2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

CoNVOI: Context-aware Navigation using Vision Language Models in Outdoor and Indoor Environments

Adarsh Jagan Sathyamoorthy[1], Kasun Weerakoon[1], Mohamed Elnoor[1], Anuj Zore[1], Brian Ichter[2], Fei Xia[2], Jie Tan[2], Wenhao Yu[2], and Dinesh Manocha[1]

Technical report, code, and video can be found at http://gamma.umd.edu/convoi/

***Abstract*— We present CoNVOI, a novel method for autonomous robot navigation in real-world indoor and outdoor environments using Vision Language Models (VLMs). We employ VLMs in two ways: first, we leverage their zero-shot image classification capability to identify the *context* or scenario (e.g., indoor corridor, outdoor terrain, crosswalk, etc) of the robot's surroundings, and formulate context-based navigation behaviors as simple text prompts (e.g. "*stay on the pavement*"). Second, we utilize their state-of-the-art semantic understanding and logical reasoning capabilities to compute a suitable trajectory given the identified context. To this end, we propose a novel multi-modal visual marking approach to annotate the obstacle-free regions in the RGB image used as input to the VLM with numbers, by correlating it with a local occupancy map of the environment. The marked numbers ground image locations in the real-world, direct the VLM's attention solely to navigable locations, and elucidate the spatial relationships between them and terrains depicted in the image to the VLM. Next, we query the VLM to select numbers on the marked image that satisfy the context-based behavior text prompt, and construct a reference path using the selected numbers. Finally, we propose a method to extrapolate the reference trajectory when the robot's environmental context has not changed to prevent unnecessary VLM queries. We use the reference trajectory to guide a motion planner, and demonstrate that it leads to human-like behaviors (e.g. not cutting through a group of people, using crosswalks, etc.) in various real-world indoor and outdoor scenarios. We perform several ablations and navigation comparisons and demonstrate that CoNVOI's trajectories are most similar to human teleoperated ground truth in terms of Fréchet distance (9.7-58.2% closer), lowest path errors (up to 88.13% lower), and up to 86.09% lower % of unacceptable paths.**

I. Introduction

Wheeled and legged robots have been used to navigate many kinds of challenging indoor and outdoor environments for applications such as delivery, surveillance, job-site monitoring in construction, disaster response, etc. The navigation behaviors required while performing these tasks can vary significantly in indoor and outdoor scenarios. For instance, in indoor settings, a robot must exhibit socially acceptable behaviors in narrow corridors, and in the presence of humans. In outdoor environments, the robot must avoid navigating on uneven or bumpy terrains and roads with oncoming traffic. It should favor stable, paved surfaces, identify locations to cross streets/roads, etc.

Humans are able to effectively navigate all these environments because we identify the surrounding scenario or *context* (e.g. indoor hallway, outdoor terrain, etc), and follow certain explicit (e.g. crossing roads at crosswalks) and implicit (e.g. keep to one side of a corridor) social rules, and preferences (e.g. walk on well-paved surfaces). Numerous research works in indoor and outdoor navigation have been proposed to exhibit these behaviors. For instance, works in indoor navigation have focused on socially-compliant navigation behaviors in avoiding and overtaking dynamic pedestrians [5], [6], [7], [8], groups of people [9], keeping to one side in corridors which helps avoid oncoming humans [10], [11], etc. In outdoor navigation, works have predominantly focused on estimating terrain traversability using semantic segmentation [12], [3], [13], self-supervised learning [14], [15], etc, and detecting complex outdoor obstacles [16].

However, prior model-based and learning-based approaches are tailored for specific perception or navigation tasks [17], limiting their applicability to other challenges or

Fig. 1: **[Top]**: The trajectories of a Spot robot crossing the road when using CoNVOI with GPT-4v [1] (in green), CoNVOI with Gemini [2] (in purple), teleoperated by a human (in red), GA-Nav [3], and DWA [4]. CoNVOI navigates the robot on the crosswalk by understanding the environmental context. **[Bottom]**: The trajectories when the Spot robot navigates to a goal beyond a blockade. While using CoNVOI, handling such scenarios can be added on the fly using a simple text prompt without any reformulation. The RGB images from the robot are shown with CoNVOI's multi-modal visual marking (numbers in yellow). GPT-4v/Gemini is queried with the marked image and a context-based text prompt in quotes, and it returns the green reference path that follows explicit, and implicit social rules, and human-like preferences during indoor and outdoor navigation.

This work was supported in part by ARO Grants W911NF2310046, W911NF2310352, and U.S. Army Cooperative Agreement W911NF2120076.

[1] Authors are with the University of Maryland, College Park.
[2] Authors are with Google DeepMind.

their generalization to different indoor and outdoor settings. Conversely, Large Language Models (LLMs) [18], [1], [19], [20], [2] that use only text inputs and subsequently large Vision Language Models (VLMs) that use both RGB images and text prompts as inputs [1], [2], [19], [21] have overcome this generalization limitation to a large extent. They have demonstrated impressive zero-shot classification [22], [23], semantic visual understanding [24], and logical reasoning capabilities [25] in various tasks involving embodied vision language navigation [26], [27], open vocabulary manipulation [28], [29], etc. These capabilities would be beneficial for perceiving and reasoning about complex indoor and outdoor environments during autonomous navigation.

However, there are a few key challenges in using large VLMs for navigating mobile robots. Due to their large model size and high memory and computational demands, these models cannot be run on robot-mounted edge processors. They typically require several seconds to process queries and the response time varies based on the size of the queries [30]. Therefore, current VLMs cannot be queried for safety-critical tasks such as realtime collision avoidance with dynamic obstacles. Additionally, queries should be succinct and account for the environment's context to elicit quick and accurate responses.

Main Contributions: We present a novel method to harness the state-of-the-art perception capabilities (generalization, semantic understanding, etc.) of VLMs, obtain a context-based reference path, and integrate it with a motion planning algorithm for real-time robot navigation in complex indoor and outdoor environments. The novel components of are work include:

- A novel context-based prompting method that involves querying a compact VLM [22] to determine the robot's environment context (e.g., indoor corridors, outdoor terrains, crosswalks). Next, our approach utilizes a predefined text prompt that describes a context-based navigation behavior and a marker-overlayed RGB image to a large VLM [1], [2], [20] to generate a reference trajectory for the robot. We demonstrate that using our context-based text prompts (that state explicit rules) helps generate reference trajectories that adhere to explicit and implicit social rules while also mimicking human preferences (e.g. walking on pavement, crossing at crosswalks, etc).
- A novel multi-modal visual marking method that enhances the RGB image inputs to a large VLM by overlaying visual markers/numbering on obstacle-free regions, aiding the VLM to focus on navigable regions, comprehending spatial relationships and underlying terrains across different regions within the image. Our method correlates an occupancy grid map obtained from lidar point clouds with the RGB image to determine the obstacle-free locations in images to add markers. We demonstrate that our visual prompting method results in more accurate responses (up to 88.13% lower path errors, and preventing unsafe paths up to 86.09% fewer times) to queries from the VLM when compared with visual markers that are overlayed on obstacles as well.
- A method to extrapolate the reference trajectory to future time steps when the robot's context has not changed. This allows our method to navigate without unnecessarily querying the VLM to generate new reference trajectories. We integrate our VLM-based perception with a planner to demonstrate smooth, uninterrupted, real-time navigation in several indoor and outdoor environments using a Turtlebot and a Spot robot without any domain specific fine-tuning. CoNVOI's trajectories match human teleoperated trajectories (upto 58.26% closer Fréchet distance) than existing baseline navigation methods.

