstract

Grounding spatiotemporal navigation commands to structured task specifications enables autonomous robots to understand a broad range of natural language and solve long-horizon tasks with safety guarantees. Prior works mostly focus on grounding spatial or temporally extended language for robots. We propose Lang2LTL-2, a modular system that leverages pretrained large language and vision-language models and multimodal semantic information to ground spatiotemporal navigation commands in novel city-scaled environments without retraining. Lang2LTL-2 achieves 93.53% language grounding accuracy on a dataset of 21,780 semantically diverse natural language commands in unseen environments. We run an ablation study to validate the need for different modalities. We also show that a physical robot equipped with the same system without modification can execute 50 semantically diverse natural language commands in both indoor and outdoor environments.

Introduction

When giving directions, humans often use natural language that describes goals, as well as temporal and spatial constraints. For example, consider the command "Visit the Starbucks, only then go to the red car to the right of the building, and always avoid the crowded restaurant near the cafe." An autonomous robot following this spatiotemporal command must understand that it specifies a temporally extended task of visiting two locations in a strict order while avoiding the third throughout the execution. The robot must ground the three referring expressions, i.e., "the Starbucks," "the car," and "the crowded restaurant," to specific locations with respect to other landmarks in the environment.

Existing approaches focus on developing the robot's spatial or temporal reasoning ability separately. Many works develop systems to ground natural language commands that contain rich spatial relations in indoor (Anderson et al. 2018; Ku et al. 2020; Zheng et al. 2021) and outdoor (Chen et al. 2019; Shah et al. 2023b) environments. Their approaches use a map that contains multimodal semantic information to identify various target landmarks with respect to others in the environment, yet they cannot handle complex temporal constraints. Separately, structured task specifications, like linear temporal logic (LTL), can capture a wide range of semantically diverse temporal patterns (Menghi et al. 2021) and enable the synthesis of verifiable robot behaviors with safety guarantees. However, systems that can ground complex temporal language have limited spatial reasoning capability (Gopalan et al. 2018; Liu et al. 2023; Chen et al. 2023).

Figure 1: Our system Lang2LTL-2 grounds spatiotemporal navigation commands in indoor and outdoor environments. The spatial and temporal components of the example commands are highlighted in blue and red, respectively.

To achieve the best of both worlds, we introduce a modular language grounding system, Lang2LTL-2, that grounds spatiotemporal navigation commands for robots. Lang2LTL-2 uses large language models (LLMs) to recognize spatial referring expressions, like "the red car to the right of the building," and to translate language commands to LTL task specifications, which are compatible with many planning and reinforcement learning algorithms (Littman et al. 2017; Camacho et al. 2019; Oh et al. 2019; Icarte et al. 2022; Liu et al. 2024). Using pretrained vision-language models (VLMs) and text embedding, Lang2LTL-2 grounds referring expressions to specific locations in novel city-scaled environments without retraining, given a semantic database of textual and visual descriptions of the landmarks.

We evaluated Lang2LTL-2 on a dataset of 21,780 semantically diverse spatiotemporal commands with 1,723 spatial referring expressions, 19 spatial relations, and 15 temporal patterns. We also ran an ablation study and showed that using multimodal semantic information for spatiotemporal

language grounding outperforms using any modality alone. Finally, we demonstrated that a mobile robot equipped with the same system without modification could execute 50 semantically diverse spatiotemporal commands in both indoor and outdoor environments.

Preliminaries

Large Language Models and Vision-Language Models

Large language models (LLMs) are transformer neural networks (Vaswani et al. 2017) trained to maximize the probability of a successive token given a context window. They achieve state-of-the-art (SoTA) performance on a wide variety of natural language processing tasks (Radford et al. 2019). Pretrained LLMs can also produce high-dimensional embedding vectors of text. We can measure the semantic similarity of two pieces of text by computing the cosine similarity of their embeddings. In this work, we used OpenAI's GPT-4 model (OpenAI 2023) and embedding API for text completion and text embedding, respectively, and a fine-tuned T5-base model (Raffel et al. 2020) to translate natural language commands to temporal task specification.

Vision-language models (VLMs) are multimodal models jointly trained on text and images (Radford et al. 2021). They produce SoTA results on many language-conditioned vision tasks (Zhang et al. 2024), e.g., open-vocabulary object detection (Lüddecke and Ecker 2022; Minderer et al. 2022), image captioning (Chen et al. 2022), image retrieval (Liu et al. 2021), and visual question answering (Du et al. 2023). In this work, we prompted the GPT-4V(ision) model (Yang et al. 2023) to generate captions for images of landmarks and objects.

Temporal Task Specification

Linear temporal logic (LTL) (Pnueli 1977) is a promising task specification language for human-centered specification elicitation (Gopalan et al. 2018; Berg et al. 2020; Shah, Li, and Shah 2020; Shah et al. 2023a), planning (Shah, Li, and Shah 2020; Liu et al. 2024), and reinforcement learning (Littman et al. 2017; Icarte et al. 2018). The syntax of LTL is defined through the following recursive grammar:

$$\varphi := \alpha \mid \neg\varphi \mid \varphi_1 \vee \varphi_2 \mid \mathbf{X}\varphi \mid \varphi_1 \ \mathbf{U} \ \varphi_2 \tag{1}$$

Here α represents an atomic Boolean proposition, and $\varphi,\ \varphi_1,\ \varphi_2$ are any valid LTL formulas. The operators $\neg$ (not) and $\vee$ (or) are identical to propositional logic operators. The formula $\mathbf{X}\varphi$ holds if φ holds at the next time step, and $\varphi_1 \ \mathbf{U} \ \varphi_2$ holds if φ_1 holds at least until φ_2 first holds, which must happen at the current or a future time. LTL syntax also admits abbreviated operators defined through the compositions of the primitive operators. In this work, we use the operators $\wedge$ (and), $\mathbf{F}$ (read "finally" or "eventually"), and $\mathbf{G}$ (read "globally" or "always"). $\mathbf{F}\varphi$ specifies that the formula φ must hold at least once in the future, and $\mathbf{G}\varphi$ specifies that φ must always hold.

