Green for Go, Red for No: Visual Grounding via Semantic Segmentation for VLA Navigation Policies

Adrian Szvoren
Department of Computer Science
University College London
adrian.szvoren.23@ucl.ac.uk

Dimitrios Kanoulas
Department of Computer Science
University College London
d.kanoulas@ucl.ac.uk

Nilufer Tuptuk
Department of Security
and Crime Science
University College London
n.tuptuk@ucl.ac.uk

***Abstract*—Vision-language-action (VLA) models enable robot navigation from natural language and visual goals, but remain susceptible to perceptual distractions and ambiguous scene interpretations. This paper presents the first empirical evaluation of visual grounding for VLA navigation policies. We propose a real-time segmentation-based grounding method that highlights traversable areas in green and non-traversable areas in red using SegFormer. Two variants are evaluated: observation-only segmentation and joint observation-goal augmentation. Using OmniVLA on the Grand Tour dataset, we show that visual grounding reduces the mean waypoint error by $27-44\%$ at the farthest waypoint, depending on the instruction length. The benefits are greater for long instructions than for short instructions, and grounding provides little improvement for image goals. Normalized error analysis indicates that grounding primarily acts as a trajectory length regularizer, reducing the predicted path length by 30% without improving per-unit-distance reasoning. Our results indicate that visual grounding offers a simple, computationally inexpensive method to improve VLA navigation without model retraining, although it cannot compensate for missing training signals in out-of-distribution instructions.**

I. Introduction

Vision-language-action (VLA) models have emerged as a promising development for robot navigation, enabling agents to interpret visual observations and natural language instructions to generate actions or navigation trajectories. Unlike traditional navigation systems that often rely on pre-built maps or explicit waypoint sequences, VLA models leverage large-scale pretraining on diverse datasets to generalize to novel environments and instructions without task-specific fine-tuning. Recent advances have demonstrated impressive capabilities in following free-form language goals and adapting to unseen spatial layouts [9, 3, 4]. However, despite these successes, VLA policies remain susceptible to perceptual distractions, ambiguous scene interpretations, and the inherent difficulty of translating high-dimensional visual input into precise navigational actions.

One promising direction for improving VLA robustness is visual grounding. Visual grounding techniques augment or preprocess visual input to make spatial relationships and traversability cues more explicit to the underlying model. In the context of vision-language models, methods have shown that overlaying images with segmentation masks, numerical labels, or action visualizations can significantly enhance reasoning and visual question answering accuracy [19, 13]. This leads to the following research question: can visual grounding similarly benefit navigation VLA policies, where the agent must continuously reason about traversable space, obstacles, and goal-directed movement in real-time?

To empirically evaluate this, we instantiate two variants of our grounding pipeline: one that segments only the current visual observation, and a second that additionally augments the goal modality (appending explicit traversability instructions to language goals or segmenting image goals). We focus on three key constraints: real-time inference, generalization to unseen environments, and multimodal goal support.

The key contributions of this paper are: (1) to our knowledge, the first empirical evaluation of visual grounding for a VLA navigation policy; (2) a real-time segmentation-based grounding method using SegFormer [18]; (3) two grounding variants for ablative analysis; (4) quantitative results showing that grounding reduces far-waypoint error by 27–44% depending on the instruction length, with greater benefits for long than short instructions, and minimal improvement for image goals; and (5) identification of the "stop" instruction failure mode in current VLA models.

The remainder of this paper is organized as follows. Section II reviews related work on VLM and VLA navigation and visual grounding. Section III formalizes the trajectory generation problem. Section IV describes our segmentation-based grounding method and implementation choices. Section V presents the experimental setup, evaluation metrics, and quantitative results. Section VI discusses limitations and Section VII concludes with directions for future work.