II. Related Works

A. Indoor Navigation

Many recent works have focused on navigating in indoor settings in a socially acceptable manner [31], [32], [7], [33]. The objective of social navigation is to compute trajectories that are not only collision-free but also follow certain common norms [33] such as keeping to one side in a corridor, not interrupting groups of people by moving in-between them [9], etc. A robot must also maintain sufficient distance from humans even if they are closer to inanimate obstacles [31].

Many techniques that are based on reinforcement learning (RL) [34], and inverse reinforcement learning (IRL) [35] have been proposed to develop such behaviors. These methods train end-to-end models to avoid collisions, sudden rotatory maneuvers, and generate smooth trajectories. However, all these methods focus on modeling *some* aspects of social navigation and do not possess the general logical reasoning [17] that VLMs may possess to adapt to a new scenario. Our method achieves several social behaviors using our context-aware prompting without any training or reformulation.

B. Outdoor Navigation

For navigating outdoor environments, a robot must estimate the traversability of the terrains ahead of the robot, and compute trajectories on the most safe, and smooth terrain. To this end, there have been several works in pixel-wise semantic segmentation to classify a terrain into multiple predefined traversability classes (smooth, rough, bumpy, forbidden, etc.) [3], [36]. These works are typically trained in a supervised manner using human-annotated image datasets. There have also been unsupervised learning-based works that automatically label terrains by characterizing the robot-terrain interaction using other sensor data such as forces/torques [37], vibrations and differences in odometry [14], [15], acoustic data [38], vertical acceleration experienced [39], and stereo depth [40], [41], etc.

However, such models often do not generalize to new outdoor environments, and the associated planners may not be applicable to indoor settings. Our proposed approach using VLMs is applicable to both indoor and outdoor settings without requiring any environment-specific dataset or training.

C. Vision Language Models in Navigation

Over the past few years, LLMs [18] and VLMs [22], [1], [2], [19], [20], [21] have demonstrated impressive accuracy in tasks such as scene/semantic understanding [42], grounding objects in a scene based on language description, instruction following, code generation [43], etc that are useful for robot manipulation and navigation tasks. For instance, Shah et al. [27] utilized GPT-3 and CLIP to extract landmark descriptions from a text navigation instruction and ground them in images, respectively, to guide a robot's

navigation. Further, Shah et al. [44] demonstrated how the semantic understanding that large VLMs possess can help bias a robot's exploration to search for a user-defined object/location. Similar to [27], Chen et al. [45] utilize LLMs to decompose navigation instructions into a series of actionable tasks that are executed by an action-aware navigation policy.

Zhu et al. [46] propose a system to use VLMs to describe a scene, and an LLM to verify if the scene matches a user-defined target. Visual Language Maps [47] demonstrate fusing a VLM descriptions of landmarks with a 3D map that enables the robot to understand spatial references relative to landmark. OK-Robot [28] vision-language representations computed from an environmental scan by VLMs are stored as semantic memory [23], and is matched with a language query to move to a desired target object and pick it up in a zero-shot manner.

Existing research has primarily used VLMs to identify navigation targets (*where* to navigate) [26], [27] and parse natural language commands but has not focused on the *how* (lower-level behaviors such as walking on pavements, not disturbing interacting pedestrians) to navigate robots to their targets. Additionally, while many studies have been evaluated in realistic simulated environments, those tested in real-world settings [27], [28], [29] have not addressed the use of semantic and contextual understanding of the environment to guide the robot's navigation behaviors. CoNVOI's goal is to leverage VLMs to guide low-level navigation behaviors to be the most suitable for a given scenario.

III. Preliminaries

In this section, we introduce and define the symbols and key concepts used in our work.

A. Symbols, Definitions, and Assumptions

CoNVOI addresses the challenge "what navigation behaviors should a robot exhibit in its current local environmental context". This involves determining the locations within the robot's field of view to navigate toward to exhibit these behaviors, extending beyond simple obstacle avoidance.

For our formulation, we assume a robot equipped with an RGB camera and a 2D or 3D lidar rigidly attached to a wheeled or legged mobile robot base. The camera provides an RGB image I_t^{RGB} to view the environment and the lidar provides 2D laser scans at time instant t. Our formulation uses an $n \times n$ robot-centric occupancy grid map $\mathcal{O}_t$ obtained using the lidar's laser scan to represent the environmental obstacles. The grids of $\mathcal{O}_t$ are marked with 1's at grids that contain obstacles, and 0's otherwise. The obstacles are inflated by the robot's radius, and the robot is considered as a point at $(n/2, n/2)$, and the set of obstacles is denoted as obs_t. The pixels in I_t^{RGB} are correlated with the grids in $\mathcal{O}_t$ using a homography perspective projection.

The robot is controlled using linear and angular velocity commands (v, ω), respectively. We consider *three coordinate frames*: 1. robot - a frame attached to the robot's center of mass with X, Y, and Z axes pointing forward, leftward, and upward relative to the robot, 2. Occupancy map ($\mathcal{O}$) - a frame attached at the center of $\mathcal{O}_t$ with X, and Y axes pointing forward and leftward, and 3. RGB - a frame attached to the top-left corner of the RGB image obtained from the camera, with X, and Y axes pointing rightward (across columns), and downward (across rows). Quantities belong to the frame indicated by their superscripts (e.g. x^{rob} belongs to the robot frame). Transformation matrices between these frames are represented as T_{rob}^{odom}, $T_{\mathcal{O}}^{rob}$, $T_{RGB}^{\mathcal{O}}$. These matrices transform points in the subscript frame to the superscript frame. Finally, we use symbols i, j to denote indices. CoNVOI's overall architecture is shown in Fig. 2.

B. Motion Planner

Our formulation uses a separate motion planner integrated with the VLM to smoothly navigate the robot in real-time. In this section, we provide a brief overview of the planner's preliminary concepts.