Task Execution for Temporal Task Specification

Our language grounding system Lang2LTL-2 is compatible with many planning and reinforcement learning algorithms that solve LTL tasks (Littman et al. 2017; Camacho et al. 2019; De Giacomo et al. 2019; Oh et al. 2019; Icarte et al. 2022; Liu et al. 2024). We can transform LTL formulas to Büchi automata (Vardi 1996; Gerth et al. 1996). Transitions in the environment induce transitions in the automaton, so we can track task progress by tracking the automaton's state transition. We can then compute a policy on the product MDP of the task automaton and the environment MDP.

Problem Definition

Our language grounding system Lang2LTL-2 receives a natural language utterance u from the user that specifies a navigation task in an environment modeled as $\langle \mathcal{S}, \mathcal{A}, T \rangle$, where $\mathcal{S}$ and $\mathcal{A}$ represent the robot's states and actions, and $T(s, a) \to s'$ captures the transition dynamics. In this work, we consider navigational actions that transition a robot from one location to another in the environment represented as a semantic map. We assume the robot has access to a multimodal semantic database $\mathcal{D} = \{p : (d, f)\}$, where p is a proposition that uniquely represents a landmark in the environment, d is a semantic description of the landmark, and $f : \mathcal{S} \to \{0, 1\}$ is a Boolean-valued function that determines the true value of the proposition in a given state. The semantic description of a landmark can be a piece of text (including its name, amenity, street address, etc.), an image, or both. Lang2LTL-2 translates the input command to a linear temporal logic (LTL) formula φ and grounds its propositions to landmarks in the real world. We assume the robot can track its state in a semantic map and has access to an automated planner that, given an LTL formula as task specification, produces a trajectory in the semantic map. Many planning and reinforcement learning algorithms (Littman et al. 2017; Camacho et al. 2019; Oh et al. 2019; Icarte et al. 2022; Liu et al. 2024) are compatible with LTL task specification. We use the AP-MDP planner (Oh et al. 2019). Figure 2 shows an example execution by the full system, i.e., language grounding and planning.

Lang2LTL-2: Spatiotemporal Language Grounding

We approach the problem of spatiotemporal language grounding with a modular design, where we extract spatial referring expressions and translate temporal commands using large language models, and ground referring expressions to physical landmarks using a vision-language model and text embedding. Our system Lang2LTL-2 produces a grounded temporal task specification with grounded referring expressions and spatial relations. Figure 3 shows an overview of our language grounding system.

Spatial Referring Expression Recognition (SRER)

The spatial referring expression recognition (SRER) module identifies spatial referring expressions in a given language command. Referring expressions (REs) are noun phrases, pronouns, and proper names that refer to some entity in an

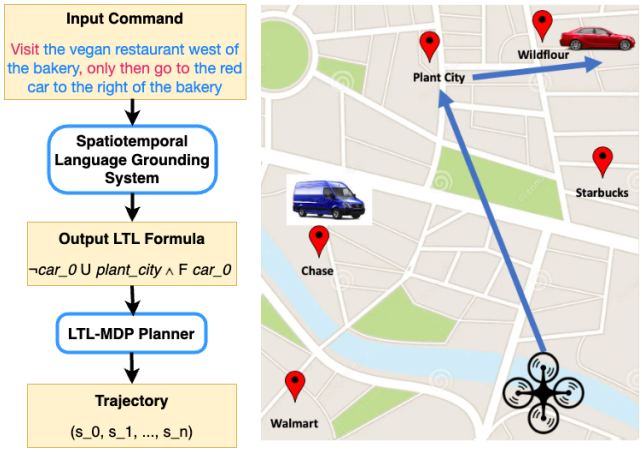


Figure 2: An example of an input spatiotemporal navigation command whose spatial and temporal components are highlighted in blue and red, respectively, an output LTL formula whose propositions are grounded to physical landmarks, and an execution trajectory in the environment.

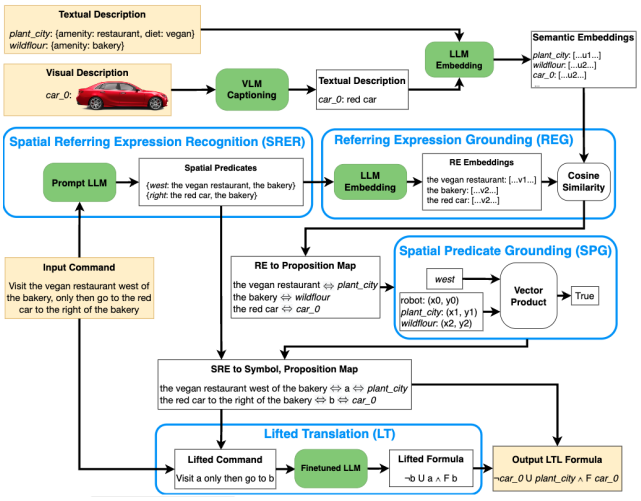


Figure 3: Lang2LTL-2 Language Grounding System Overview: input and output are in yellow blocks; modules are in blue blocks; pretrained and fine-tuned models are in green blocks.

environment, such as landmarks and objects (Lyons 1977). In this work, we only consider noun phrases and proper names and leave the coreference resolution problem to future work. Spatial referring expressions (SREs) are phrases where referring expressions are connected by a spatial relation. For example, in the language command "Go to the red car to the right of the bakery," the SRE "the red car to the right of the bakery" contains two REs, "the red car" and "the bakery," termed the figure e_f and the ground e_g, respectively, by Landau and Jackendoff (1993). The figure e_f and the ground e_g are connected by the spatial relation r "to the right of." We define a diverse set $\mathcal{R}$ of 19 spatial relations, such as near, in front of, behind, to the left of, to the right of, between, and four cardinal directions. The SRER module extracts referring expressions and their spatial relations from a spatiotemporal language command by prompting an LLM. We use GPT-4 (OpenAI 2023). The output of the SRER module is a spatial predicate denoted by $\{r : (e_f, e_g)\}$. Please see the supplementary materials for the complete set of spatial relations and the prompt used for SRER.

Referring Expression Grounding (REG)

To ground the referring expressions (REs) e_f and e_g to physical landmarks in the environment, we use a multimodal semantic database with textual and visual descriptions of landmarks. Having both modalities enables the referring expression grounding (REG) module to ground more complex REs and improves grounding accuracy. For certain REs, one modality is more descriptive than the other. For example, a textual description including a landmark's amenity and cuisine type better matches the RE "the vegan restaurant" than an image of the restaurant front. The REG module is important for identifying possible candidates for each RE, especially when the environment contains multiple similar landmarks or objects.

In practice, detailed textual descriptions of objects are not always available, e.g., "the red car," but can be extracted from images. We prompt a pretrained vision-language model (VLM) to generate captions of images with the question, "What is the most obvious object in this image?" In this work, we use GPT-4V(ision) (Yang et al. 2023). We then use an LLM to generate text embeddings for the image captions, the textual descriptions of landmarks in the semantic database, and the query REs (i.e., e_f and e_g) extracted from the language command. Finally, we use the cosine similarity between text embeddings to find the landmarks that best match the query REs. Let $g_{caption} : i \rightarrow t$ be the function that generates a caption t for image i parameterized by the weights of the VLM, and $g_{embed} : t \rightarrow z$ be the function that computes an n-dimensional embedding z of a text string t parameterized by the weights of the LLM. The cosine similarity score is defined as follows,

$$score(e_{f/g}, t) = \frac{g_{embed}(e_{f/g})^T g_{embed}(t)}{||g_{embed}(e_{f/g})|| \cdot ||g_{embed}(t)||}, \quad (2)$$

where we substitute $t = g_{caption}(i)$ when the semantic description of the landmark is an image i, and $e_{f/g}$ denotes the query RE being the figure e_f or the ground e_g.

We also explored using CLIP's text and image encoders (Radford et al. 2021) to encode text and images, then the cosine similarity of the text and image embeddings to find the best matching landmark for a query RE. However, we discovered that the gap between the text and image embedding spaces is large for the pretrained CLIP model. Liang et al. (2022) documented this phenomenon in more detail. Instead of training another neural network to align the text and image embedding spaces, we use a pretrained VLM to transcribe images to text and work solely in the text embedding space.

Spatial Predicate Grounding (SPG)

After grounding the figure e_f and the ground e_g to candidate landmarks, we perform spatial predicate grounding (SPG) to identify the most likely landmark referred to by e_f given e_g and the spatial relation r. We assume that users give commands with respect to the robot's initial location. For each spatial referring expression (SRE) and its corresponding spatial predicate $\{r : (e_f, e_g)\}$, we rank all the candidate landmarks of e_f based on the product of the similarity scores computed by the referring expression grounding (REG) module for the candidate landmarks of e_f and e_g, then select the proposition p_f with the highest product score,

$$p_f^* = \underset{p_f:(d_f,\_)\in\mathcal{D}, p_g:(d_g,\_)\in\mathcal{D}}{\arg\max} score(e_f, d_f) \cdot score(e_g, d_g). \tag{3}$$

To validate each pair of candidate landmarks, we first compute a ground vector from the ground landmark to the robot, which serves as an anchor for computing the range where the figure landmark should be. We then compute a figure vector from the ground landmark to the figure landmark. Based on the spatial relation, we compute a range where the figure vector should lie relative to the ground vector. Figure 4 illustrates the ground and the figure vectors for the SRE "the red car to the right of the bakery."

For each known spatial relation $r \in \mathcal{R}$, we specify a set of rules to validate a pair of candidate landmarks for e_f and e_g. In the example of "the red car to the right of the bakery," the spatial relation "to the right of" means the figure vector must lie within the half circle between the ground vector and 180 degrees counterclockwise from the ground vector. Please see the supplementary materials for the definition of all spatial relations. We also specify a distance threshold in meters between a figure and the ground to eliminate candidate figures too far from the ground. To resolve an unseen spatial relation, we use LLM text embedding and cosine similarity to find the most semantically similar spatial relation $r \in \mathcal{R}$.

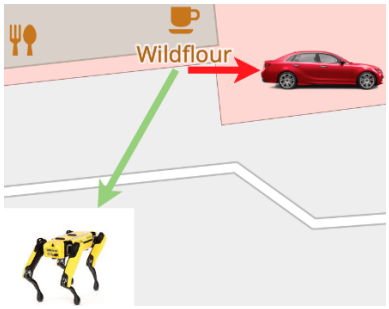


Figure 4: An illustration of the ground vector and the figure vector, depicted as the green and the red arrow, respectively, computed by the spatial predicate grounding (SPG) module (Section) to resolve the spatial referring expression "the red car to the right of the bakery."

Lifted Translation (LT)

After the SRER module extracts all the spatial referring expressions (SREs) from a given command, we transform it into a lifted command by substituting the SREs with symbols, which are grounded to physical landmarks by the REG (Section) and the SPG (Section) modules. For example, the input command "Go to the red car to the right of the bakery" is transformed to a lifted command "Go to a" where the symbol a substitutes the SRE "the red car to the right of the bakery." We then translate the lifted command to a lifted LTL formula compatible with many planning and reinforcement learning algorithms (Littman et al. 2017; Camacho et al. 2019; Oh et al. 2019; Icarte et al. 2022; Liu et al. 2024). We evaluate the following models for lifted translation.