We use the method proposed in [4] for motion planning. We represent the robot's actions as linear and angular velocity pairs (v, ω). The trajectory $traj^{rob}(v, w)$ that a (v, ω) pair leads to is calculated as a set of positions $\{(x_t^{rob}, y_t^{rob}), (x_{t+\Delta t}^{rob}, y_{t+\Delta t}^{rob}), ..., (x_{t+t_{hor}}^{rob}, y_{t+t_{hor}}^{rob})\}$ relative to the robot. It is also obtained relative to $\mathcal{O}_t$ and represented as $traj^{\mathcal{O}}(v, \omega)$. Here, Δt is a time increment, and t_{hor} is a time horizon until which the trajectory is extrapolated. At any time instant t, the optimal velocity pair (v^*, ω^*) to navigate the robot is calculated by minimizing the objective function below,

$$Q_1(v, \omega) = \sigma\big(\alpha_1.head(.) + \alpha_2.obs(.) + \alpha_3.vel(.)\big), \quad (1)$$

$\forall (v, \omega) \in V_r$. Here, V_r is the set of collision-free velocities ($\forall (v, \omega)$ s.t $traj^{\mathcal{O}}(v, \omega) \bigcap obs_t = \varnothing$) that are reachable from the robot's current velocity based on its acceleration constraints. In equation 1, $head(.)$, $obs(.)$, and $vel(.)$ are the cost functions [4] to quantify a velocity pair's ((v, ω) omitted on RHS for readability) heading towards the goal, distance to the closest obstacle in obs_t from $traj^{\mathcal{O}}(v, \omega)$, and the forward linear velocity of the robot, respectively. σ is a smoothing function and $\alpha_i, (i = 1, 2, 3)$ are adjustable weights.

IV. CoNVOI: Context-aware Navigation using VLMs in Outdoors and Indoors

CoNVOI uses two novel components: 1. a context-based prompting method to query a large VLM with an appropriate *behavior* text prompt, and 2. a multi-modal visual marking method to prepare the large VLM's input image. Using both components CoNVOI extracts a reference trajectory $\mathcal{T}_{ref}$ from the VLM. $\mathcal{T}_{ref}$ obeys the explicit and implicit rules associated with the robot's current context/scenario, and also encodes human-like preferences (e.g. opting to walk on pavement instead of grass) during navigation.

A. Context-based Behavior Prompt

VLMs utilize an image and text prompt inputs to output a text response. CoNVOI's first component is an approach to frame a context-based behavioral text prompt for the VLM. To this end, we formulate a set of contexts/scenarios $\mathcal{C} = \{c_1, c_2, ..., c_k\}$ that the robot would operate in (e.g. indoor corridors, in the presence of humans, outdoor terrains, crosswalk, etc.), and a set of robot behaviors for each context. The behaviors $\mathcal{B} = \{b_1, b_2, ..., b_k\}$ are specified as simple text phrases such as "keep close to the right wall" or "move on pavement", etc. Our formulation allows for easily adding new scenarios and associated behaviors to $\mathcal{C}$ and $\mathcal{B}$, respectively.

Next, we query a light-weight VLM (e.g. CLIP [22]) with I_t^{RGB} and text prompts T_i framed as *"This is*

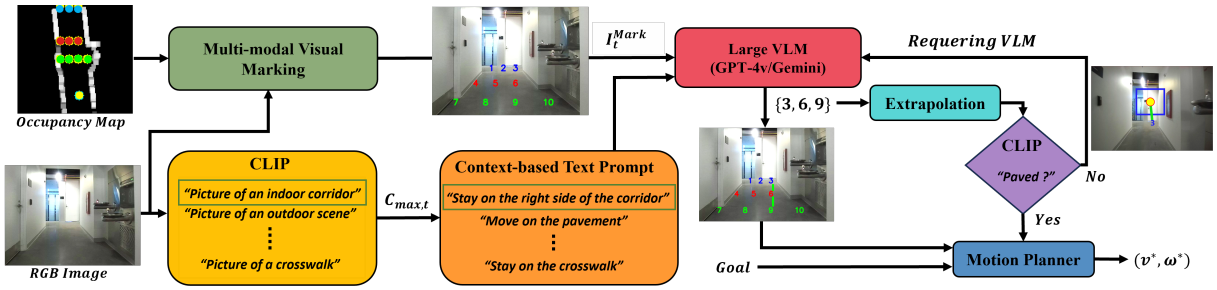

Fig. 2: CoNVOI's architecture utilizes CLIP to interpret the context of the robot's environment from an RGB image (I_t^{RGB}), identifying features such as indoor corridors, social scenarios with people, outdoor terrains, etc. Next, CoNVOI queries a large VLM with a context-based text prompt, and the RGB image marked with numbers (I_t^{Mark}) in the free space detected in an occupancy grid map to generate a reference path (in green) that adheres to explicit and implicit social rules (e.g., staying on pavement, using crosswalks). A dedicated motion planner then follows this path while avoiding obstacles. To prevent unnecessarily requerying the large VLM, we extrapolate this reference path linearly and check if the extrapolated point (in yellow) lies on a paved path (sidewalk, corridor, etc) to either use it for navigation, or requery the VLM again. Instead of using VLMs for direct robot control or for well-addressed tasks such as goal-reaching and obstacle avoidance, we leverage its context-understanding capabilities to achieve more intricate, zero-shot navigation behaviors using a separate planner.

a picture of c_i*"* $(\forall i)$, and obtain a set of probabilities $\{p_1, p_2, ..., p_k\}$ $\left(\sum_i p_i = 1\right)$, where $p_i = P(c_i | I_t^{RGB}, \{T_1, T_2, ..., T_k\})$ is the probability that the context/scenario in I_t^{RGB} corresponds to context c_i. Let $c_{max,t}$ correspond to the context with maximum probability at time t, and b_{max} be its associated robot behavior. Next, we frame a behavioral prompt $T_{b_{max}}$ in the format *"You are navigating a robot in this scenario. The numbers marked in the image denote regions where you can navigate the robot. Pick a list of numbers marked in the image such that the robot* _____*"*, and the blank is filled with the behavioral phrase in b_{max}.

B. Multi-modal Visual Marking

To prepare the image input for the VLM, we take into account the following considerations. Firstly, VLMs do not reliably infer spatial relationships, such as the distance of a point from the camera and the relative locations of various regions in the image (e.g., *point A is closer to the right side of the image*)[48]. Secondly, the VLM does not need to *focus* on obstacle regions since its task is solely to identify regions within the free space to identify *how* the robot should navigate. To this end, we propose to use the correlation between the robot's occupancy grid map $\mathcal{O}_t$ and the RGB image I_t^{RGB} to detect the free space in the robot's vicinity, and overlay markers (numbers) [25] on the corresponding free space in I_t^{RGB}. This approach combines the benefits of the spatial information encoded in occupancy grids with the semantic information in RGB images and results in a richer input to the VLM.