Fine-tuned LLM: Liu et al. (2023) tested four models that use LLMs for lifted translation. The T5-Base (220M parameters) model (Raffel et al. 2020) fine-tuned on the semantically diverse dataset they collected overperformed the fine-tuned GPT-3 (Brown et al. 2020), the Prompt GPT-3 (Brown et al. 2020) and the Prompt GPT-4 (OpenAI 2023) models. Thus, we use their best-performing model fine-tuned T5-Base through HuggingFace's Transformer library (Wolf et al. 2020).

Retrieval Augmented Generation (RAG): We evaluate retrieval augmented generation (RAG), which dynamically constructs a prompt to an LLM based on the query (Lewis et al. 2020) for lifted translation. To translate a lifted command to a lifted LTL formula with RAG, we use cosine similarity of text embeddings to find semantically similar commands from the lifted dataset collected in (Liu et al. 2023), then use these commands and their corresponding LTL formulas as in-context examples to query GPT-4 (OpenAI 2023). We test varying numbers of in-context examples. Please see the supplementary materials for an example prompt used for RAG.

Evaluation of Language Grounding

We conducted three sets of evaluations of our spatiotemporal language grounding system Lang2LTL-2: 1) a modular evaluation, where we tested the performance of individual modules introduced in Section , 2) a full system evaluation, where we evaluated the final output of our system, and 3) an ablation study of the text and the image modality.

Language Grounding Dataset

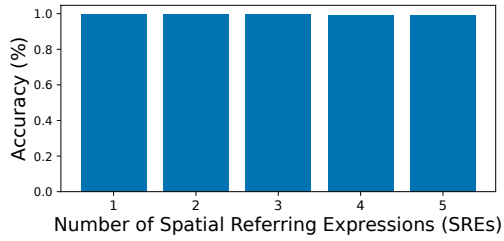
Our evaluation used four city-scaled environments with an increasing number of landmarks, i.e., 9, 34, 44, and 175. The landmarks were described by text from OpenStreetMap (Contributors 2017) (e.g., names, street addresses, amenities, and GPS coordinates, etc.) and images from Google StreetView (Anguelov et al. 2010). Having a dataset where landmarks are described by both modalities helps evaluate whether the referring expression grounding (REG) module can use a proper modality to ground referring expressions to the correct landmarks.

To obtain semantically diverse spatiotemporal navigation commands, we first collected 1,723 spatial referring expressions (SREs) with respect to the robot's initial location from human users, then substituted the SREs in the 1,089 lifted natural language commands provided by Liu et al. (2023).

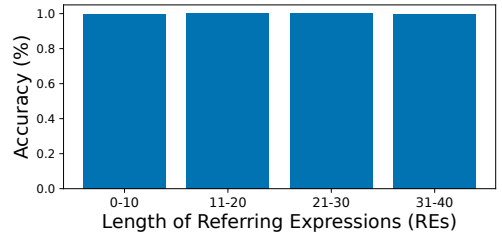
Table 1: Modular Performance

Module		**Accuracy** (averaged over five seeds)				
		City 1 (9 landmarks)	*City 2* (34 landmarks)	*City 3* (44 landmarks)	*City 4* (175 landmarks)	*Average*
SRER		$99.45 \pm 0.12\%$	$99.43 \pm 0.26\%$	$99.56 \pm 0.63\%$	$99.39 \pm 0.21\%$	$99.46 \pm 0.34\%$
REG	Top-1	$99.68 \pm 0.72\%$	$97.98 \pm 1.07\%$	$88.74 \pm 2.14\%$	$78.35 \pm 1.97\%$	$91.19 \pm 8.84\%$
	Top-5	$100.00 \pm 0.00\%$	$100.00 \pm 0.00\%$	$99.56 \pm 0.24\%$	$99.15 \pm 0.34\%$	$99.68 \pm 0.41\%$
	Top-10	$100.00 \pm 0.00\%$	$100.00 \pm 0.00\%$	$99.70 \pm 0.17\%$	$99.98 \pm 0.05\%$	$99.92 \pm 0.15\%$
SPG		$100.00 \pm 0.00\%$	$100.00 \pm 0.00\%$	$99.53 \pm 0.33\%$	$99.35 \pm 1.46\%$	$99.72 \pm 0.75\%$
LT	Finetuned T5-base	$99.45 \pm 0.00\%$	$99.45 \pm 0.00\%$	$99.45 \pm 0.00\%$	$99.45 \pm 0.00\%$	$99.45 \pm 0.00\%$
	RAG-10	$69.33 \pm 0.25\%$	$70.34 \pm 0.13\%$	$69.65 \pm 0.58\%$	$70.39 \pm 0.84\%$	$69.93 \pm 0.62\%$
	RAG-50	$83.79 \pm 0.06\%$	$83.93 \pm 0.12\%$	$83.75 \pm 0.52\%$	$83.93 \pm 0.65\%$	$83.85 \pm 0.33\%$
	RAG-100	$88.20 \pm 0.58\%$	$88.25 \pm 1.04\%$	$87.79 \pm 0.39\%$	$87.70 \pm 0.13\%$	$87.98 \pm 0.54\%$

The lifted commands cover 15 temporal patterns for common robotic tasks, each with 20 to 38 lifted commands. For example, given the lifted command "Visit a only then go to b", we can substitute the symbols a and b with the SREs "the vegan restaurant west of the bakery" and "the red car," respectively, to obtain the grounded natural language command "Visit the vegan restaurant west of the bakery only then go to the red car." We constructed 21,780 unique spatiotemporal language commands using five seeds to sample SREs for substitution. The commands contain varying numbers of SREs ranging from one to five.



(a) SRER Accuracy vs. Utterance Complexity



(b) REG Accuracy vs. RE Complexity

Figure 5: Figure 5a shows the accuracies of the spatial referring expression recognition (SRER) module as the complexity of utterances (measured by the number of SREs in an utterance) increases. Figure 5b shows the top-10 accuracy of the referring expression grounding (REG) module as the complexity of REs (measured by string length) increases.

Modular Evaluation

We first evaluated each module introduced in Section on the semantically diverse dataset introduced in Section . All results were averaged over five seeds.

Spatial Referring Expression Recognition (SRER): We evaluated the LLM's ability to correctly extract all spatial referring expressions (SREs) from a natural language command and identify their spatial predicates described in Section , i.e., $\{r : (e_f, e_g)\}$ with spatial relation r, figure e_f and ground e_g. As shown in Table 1, the SRER module can reliably recognize SREs and their corresponding spatial predicates in language commands from unseen environments. Figure 5a further demonstrates that SRER achieves nearly perfect performance across commands with varying numbers of SREs. Occasionally, the SRER module fails to parse an SRE to the correct spatial predicate for an input command containing five long SREs.

Referring Expression Grounding (REG): We evaluated the REG module's ability to ground referring expressions, i.e., figures e_f's and grounds e_g's, to the correct physical landmarks described by text and images in the semantic map. We observe in Table 1 that the top-1 accuracy decreases as the number of landmarks increases from City 1 to City 4. Cities with more landmarks contain more instances that share similar textual or visual features. For example, there may be multiple cafe shops or red bicycles in a large environment. However, as we increase the number of top candidates from 1 to 10, REG achieves nearly perfect accuracy. Since the REG module provides candidate landmarks of figures and grounds to the SPG module (evaluated next), we hypothesize that as long as the correct landmark is among the top candidates, our system can still ground figures to the correct landmarks. We used 10 as the number of candidates for REG. Figure 5b shows that as the complexity of REs increases, the REG module consistently achieves near-perfect top-10 accuracies. These results align with those reported by Liu et al. (2023).

Spatial Predicate Grounding (SPG): We evaluated whether the SPG module could identify the correct figure landmarks using spatial reasoning described in Section . As shown in Table 1, the SPG module performs uniformly well across all environments. The few failure cases were due to the instances where the distance between the figure and

the ground landmarks was larger than the search threshold. Please see the supplementary materials for a breakdown of the accuracy per spatial relation.

Lifted Translation (LT): Liu et al. (2023) conducted a comprehensive evaluation of the generalization capability of various fine-tuned and pretrained LLMs for lifted translation. We compared the accuracies of the best-performing model in (Liu et al. 2023), i.e., T5-base fine-tuned on a large composed dataset, and retrieval augmented generation (RAG) (Lewis et al. 2020) with varying numbers of in-context examples. The fine-tuned model achieved the highest accuracies across all environments. As we increase the number of in-context examples for RAG from 10 to 100, the maximum tokens allowed by GPT-4 (OpenAI 2023), we observe that the accuracy increases but is lower than that of the fine-tuned model. Thus, we used the fine-tuned T5-base model for lifted translation in our system. For cost effective reasons, we averaged the RAG results over two seeds per city.

Full System Evaluation and Ablation Study

We tested the overall performance of our language grounding system Lang2LTL-2, which takes a spatiotemporal navigation command as input and produces an LTL formula whose propositions are grounded to physical landmarks in the environment. The full system achieved an accuracy of $93.53 \pm 4.33\%$ on a dataset of 21,780 randomly sampled semantically diverse language commands.

To evaluate the effectiveness of using multimodal semantic information for language grounding, we conducted an ablation study where we only used one modality, i.e., text or images, in the referring expression grounding (REG) module. As shown in Figure 6, the full system using both modalities significantly outperformed the text-only and the image-only systems because either modality alone often did not provide enough semantic information. For example, an image of a restaurant front does not specify its cuisine being vegan, thus it will not be useful for grounding the referring expression "the vegan restaurant." However, the additional visual features provided by images can further disambiguate similar landmarks. For example, colors can help disambiguate a red and a yellow bicycle. In practice, detailed textual descriptions of landmarks are not always available, e.g., "the red brick building," but can be easily extracted from images by querying a pretrained VLM for image captions. Note that the text-only system is the same as Lang2LTL (Liu et al. 2023) with an additional spatial reasoning capability. Liu et al. (2023) showed that Lang2LTL outperforms Code-as-Policies (Liang et al. 2023), a prominent system that grounds natural language instructions to Python code directly executable on robots.

The accuracy of the spatial predicate grounding (SPG) module when given the top-10 candidate groundings from the referring expression grounding (REG) module was $97.26 \pm 2.07\%$. It supports our hypothesis that if the correct grounding landmark is among the top candidates output by the REG module, the SPG module can identify the correct figure landmark based on spatial reasoning.

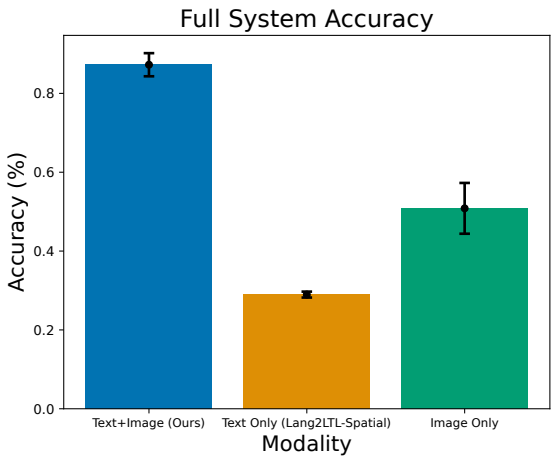


Figure 6: A comparison of the average accuracies of spatiotemporal language grounding systems using different modalities across four environments and five seeds per environment.

Robot Demonstration

To demonstrate Lang2LTL-2's ability to inform an automated planner and enable the execution of spatiotemporal commands, we deploy the same system without modification at the task planning level on a quadruped robot Spot (Boston Dynamics) in an indoor and outdoor environment. These environments contain nine and five objects, respectively, with multiple objects and landmarks that have similar textual or visual features, e.g., tables, couches, buildings, dumpsters, and cars.