1) Detecting Free Regions to Add Markers: We consider l rows and m columns of grids each on $\mathcal{O}_t$ as candidate locations where the robot could traverse to exhibit context-aware behaviors. These grids are uniformly spaced by a distance d_{col} along the columns, and d_{row} along the rows. Let us consider the ij^{th} candidate grid $\mathbf{g}_{ij}, i \in \{1, 2, ..., l\}, j \in \{1, 2, ..., m\}$. Let L_{ij} be the line connecting $\mathbf{g}_{ij}$ and the robot's location $(n/2, n/2)$ in $\mathcal{O}_t$. $\mathbf{g}_{ij}$ is considered as a navigable candidate if,

$$L_{ij} \bigcap obs_t = \varnothing. \tag{2}$$

This ensures that locations obstructed by obstacles from the robot and its camera's view are not considered viable for context-aware navigation. We construct the set of candidate coordinates satisfying equation 2 as $X_t^{\mathcal{O}} = \{\mathbf{g}_{ij}\} \forall (i, j)$. The points in $X_t^{\mathcal{O}}$ are transformed into the RGB frame as,

$$(x_{ij}^{RGB}, y_{ij}^{RGB}) = T_{\mathcal{O}}^{RGB} \cdot \mathbf{g}_{ij} \quad \forall \mathbf{g}_{ij} \in X_t^{\mathcal{O}}. \tag{3}$$

We construct the set of candidate coordinates as $X_t^{RGB} = \{(x_{ij}^{RGB}, y_{ij}^{RGB})\} \forall (i, j)$. The points in X_t^{RGB} are ordered from left to right and row by row (see Fig. 2). Next, we mark numbers on all the points in X_t^{RGB} in ascending order on I_t^{RGB} and refer to this marked image as I_t^{Mark}.

C. Prompting the Large VLM

Finally, using $T_{b_{max}}$ and I_t^{Mark}, we prompt a large VLM [1], [2] to extract a reference path as,

$$\mathcal{M}_{ref} = \text{VLM}(I_t^{Mark}, T_{b_{max}}). \tag{4}$$

Here, $\mathcal{M}_{ref}$ is a list of numbers (markers) marked on I_t^{Mark}. Their corresponding set of grid locations relative to $\mathcal{O}_t$ are obtained by using the numbers in $\mathcal{M}_{ref}$ as indices since they are marked orderly in I_t^{Mark}. Hence, the reference path in the occupancy map frame is $\mathcal{T}_{ref}^{\mathcal{O}} = X_t^{\mathcal{O}}(\mathcal{M}_{ref})$. Their locations in the robot frame can be calculated as $\mathcal{T}_{ref}^{rob} = T_{\mathcal{O}}^{rob} \cdot \mathcal{T}_{ref}^{\mathcal{O}}$.

D. Reference Path Following

When the robot follows the reference path $\mathcal{T}_{ref}^{rob} = \{(x_{ref,1}^{rob}, y_{ref,1}^{rob}), (x_{ref,2}^{rob}, y_{ref,2}^{rob}), ..., (x_{ref,last}^{rob}, y_{ref,last}^{rob})\}$ $i = \{1, 2, ..., last\}$, it exhibits the most appropriate navigation behaviors for the context it encounters. These coordinates are repeatedly transformed into the robot's coordinate frame as the robot moves. To follow the reference path, we modify the motion planner [4] explained in section III-B as follows. Let θ_{goal} be the angle between the robot's current heading direction and the vector pointing to its goal direction. We first formulate a reference path following cost as,

$$\begin{aligned} ref(v, \omega) = |y_{ref,i}^{rob} - y_{t+t_{hor}}^{rob}|,\ y_{t+t_{hor}}^{rob} \in traj^{rob}(v, \omega), \\ \text{and } i \text{ s.t } x_{ref,i}^{rob} > 0 \text{ and} \\ x_{ref,i}^{rob} = min(x_{ref,1}^{rob}, x_{ref,2}^{rob}, ..., x_{ref,last}^{rob}), \end{aligned} \tag{5}$$

Equation 5 ensures that the robot picks (v, ω) trajectories that minimize the lateral distance between the robot and a

reference path point that is immediately ahead of the robot. A new objective function for planning as,

$$Q_2(v,\omega) = \begin{cases} Q_1(.) + \alpha_4 \cdot ref(.) & \text{if } |\theta_{goal}| \leq \theta_{fov}, \\ Q_1(.) & \text{Otherwise.} \end{cases} \quad (6)$$

Here, θ_{fov} is the field of view of the robot's camera. This condition helps to ensure that the robot does not stray away from the goal direction by following the VLM's reference paths.

E. Reference Path Extrapolation and Re-querying

Since large VLMs require several seconds to respond to queries (such as in equation 4), continuously querying to obtain new reference paths is computationally infeasible for real-time navigation. Therefore, CoNVOI reduces the frequency of querying the large VLM by extrapolating the current reference path, if the context has not changed.

Once the robot obtains a reference path and starts following it, CoNVOI repeatedly queries CLIP to check for context changes. If $c_{max,t} = c_{max,t'}(t' > t)$, and $\text{dist}((x^{rob}_{ref,last}, y^{rob}_{ref,last})) < d_{thresh}$ (since $(0,0)$ is the robot's location in its own frame, and d_{thresh} is a distance threshold), we linearly extrapolate the current reference path by fitting a line to the points in $\mathcal{T}^{rob}_{ref}$ and extending it further to a point $(x^{rob}_{ref,e}, y^{rob}_{ref,e})$ by distance d_{row}. Next, we check if the point $(x^{\mathcal{O}}_{ref,e}, y^{\mathcal{O}}_{ref,e})$ is obstacle-free (similar to section IV-B.1). If it is, we add the point to the RGB image at $(x^{RGB}_{ref,e}, y^{RGB}_{ref,e})$, and crop a $n_{pat} \times n_{pat}$ image patch I^{pat} (see Fig. 2) to query CLIP if the patch is an image of a paved (indoors or sidewalks) or unpaved (grass, gravel, asphalt) surface. If $CLIP(I^{pat})$ is paved, $(x^{rob}_{ref,e}, y^{rob}_{ref,e})$ is used as the next waypoint for the planner to follow. Otherwise, the large VLM is queried for the next reference path.

If the context has changed, CoNVOI queries the large VLM once $\text{dist}((x^{rob}_{ref,last}, y^{rob}_{ref,last})) < d_{thresh}$. d_{thresh} is chosen such that $d_{thresh} \approx v_{max} \cdot t_{query}$, where v_{max} is the maximum linear velocity of the robot and t_{query} is the average time for a large VLM to respond to a query.

V. Results and Evaluations

In this section, we explain CoNVOI's implementation on real robots, and analyze and evaluate its performance through comparisons and ablations.

A. Implementation

CoNVOI is implemented on a Turtlebot 2 for indoor evaluation, and on a Boston Dynamics Spot robot for outdoor evaluation. Both robots are equipped with a Velodyne VLP16 lidar, and a Zed 2i camera providing outputs at $\sim$ 30Hz. CoNVOI is executed on a laptop with an Intel i7 11th generation CPU and an Nvidia RTX 3060 GPU. The visual markings are added at $\sim$ 20Hz, and for context detection, CLIP is executed locally on the laptop at $\sim$ 15Hz. For the reference path queries, we use the online API for GPT-4v [1], and Gemini [2]. We use the following values for the parameters in our formulation: $n = 200, \alpha_1 = 10, \alpha_2 = 7, \alpha_3 = 1, \alpha_4 = 7.5, l = 2$ for outdoors, 3 for indoors, $m = 6, d_{row} = 2.5m, d_{col} = 2m, k = 5, v_{max} = 0.3$m/s, $t_{query} = 5s, d_{thresh} = 4$meters, $n_{pat} = 200$. l is kept higher for outdoor environments as they are more expansive and obtaining a longer reference trajectory is beneficial.

B. Comparison Methods

We compare CoNVOI with several existing methods for indoor and outdoor navigation. They are:

- **DWA** [4]: The dynamic window approach, a baseline motion planner that performs simple collision avoidance and goal-reaching behaviors.
- **GA-Nav** [3]: A planner that uses semantic segmentation-based outdoor terrain classification to assign traversability costs to various terrains and computes trajectories on smooth (low-cost) terrains.
- **PSP-Net** [49]: A planner similar to GA-Nav but employing PSP-Net for semantic segmentation.
- **Frozone** [7]: A social-compliant indoor navigation method that computes trajectories that are least obtrusive to surrounding pedestrians. We enhance Frozone's formulation by incorporating a method to detect corridors from the robot's local occupancy grid. Additionally, we assign higher costs to regions closer to the left wall, which encourages the robot to stay closer to the right side.

In addition to baseline methods, we conduct thorough ablation studies by analyzing our approach's reference path selection performance by removing its multi-modal visual marking, and context-based prompting components.

C. Evaluation Metrics

We use the following quantitative metrics to evaluate the navigation performance of all the methods in real-world scenarios.

- **Fréchet Distance w.r.t. Human Teleoperation**: Measures the Fréchet distance [50] (a measure of similarity between two curves) between an average of several human teleoperated robot trajectories versus a comparison method's trajectory.
- **Normalized Trajectory Length**: The robot's trajectory length normalized over the straight-line distance to the goal.
- **Mean Velocity**: The robot's linear velocity (that indicates its progress towards the goal) averaged over all the trials.

We use the following metrics to evaluate CoNVOI's reference path selecting capabilities by comparing it with a human chosen ground truth points in the marked image.

- **Reference Path Error**: Measures the distance between the mean points in the ground truth path and the VLM's output reference path.
- **Cosine Similarity**: Quantifies the alignment between the ground truth and the VLM's reference path in terms of orientation.
- **Number of unacceptable paths**: Counts the number of times a reference path was on an unacceptable terrain/location (e.g. on an obstacle, asphalt road that could have traffic, etc).
- **Inference Time**: The time between sending a query to the large VLM and it returning a reference path. This includes network latencies.

D. Navigation Testing Scenarios

We compare all the methods across four types of scenarios: 1. Indoor corridors, 2. Indoor scenarios with people, 3. Outdoor scenarios with multiple terrains, 4. Outdoor crosswalks.

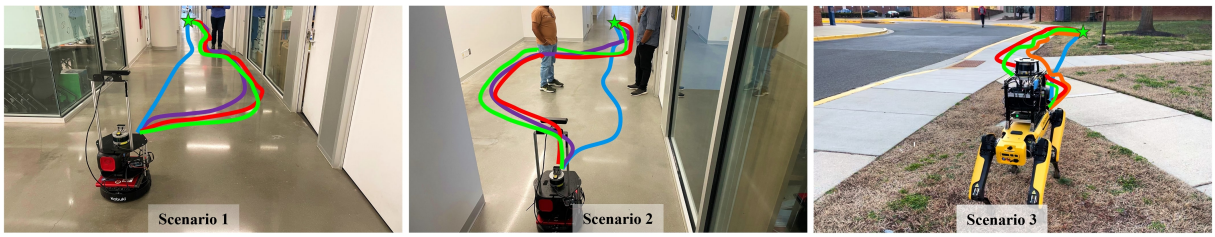

Fig. 3: Robot trajectories when navigating in different complex indoor and outdoor environments using various methods: CoNVOI (in green), teleoperated by a human (in red), DWA [4] (in blue), Frozone [7] (in violet), GA-Nav [3] (in orange). CoNVOI exhibits social-compliant behaviors such as not moving in-between humans even if there is sufficient space, similar to Frozone, a method formulated for indoor social navigation. CoNVOI's behaviors also match GA-Nav, a semantic segmentation-based navigation approach that prefers to navigate on smooth, well-paved outdoor terrains. CoNVOI achieves these behaviors in a zero-shot manner, and does not require domain-specific fine-tuning. CoNVOI's trajectories also closely match human-teleoperated ground truth paths (in red).

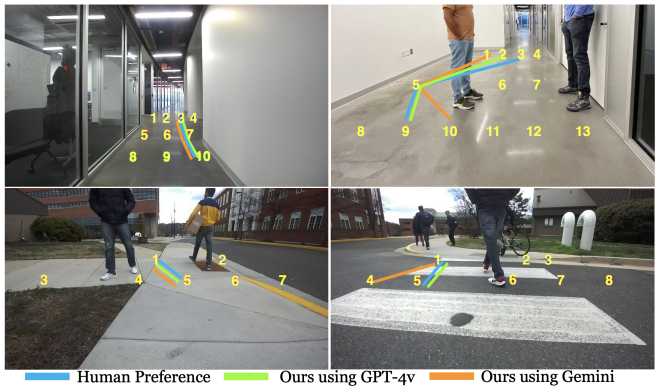

Fig. 4: Qualitative comparison of the reference trajectory generated by GPT-4v [1] (in green), and Gemini [2] (in orange) compared with human provided ground truth (in blue) by connecting the marked numbers (in yellow). We observe that the reference paths generated by the VLMs is comparable to that of the human-preferred path in many complex indoor and outdoor environments.

Additionally, we illustrate the addition of behaviors for a new scenario and showcase the resulting robot behavior.