We use Spot's GraphNav software to build a semantic map of the environment and capture images of landmarks and objects of interest. We only use the image modality for indoor experiments. For the outdoor environment, we additionally download textual descriptions of landmarks in the region from OpenStreetMap (Contributors 2017). Given a grounded LTL task specification output by our language grounding system Lang2LTL-2, we use the AP-MDP planner (Oh et al. 2019) to produce a sequence of locations through the semantic map and Spot's onboard motion planner to move between two locations. We executed 50 semantically diverse spatiotemporal natural language commands on the physical robot. With the formal safety guarantee offered by LTL and the AP-MDP planner, the robot was able to abort the execution when a given task was infeasible. Please see the supplementary materials for the list of all the language commands.

Related Work

Most existing robotic language grounding works focus on either spatial or temporal commands, with a few papers on grounding spatiotemporal commands to a limited capacity (Tellex et al. 2020; Cohen et al. 2024).

Grounding Spatial Commands for Robots

SLOOP (Zheng et al. 2021) is a system that grounds spatial commands in partially observable environments by using the spatial relations between a target object and multiple landmarks to construct an initial belief for a POMDP planner. LanguageRefer (Roh et al. 2022) is a learned transformer-based model that takes as inputs a spatial language command, a 3D point cloud of the scene, and bounding boxes of objects, then predicts the target object. RoboHop (Garg et al. 2024) builds a topological map of the environment with image segments as nodes. Like our work, RoboHop uses an LLM to extract referring expressions (REs) from a language command then a VLM to ground REs to nodes in the topological map.

Grounding Temporal Commands for Robots

Linear temporal logic (LTL) (Pnueli 1977) is a mathematically precise language that can specify robotic tasks and provide satisfaction guarantees, especially for long-horizon, temporally-extended tasks with non-Markovian rewards. Early work of using LTL for temporal command grounding was limited to structured language (Kress-Gazit, Fainekos, and Pappas 2007). Gopalan et al. (2018) trained a Seq2Seq network (Sutskever, Vinyals, and Le 2014) on natural language and LTL pairs in every new environment to ground language commands for navigation and manipulation. Like our work, Berg et al. (2020) and Hsiung et al. (2022) first translated commands to lifted LTL formulas then grounded the propositions to landmarks or objects but used a Seq2Seq network with limited capabilities.

To mitigate the lack of training data, Pan, Chou, and Berenson (2023) used an LLM to paraphrase structured language commands constructed from algorithmically generated LTL formulas. Patel, Pavlick, and Tellex (2020) and Wang et al. (2021) proposed weakly supervised methods that use executed trajectories instead of LTL annotations to guide language grounding. Lang2LTL (Liu et al. 2023) is also a modular system that uses LLMs to ground temporally extended navigation commands in indoor and outdoor environments without retraining, given a text-based semantic database. However, Lang2LTL cannot ground spatial referring expressions or landmarks with visual descriptions. Our system Lang2LTL-2 improves upon Lang2LTL by incorporating spatial reasoning and using a VLM to process images.

Grounding Spatiotemporal Commands for Robots

Language commands from existing works of indoor (Anderson et al. 2018; Ku et al. 2020; Zheng et al. 2021; Quartey et al. 2024) and outdoor (Chen et al. 2019; Shah et al. 2023b) navigation are rich in spatial relations, but lack diverse temporal patterns. LM-Nav (Shah et al. 2023b) uses an LLM to extract a sequence of referring expressions (REs) from a navigation command, then a VLM to ground the REs to images of physical landmarks. LM-Nav only grounds language commands of the sequenced visit type defined in (Menghi et al. 2021). VLMaps (Huang et al. 2023) fuses pretrained vision-language features with depth information to construct a spatial map of the environment then directly indices a sequence of spatial referring expressions (SREs) extracted by an LLM in the map. LIMP (Quartey et al. 2024) uses RGB-D information, an LLM, and a VLM to construct a 3D semantic map conditioned on the input language for motion planning to solve indoor mobile manipulation tasks. It translates free-form language commands to one of three temporal patterns using an LLM. Our system Lang2LTL-2 can ground language commands containing 15 temporal patterns commonly used in robotics (Menghi et al. 2021). An additional advantage of Lang2LTL-2 is its ability to ground REs that are not easily represented by visual description, e.g., generic referring expressions like "the vegan restaurant," and proper names like "Wildflour" (the name of a bakery) by using additional textual description from OpenStreetMap (Contributors 2017) in grounding city-scaled navigation commands.

Conclusion

We propose a modular language grounding system that consists of pretrained large language and vision-language models to ground spatiotemporal navigation commands to landmarks described by text and images in a semantic map of novel indoor and outdoor environments. We evaluate the individual modules and the full language grounding system on a semantically diverse dataset of 21,780 spatiotemporal navigation commands in four novel city-scaled environments. Our system achieved 93.53% accuracy, outperforming the previous SoTA. An autonomous robot with access to a semantic map and position tracking can use the same system without modification to follow spatiotemporal navigation commands in novel indoor and outdoor environments. We envision incorporating interaction with human users to further improve spatiotemporal language grounding.

References

Anderson, P.; Wu, Q.; Teney, D.; Bruce, J.; Johnson, M.; Sünderhauf, N.; Reid, I.; Gould, S.; and Van Den Hengel, A. 2018. Vision-and-Language Navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3674–3683.

Anguelov, D.; Dulong, C.; Filip, D.; Frueh, C.; Lafon, S.; Lyon, R.; Ogale, A.; Vincent, L.; and Weaver, J. 2010. Google Street View: Capturing the World at Street Level. *Computer*, 43(6): 32–38.

Berg, M.; Bayazit, D.; Mathew, R.; Rotter-Aboyoun, A.; Pavlick, E.; and Tellex, S. 2020. Grounding Language to Landmarks in Arbitrary Outdoor Environments. In *2020 IEEE International Conference on Robotics and Automation*, 208–215. IEEE.