Scenario	Method	Reference Path Error ↓	Cosine Similarity ↑	% Unacceptable Paths ↓	Inference Time ↓
Scen. 1	Gemini w/o MMVM w. CbP	1.442	0.926	0.857	5.24
	GPT4 w/o MMVM w. CbP	1.576	0.913	0.913	4.86
	Gemini w. MMVM w/o CbP	0.315	0.995	**0.0**	4.78
	GPT4 w. MMVM w/o CbP	0.335	0.993	**0.0**	4.10
	Gemini w. CoNVOI	0.202	**0.996**	**0.0**	3.76
	GPT4 w. CoNVOI	**0.187**	0.995	**0.0**	**3.48**
Scen. 2	Gemini w/o MMVM w. CbP	1.425	0.914	0.786	5.11
	GPT4 w/o MMVM w. CbP	2.325	0.802	0.798	4.69
	Gemini w. MMVM w/o CbP	0.484	0.989	0.278	4.91
	GPT4 w. MMVM w/o CbP	0.589	0.988	0.223	4.46
	Gemini w. CoNVOI	**0.444**	0.981	**0.111**	3.55
	GPT4 w. CoNVOI	0.456	**0.991**	0.167	**3.28**
Scen. 3	Gemini w/o MMVM w. CbP	0.891	0.973	0.433	6.35
	GPT4 w/o MMVM w. CbP	0.865	0.977	0.366	5.60
	Gemini w. MMVM w/o CbP	0.817	**0.978**	0.291	6.28
	GPT4 w. MMVM w/o CbP	0.872	0.971	0.125	5.46
	Gemini w. CoNVOI	0.979	0.970	0.125	4.88
	GPT4 w. CoNVOI	**0.785**	0.975	**0.083**	**4.15**
Scen. 4	Gemini w/o MMVM w. CbP	1.210	0.957	0.285	6.36
	GPT4 w/o MMVM w. CbP	1.205	0.966	0.190	4.89
	Gemini w. MMVM w/o CbP	1.229	0.938	0.223	5.98
	GPT4 w. MMVM w/o CbP	1.159	0.962	0.166	4.95
	Gemini w. CoNVOI	1.205	0.977	**0.055**	5.26
	GPT4 w. CoNVOI	**0.819**	**0.978**	**0.055**	**4.83**

TABLE I: Ablation studies comparing the impact of removing multi-modal visual marking (MMVM) and context-based prompting (CbP) from our formulation on generating reference paths. We also compare the use of two state-of-the-art large VLMs: GPT-4v [1] and Gemini [2]. Our results show that MMVM and CbP significantly improve the VLM's performance in generating human-like reference paths.

E. Analysis and Comparisons

The results of our qualitative and quantitative comparisons are shown in Fig. 3, and Table II, respectively. The results of our ablation studies are shown in Table I.

Ablation Study: We analyze the effects of our multi-modal visual marking (MMVM), and context-based prompting on generating the reference path by removing these components individually from CoNVOI. Next, we compare the generated reference path with human-provided reference paths (see Fig. 4). First, removing MMVM, the marked image I_t^{Mark} would contain numbers on obstacles (walls, people, etc.). Therefore, while choosing the numbers for the reference path, the VLM must also ensure that they do not lie on obstacles (thus being responsible for avoiding them). However, we observe that this results in high reference path errors, and percentage of unacceptable paths as a subset of the chosen points often lie on obstacles in indoor scenarios (1 and 2). Additionally, the cosine similarity when compared with human GT is low, indicating that the paths are not generally in the direction preferred by a human. In outdoor environments, removing MMVM does not affect the metrics as adversely, since these scenarios are expansive and less constrained by humans. However, the percentage of unacceptable paths is higher as the returned reference paths some times intersect with walking pedestrian obstacles. The inference rate is also increased since the VLM must now pay attention to more regions before choosing points for the reference path. We observe that current VLMs cannot be used for zero-shot obstacle avoidance reliably.

In the second ablation study, we removed context-based prompting (CbP) and used a single detailed prompt describing the rules for navigating all associated scenarios. We observe that this leads to a performance comparable to CoNVOI in terms of the path error and cosine similarity. However, there is a considerable increase in the VLM's inference time. This could be due to the higher number of tokens the VLM must process given a detailed text prompt. Additionally, we observe a significant increase in the percentage of unacceptable paths in scenarios 2 and 4, where certain intricate behaviors (avoid moving in-between people, navigate on crosswalk) are expected. Such behaviors may be harder to interpret accurately when included with behaviors for other scenarios in the detailed prompt. We observe the lowest path errors, percentage of unacceptable paths, inference time, and highest cosine similarity for CoNVOI (with MMVM, CbP) in all cases. Since using CoNVOI with GPT-4v [1] has the least inference rate along with the best reference paths, we use it for comparison with other

Scenario	Method	Fréchet Dist. ↓	Norm. Traj. Length ≈ 1	Mean Velocity ↑
Scen. 1	DWA[4]	0.864	1.082	0.395
	Frozone[7]	0.577	1.239	0.392
	CoNVOI (ours)	0.521	1.207	0.289
Scen. 2	DWA[4]	1.087	1.105	0.390
	Frozone[7]	0.866	1.187	0.388
	CoNVOI (ours)	0.709	1.194	0.276
Scen. 3	DWA[4]	2.475	1.067	0.485
	PSP-Net [49]	2.188	1.121	0.458
	GA-Nav[3]	1.957	1.160	0.473
	CoNVOI (ours)	1.083	1.213	0.238
Scen. 4	DWA[4]	2.095	1.091	0.491
	PSP-Net [49]	2.068	1.089	0.455
	GA-Nav[3]	2.157	1.112	0.466
	CoNVOI (ours)	0.863	1.237	0.243

TABLE II: Performance comparisons when a robot uses CoNVOI DWA motion planner [4] that performs only obstacle avoidance and goal-reaching, Frozone [7], a socially-compliant indoor navigation method, and PSP-Net [49] and GA-Nav [3], two semantic segmentation-based outdoor navigation methods. CoNVOI generates trajectories that are most similar to human teleoperated trajectories (lowest Fréchet distance). Qualitatively, CoNVOI performs similar to social and outdoor navigation methods in a zero-shot manner and is applicable to both indoor and outdoor environments.

navigation methods in Table II.

Comparison with Navigation Methods: We compare all the navigation methods relative to human teleoperation, which is considered the *best* trajectory the robot could take in its current context. We observe that CoNVOI's zero-shot path resemble the human teleoperated path the closest (lowest Fréchet distance in Table II) in all the scenarios (see Fig. 1, and 3).

We observe that CoNVOI typically leads to a higher normalized trajectory length due to higher deviations from the goal to follow the context-based navigation behavior, similar to the human teleoperated path. Additionally, CoNVOI has a low mean velocity due to its limiting maximum velocity ($v_{max} = 0.3$ m/s) set to ensure uninterrupted navigation, due to the high inference time current VLMs require. However, we highlight that CoNVOI demonstrates all these context-based behaviors in a zero-shot manner, utilizing simple text and visual prompts. Unlike existing model-based and deep learning-based methods, it has demonstrated wide applicability across various indoor and outdoor environments.

Handling Novel Scenarios: We also highlight that new types of scenarios can be easily added to CoNVOI without any reformulation. We add a scenario with a detour sign and instruct CoNVOI to deviate the robot in the direction of the detour sign's arrow. We observe that CoNVOI immediately adapts to the new scenario, and navigates to its goal behind the sign (Fig. 1 [bottom]). This signifies how CoNVOI can use VLMs' complex visual understanding and reasoning capabilities for navigation in the real-world.

Extrapolation Benefits: We observe that CoNVOI's extrapolation helps avoid unnecessarily requerying the large VLM in environments where the context does not change until the goal (e.g. in corridors, straight pavements). On average, extrapolation helps reduce the requerying by $\sim 50\%$ in such environments. In dynamic scenarios with walking humans, in the presence of several terrains (extrapolated point could lie in unpaved terrain), the VLM is requeried to guide the robot. For more results, we point the reader to our technical report.

VLM Hallucinations: Since during navigation the large VLM is typically fed with a sequence of *similar*-looking images and text prompts, we observe that it often starts to hallucinate and return reference path numbers in a certain pattern (e.g. [1, 2, 3, 4, 5], or [1, 2, 1, 2]) without considering the scenario. This occurs when the VLM uses its recent memory to perform the current task. To alleviate hallucinations, we add a phrase to the text prompt to instruct the VLM that it must consider the current query as a new task and disregard past inputs and outputs.

Wavy Trajectories: We observe that when there are several correct paths that satisfy the behavior specified in the context-based text prompt, the VLM may return any one of them. This could sometimes result in a wavy trajectory (as shown in Fig. 1 on the crosswalk, while passing the barricade). This is primarily due to the spacing parameter d_{col} between the numbers marked in the image, which affects the spatial resolution of the reference paths. However, decreasing d_{col} leads to more numbers annotated on the image, which leads to higher inference rates to process, similar to removing MMVM in Table I. Such limitations could be alleviated with faster large VLMs in the future.

VI. Conclusions, Limitations, and Future Work

We present CoNVOI, a novel method to use compact and large VLMs for indoor and outdoor navigation without any additional training or fine-tuning. CoNVOI utilizes a novel multi-modal visual marking scheme, and a context-based prompting method to achieve complex behaviors (e.g. avoiding moving in-between groups of humans, interpreting signs, etc) in various indoor and outdoor scenarios. CoNVOI also uses an extrapolation method to avoid unnecessary requerying of the large VLM that reduces the queries by up to 50% in certain scenarios.

Our method has a few limitations. Since CoNVOI uses a large VLM that is hosted remotely, its processing time and network latency affects the robot's navigation and its maximum velocity. In outdoor environments, since network speeds also depend on weather conditions, we observe some rare instances where the robot does not receive a reference path, and uses the baseline planner to navigate. Additionally, this limits VLMs' use in highly dynamic environments with crowds. However, such limitations can be overcome as large VLMs become faster, and locally executable. Our formulation also assumes that each contexts does not overlap with other contexts, and uses a single rule per context. We also observe that outputs could be sensitive to vague terms in the text prompt (e.g. "*cross the street like a human*" instead of specifying "*stay on the crosswalk*"). Our current formulation does not use any global information like a top-down view of the environment, which would be needed to assess factors like where to cross the road, when the crosswalk is not visible from the robot's camera. We plan to address these limitations in our future work, and extend our formulation to long-range navigation.

References

[1] OpenAI, "Gpt-4 technical report," 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:257532815

[2] Gemini Team, "Gemini: A Family of Highly Capable Multimodal Models," *arXiv e-prints*, p. arXiv:2312.11805, Dec. 2023.

[3] T. Guan, D. Kothandaraman, R. Chandra, A. J. Sathyamoorthy, K. Weerakoon, and D. Manocha, "Ga-nav: Efficient terrain segmentation for robot navigation in unstructured outdoor environments," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 8138–8145, 2022.

[4] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robotics Automation Magazine*, vol. 4, no. 1, pp. 23–33, March 1997.

[5] D. Helbing and P. Molnár, "Social force model for pedestrian dynamics," vol. 51, no. 5, pp. 4282–4286, May 1995.

[6] Y. F. Chen, M. Everett, M. Liu, and J. How, "Socially aware motion planning with deep reinforcement learning," 09 2017, pp. 1343–1350.

[7] A. J. Sathyamoorthy, U. Patel, T. Guan, and D. Manocha, "Frozone: Freezing-free, pedestrian-friendly navigation in human crowds," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4352–4359, 2020.

[8] D. Mehta, G. Ferrer, and E. Olson, "Autonomous navigation in dynamic social environments using multi-policy decision making," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 1190–1197.

[9] A. J. Sathyamoorthy, U. Patel, M. Paul, N. K. S. Kumar, Y. Savle, and D. Manocha, "Comet: Modeling group cohesion for socially compliant robot navigation in crowded scenes," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 1008–1015, 2022.

[10] D. V. Lu and W. D. Smart, "Towards more efficient navigation for robots and humans," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 1707–1713.

[11] J. Guzzi, A. Giusti, L. M. Gambardella, G. Theraulaz, and G. A. Di Caro, "Human-friendly robot navigation in dynamic environments," in *2013 IEEE International Conference on Robotics and Automation*, 2013, pp. 423–430.

[12] G. Humblot-Renaux, L. Marchegiani, T. B. Moeslund, and R. Gade, "Navigation-oriented scene understanding for robotic autonomy: learning to segment driveability in egocentric images," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2913–2920, 2022.

[13] A. Shaban, X. Meng, J. Lee, B. Boots, and D. Fox, "Semantic terrain classification for off-road autonomous driving," in *Conference on Robot Learning*. PMLR, 2022, pp. 619–629.

[14] A. J. Sathyamoorthy, K. Weerakoon, T. Guan, J. Liang, and D. Manocha, "Terrapn: Unstructured terrain navigation using online self-supervised learning," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 7197–7204.

[15] A. Polevoy, C. Knuth, K. M. Popek, and K. D. Katyal, "Complex Terrain Navigation via Model Error Prediction," *arXiv e-prints*, p. arXiv:2111.09768, Nov. 2021.

[16] A. J. Sathyamoorthy, K. Weerakoon, T. Guan, M. Russell, D. Conover, J. Pusey, and D. Manocha, "Vern: Vegetation-aware robot navigation in dense unstructured outdoor environments," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 11 233–11 240.

[17] R. Firoozi, J. Tucker, S. Tian, A. Majumdar, J. Sun, W. Liu, Y. Zhu, S. Song, A. Kapoor, K. Hausman, B. Ichter, D. Driess, J. Wu, C. Lu, and M. Schwager, "Foundation Models in Robotics: Applications, Challenges, and the Future," *arXiv e-prints*, p. arXiv:2312.07843, Dec. 2023.

[18] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, and et al., "Language Models are Few-Shot Learners," *arXiv e-prints*, p. arXiv:2005.14165, May 2020.

[19] M. AI, "Llama 2: Open Foundation and Fine-Tuned Chat Models," *arXiv e-prints*, p. arXiv:2307.09288, July 2023.

[20] H. Liu, C. Li, Y. Li, and Y. J. Lee, "Improved Baselines with Visual Instruction Tuning," *arXiv e-prints*, p. arXiv:2310.03744, Oct. 2023.