Boston Dynamics. ???? Spot® - The Agile Mobile Robot. https://www.bostondynamics.com/products/spot.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; Agarwal, S.; Herbert-Voss, A.; Krueger, G.; Henighan, T.; Child, R.; Ramesh, A.; Ziegler, D.; Wu, J.; Winter, C.; Hesse, C.; Chen, M.; Sigler, E.; Litwin, M.; Gray, S.;

Chess, B.; Clark, J.; Berner, C.; McCandlish, S.; Radford, A.; Sutskever, I.; and Amodei, D. 2020. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*, volume 33, 1877–1901.

Camacho, A.; Icarte, R. T.; Klassen, T. Q.; Valenzano, R. A.; and McIlraith, S. A. 2019. LTL and Beyond: Formal Languages for Reward Function Specification in Reinforcement Learning. In *IJCAI*, volume 19, 6065–6073.

Chen, H.; Suhr, A.; Misra, D.; Snavely, N.; and Artzi, Y. 2019. Touchdown: Natural Language Navigation and Spatial Reasoning in Visual Street Environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12538–12547.

Chen, J.; Guo, H.; Yi, K.; Li, B.; and Elhoseiny, M. 2022. VisualGPT: Data-Efficient Adaptation of Pretrained Language Models for Image Captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18030–18040.

Chen, Y.; Gandhi, R.; Zhang, Y.; and Fan, C. 2023. NL2TL: Transforming Natural Languages to Temporal Logics using Large Language Models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 15880–15903.

Cohen, V.; Liu, J. X.; Mooney, R.; Tellex, S.; and Watkins, D. 2024. A Survey of Robotic Language Grounding: Tradeoffs between Symbols and Embeddings. In *International Joint Conference on Artificial Intelligence*.

Contributors, O. 2017. Planet OSM. https://www.openstreetmap.org.

De Giacomo, G.; Iocchi, L.; Favorito, M.; and Patrizi, F. 2019. Foundations for Restraining Bolts: Reinforcement Learning with LTLf/LDLf Restraining Specifications. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 29, 128–136.

Du, Y.; Li, J.; Tang, T.; Zhao, W. X.; and Wen, J.-R. 2023. Zero-shot Visual Question Answering with Language Model Feedback. In *Findings of the Association for Computational Linguistics*, 9268–9281. Association for Computational Linguistics.

Garg, S.; Rana, K.; Hosseinzadeh, M.; Mares, L.; Suenderhauf, N.; Dayoub, F.; and Reid, I. 2024. RoboHop: Segment-based Topological Map Representation for Open-World Visual Navigation. In *IEEE International Conference on Robotics and Automation*. IEEE.

Gerth, R.; Peled, D.; Vardi, M. Y.; and Wolper, P. 1996. Simple On-the-fly Automatic Verification of Linear Temporal Logic. In *Protocol Specification, Testing and Verification XV: Proceedings of the Fifteenth IFIP WG6. 1 International Symposium on Protocol Specification, Testing and Verification*, 3–18. Springer.

Gopalan, N.; Arumugam, D.; Wong, L. L.; and Tellex, S. 2018. Sequence-to-Sequence Language Grounding of Non-Markovian Task Specifications. In *Robotics: Science and Systems*.

Hsiung, E.; Mehta, H.; Chu, J.; Liu, J. X.; Patel, R.; Tellex, S.; and Konidaris, G. 2022. Generalizing to New Domains by Mapping Natural Language to Lifted LTL. In *International Conference on Robotics and Automation*, 3624–3630. IEEE.

Huang, C.; Mees, O.; Zeng, A.; and Burgard, W. 2023. Visual Language Maps for Robot Navigation. In *IEEE International Conference on Robotics and Automation*, 10608–10615. IEEE.

Icarte, R. T.; Klassen, T.; Valenzano, R.; and McIlraith, S. 2018. Using Reward Machines for High-Level Task Specification and Decomposition in Reinforcement Learning. In *International Conference on Machine Learning*, 2107–2116. PMLR.

Icarte, R. T.; Klassen, T. Q.; Valenzano, R.; and McIlraith, S. A. 2022. Reward Machines: Exploiting Reward Function Structure in Reinforcement Learning. *Journal of Artificial Intelligence Research*, 73: 173–208.

Kress-Gazit, H.; Fainekos, G. E.; and Pappas, G. J. 2007. From Structured English to Robot Motion. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2717–2722. IEEE.

Ku, A.; Anderson, P.; Patel, R.; Ie, E.; and Baldridge, J. 2020. Room-Across-Room: Multilingual Vision-and-Language Navigation with Dense Spatiotemporal Grounding. In *Conference on Empirical Methods for Natural Language Processing*.

Landau, B.; and Jackendoff, R. 1993. "What" and "where" in spatial language and spatial cognition. *Behavioral and Brain Sciences*, 16: 217–265.

Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.-t.; Rocktäschel, T.; et al. 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. *Advances in Neural Information Processing Systems*, 33: 9459–9474.

Liang, J.; Huang, W.; Xia, F.; Xu, P.; Hausman, K.; Ichter, B.; Florence, P.; and Zeng, A. 2023. Code as Policies: Language Model Programs for Embodied Control. In *IEEE International Conference on Robotics and Automation*.

Liang, V. W.; Zhang, Y.; Kwon, Y.; Yeung, S.; and Zou, J. Y. 2022. Mind the Gap: Understanding the Modality Gap in Multi-modal Contrastive Representation Learning. *Advances in Neural Information Processing Systems*, 35: 17612–17625.

Littman, M. L.; Topcu, U.; Fu, J.; Isbell, C.; Wen, M.; and MacGlashan, J. 2017. Environment-Independent Task Specifications via GLTL. *arXiv preprint arXiv:1704.04341*.