[21] H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Visual Instruction Tuning," *arXiv e-prints*, p. arXiv:2304.08485, Apr. 2023.

[22] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning Transferable Visual Models From Natural Language Supervision," *arXiv e-prints*, p. arXiv:2103.00020, Feb. 2021.

[23] N. M. M. Shafiullah, C. Paxton, L. Pinto, S. Chintala, and A. Szlam, "Clip-fields: Weakly supervised semantic fields for robotic memory," *arXiv preprint arXiv: Arxiv-2210.05663*, 2022.

[24] H. Ha and S. Song, "Semantic abstraction: Open-world 3d scene understanding from 2d vision-language models," in *6th Annual Conference on Robot Learning*, 2022.

[25] S. Nasiriany, F. Xia, W. Yu, T. Xiao, and et al., "PIVOT: Iterative Visual Prompting Elicits Actionable Knowledge for VLMs," *arXiv e-prints*, p. arXiv:2402.07872, Feb. 2024.

[26] V. S. Dorbala, G. A. Sigurdsson, J. Thomason, R. Piramuthu, and G. S. Sukhatme, "CLIP-nav: Using CLIP for zero-shot vision-and-language navigation," in *Workshop on Language and Robotics at CoRL 2022*, 2022. [Online]. Available: https://openreview.net/forum?id=yEdWwYnQpD

[27] D. Shah, B. Osinski, B. Ichter, and S. Levine, "LM-nav: Robotic navigation with large pre-trained models of language, vision, and action," in *6th Annual Conference on Robot Learning*, 2022. [Online]. Available: https://openreview.net/forum?id=UW5A3SweAH

[28] P. Liu, Y. Orru, C. Paxton, N. M. M. Shafiullah, and L. Pinto, "Ok-robot: What really matters in integrating open-knowledge models for robotics," *arXiv preprint arXiv:2401.12202*, 2024.

[29] M. Ahn, A. Brohan, N. Brown, Y. Chebotar, O. Cortes, B. David, and et al., "Do as i can and not as i say: Grounding language in robotic affordances," in *arXiv preprint arXiv:2204.01691*, 2022.

[30] Y. Chang, X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang, Y. Wang, *et al.*, "A survey on evaluation of large language models," *ACM Transactions on Intelligent Systems and Technology*, 2023.

[31] Y. F. Chen, M. Everett, M. Liu, and J. How, "Socially aware motion planning with deep reinforcement learning," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 09 2017, pp. 1343–1350.

[32] L. Tai, J. Zhang, M. Liu, and W. Burgard, "Socially compliant navigation through raw depth inputs with generative adversarial imitation learning," in *ICRA*, May 2018, pp. 1111–1117.

[33] T. Kruse, A. K. Pandey, R. Alami, and A. Kirsch, "Human-aware robot navigation: A survey," *Robotics Auton. Syst.*, vol. 61, pp. 1726–1743, 2013.

[34] P. Long, T. Fan, X. Liao, W. Liu, H. Zhang, and J. Pan, "Towards Optimally Decentralized Multi-Robot Collision Avoidance via Deep Reinforcement Learning," *arXiv e-prints*, p. arXiv:1709.10082, Sep 2017.

[35] M. Pfeiffer, U. Schwesinger, H. Sommer, E. Galceran, and R. Siegwart, "Predicting actions to act predictably: Cooperative partial motion planning with maximum entropy models," 10 2016, pp. 2096–2101.

[36] F. Schilling, X. Chen, J. Folkesson, and P. Jensfelt, "Geometric and visual terrain classification for autonomous mobile navigation," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 2678–2684.

[37] L. Wellhausen, A. Dosovitskiy, R. Ranftl, K. Walas, C. Cadena, and M. Hutter, "Where should i walk? predicting terrain properties from images via self-supervised learning," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1509–1516, 2019.

[38] K. Zhao, M. Dong, and L. Gu, "A new terrain classification framework using proprioceptive sensors for mobile robots," *Mathematical Problems in Engineering*, vol. 2017, pp. 1–14, 09 2017.

[39] M. A. Bekhti, Y. Kobayashi, and K. Matsumura, "Terrain traversability analysis using multi-sensor data correlation by a mobile robot," in *2014 IEEE/SICE International Symposium on System Integration*, 2014, pp. 615–620.

[40] R. Hadsell, P. Sermanet, J. Ben, A. Erkan, M. Scoffier, K. Kavukcuoglu, U. Muller, and Y. LeCun, "Learning long-range vision for autonomous off-road driving," *Journal of Field Robotics*, vol. 26, no. 2, pp. 120–144, 2009, copyright: Copyright 2009 Elsevier B.V., All rights reserved.

[41] M. Procopio, J. Mulligan, and G. Grudic, "Learning terrain segmentation with classifier ensembles for autonomous robot navigation in unstructured environments," *Journal of Field Robotics*, vol. 26, pp. 145 – 175, 02 2009.

[42] D. Rivkin, G. Dudek, N. Kakodkar, D. Meger, O. Limoyo, M. Jenkin, X. Liu, and F. Hogan, "Ansel photobot: A robot event photographer with semantic intelligence," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 8262–8268.

[43] J. Liang, W. Huang, F. Xia, P. Xu, K. Hausman, B. Ichter, P. Florence, and A. Zeng, "Code as policies: Language model programs for embodied control," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 9493–9500.

[44] D. Shah, M. R. Equi, B. Osiński, F. Xia, brian ichter, and S. Levine, "Navigation with large language models: Semantic guesswork as a heuristic for planning," in *7th Annual Conference on Robot Learning*, 2023. [Online]. Available: https://openreview.net/forum?id=PsV65r0itpo

[45] P. Chen, X. Sun, H. Zhi, R. Zeng, T. H. Li, G. Liu, M. Tan, and C. Gan, "A^2Nav: Action-Aware Zero-Shot Robot Navigation by Exploiting Vision-and-Language Ability of Foundation Models," *arXiv e-prints*, p. arXiv:2308.07997, Aug. 2023.

[46] P. Zhu, L. El Hafi, and T. Taniguchi, "Visual-language decision system through integration of foundation models for service robot navigation," in *2024 IEEE/SICE International Symposium on System Integration (SII)*. IEEE, 2024, pp. 1288–1295.

[47] C. Huang, O. Mees, A. Zeng, and W. Burgard, "Visual language maps for robot navigation," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 10 608–10 615.

[48] F. Liu, G. Emerson, and N. Collier, "Visual spatial reasoning," *Transactions of the Association for Computational Linguistics*, vol. 11, pp. 635–651, 2023.

[49] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2881–2890.

[50] H. Alt and M. Godau, "Computing the fréchet distance between two polygonal curves," *International Journal of Computational Geometry & Applications*, vol. 5, no. 01n02, pp. 75–91, 1995.