Liu, J. X.; Shah, A.; Rosen, E.; Jia, M.; Konidaris, G.; and Tellex, S. 2024. LTL-Transfer: Skill Transfer for Temporally-Extended Task Specifications. *IEEE International Conference on Robotics and Automation*.

Liu, J. X.; Yang, Z.; Idrees, I.; Liang, S.; Schornstein, B.; Tellex, S.; and Shah, A. 2023. Lang2LTL: Grounding Complex Natural Language Commands for Temporal Tasks in Unseen Environments. In *Conference on Robot Learning*, 1084–1110. PMLR.

Liu, Z.; Rodriguez-Opazo, C.; Teney, D.; and Gould, S. 2021. Image Retrieval on Real-Life Images With Pre-Trained Vision-and-Language Models. In *Proceedings of*

the IEEE/CVF International Conference on Computer Vision, 2125–2134.

Lüddecke, T.; and Ecker, A. 2022. Image Segmentation Using Text and Image Prompts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7086–7096.

Lyons, J. 1977. *Semantics: Volume 2*, volume 2. Cambridge University Press.

Menghi, C.; Tsigkanos, C.; Pelliccione, P.; Ghezzi, C.; and Berger, T. 2021. Specification Patterns for Robotic Missions. *IEEE Transactions on Software Engineering*, 47(10): 2208–2224.

Minderer, M.; Gritsenko, A.; Stone, A.; Neumann, M.; Weissenborn, D.; Dosovitskiy, A.; Mahendran, A.; Arnab, A.; Dehghani, M.; Shen, Z.; et al. 2022. Simple Open-Vocabulary Object Detection. In *European Conference on Computer Vision*, 728–755. Springer.

Oh, Y.; Patel, R.; Nguyen, T.; Huang, B.; Pavlick, E.; and Tellex, S. 2019. Planning with State Abstractions for Non-Markovian Task Specifications. In *Robotics: Science and Systems*, volume 2019.

OpenAI. 2023. GPT-4 Technical Report. *arXiv preprint arXiv:2303.08774*. Accessed the model on January 9, 2025.

Pan, J.; Chou, G.; and Berenson, D. 2023. Data-Efficient Learning of Natural Language to Linear Temporal Logic Translators for Robot Task Specification. In *IEEE International Conference on Robotics and Automation*. IEEE.

Patel, R.; Pavlick, E.; and Tellex, S. 2020. Grounding Language to Non-Markovian Tasks with No Supervision of Task Specifications. In *Robotics: Science and Systems*, volume 2020.

Pnueli, A. 1977. The Temporal Logic of Programs. In *18th Annual Symposium on Foundations of Computer Science (SFCS)*, 46–57. ieee.

Quartey, B.; Rosen, E.; Tellex, S.; and Konidaris, G. 2024. Verifiably Following Complex Robot Instructions with Foundation Models. *arXiv preprint arXiv:2402.11498*.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; Krueger, G.; and Sutskever, I. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning*, 8748–8763. PMLR.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I.; et al. 2019. Language Models are Unsupervised Multitask Learners. *OpenAI blog*, 1(8): 9.

Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research*, 21(140): 1–67.

Roh, J.; Desingh, K.; Farhadi, A.; and Fox, D. 2022. LanguageRefer: Spatial-Language Model for 3D Visual Grounding. In *Conference on Robot Learning*, 1046–1056. PMLR.

Shah, A.; Kamath, P.; Li, S.; Craven, P.; Landers, K.; Oden, K.; and Shah, J. 2023a. Supervised Bayesian Specification Inference from Demonstrations. *The International Journal of Robotics Research*, 42(14): 1245–1264.

Shah, A.; Li, S.; and Shah, J. 2020. Planning with Uncertain Specifications (PUnS). *IEEE Robotics and Automation Letters*, 5(2): 3414–3421.

Shah, D.; Osiński, B.; Levine, S.; et al. 2023b. LM-Nav: Robotic Navigation with Large Pre-Trained Models of Language, Vision, and Action. In *Conference on Robot Learning*, 492–504. PMLR.

Sutskever, I.; Vinyals, O.; and Le, Q. V. 2014. Sequence to Sequence Learning with Neural Networks. *Advances in Neural Information Processing Systems*, 27.

Tellex, S.; Gopalan, N.; Kress-Gazit, H.; and Matuszek, C. 2020. Robots That Use Language. *Annual Review of Control, Robotics, and Autonomous Systems*, 3: 25–55.

Vardi, M. Y. 1996. An Automata-Theoretic Approach to Linear Temporal Logic. *Logics for Concurrency*, 238–266.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L. u.; and Polosukhin, I. 2017. Attention Is All You Need. In Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Wang, C.; Ross, C.; Kuo, Y.-L.; Katz, B.; and Barbu, A. 2021. Learning a natural-language to LTL executable semantic parser for grounded robotics. In *Conference on Robot Learning*, 1706–1718. PMLR.

Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; Davison, J.; Shleifer, S.; von Platen, P.; Ma, C.; Jernite, Y.; Plu, J.; Xu, C.; Scao, T. L.; Gugger, S.; Drame, M.; Lhoest, Q.; and Rush, A. M. 2020. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 38–45.

Yang, Z.; Li, L.; Lin, K.; Wang, J.; Lin, C.-C.; Liu, Z.; and Wang, L. 2023. The Dawn of LMMs: Preliminary Explorations with GPT-4V(ision). *arXiv preprint arXiv:2309.17421*.

Zhang, J.; Huang, J.; Jin, S.; and Lu, S. 2024. Vision-Language Models for Vision Tasks: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–20.

Zheng, K.; Bayazit, D.; Mathew, R.; Pavlick, E.; and Tellex, S. 2021. Spatial Language Understanding for Object Search in Partially Observed City-scale Environments. In *IEEE International Conference on Robot & Human Interactive Communication*, 315–322.