II. Related work

A. Vision-Language and Vision-Language-Action Models for Navigation

Large language models have seen increasing use, including for robot navigation and trajectory generation; because most robotic systems are equipped with cameras, vision-language models have emerged as a particularly promising approach

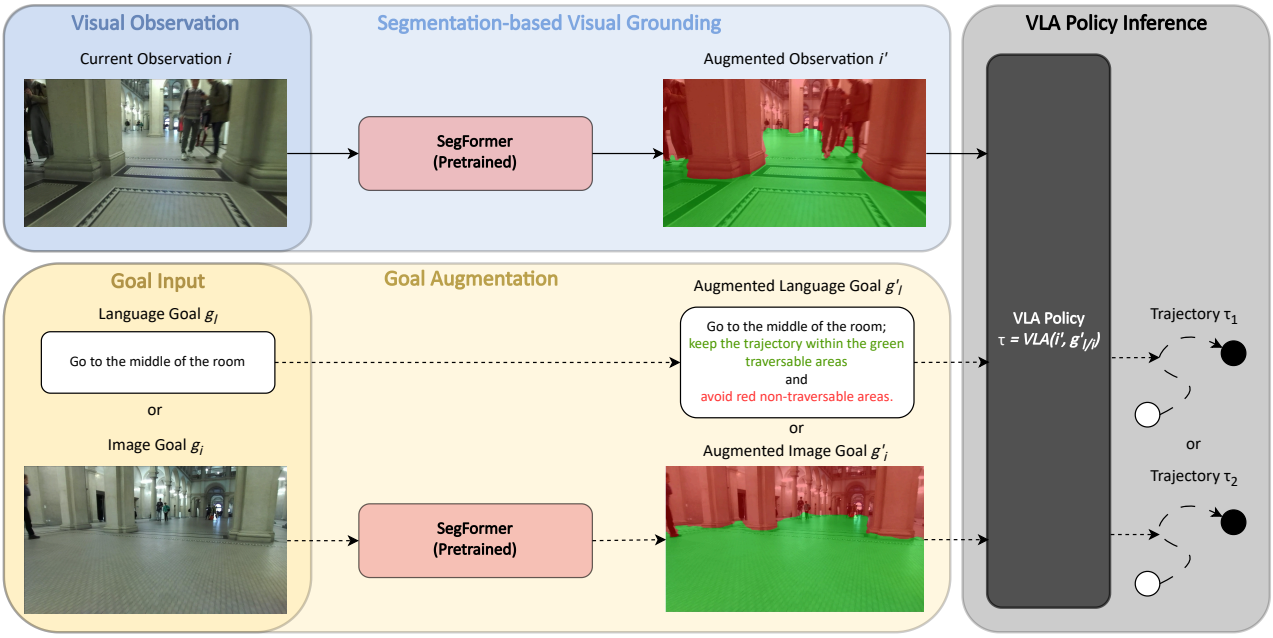

Fig. 1. Overview of the proposed segmentation-based visual grounding and goal augmentation method. SegFormer produces real-time semantic segmentation to identify traversable (green) and non-traversable (red) regions. These overlays are used to augment both current observation and goals, which are then passed to the VLA policy to produce trajectories τ_1 and τ_2.

for navigation. LM-NAV [15] uses GPT-3 [5] to extract landmark names from a language navigation instruction, then uses CLIP [14] to match those names to visual regions in the robot's camera image, enabling zero-shot outdoor navigation. VLMaps [10] uses spatial map representation that directly fuses pretrained visual-language features with a 3D reconstruction of the physical world, allowing natural language-guided navigation. Recently, VL-TGS [16] generates multiple candidate trajectories using a CVAE-based approach [1] and selects the most suitable trajectory using a VLM with visual prompting.

As a more recent development, Vision-Language-Action models, while mostly used for manipulation tasks, have also been applied to navigation tasks. MobilityVLA [4] uses topological graphs to satisfy natural language or multimodal requests after observing a prerecorded demonstration tour. NaVILA [3] uses a 2-level framework that unifies a VLA policy with locomotion skills. Instead of directly predicting low-level actions from VLA, NaVILA first generates mid-level actions with spatial information in the form of a language instruction. OmniVLA [9] trains a VLA policy with three primary goal modalities: natural language, egocentric images, 2D poses, and their combinations. This allows the model to have a strong generalization to unseen environments and the ability to follow novel language instructions.

B. Visual Grounding and Language Models

Although large language models are great at generating sentences, their ability to reason, validate their outputs, and, in the case of vision language models, comprehend images remains limited [11]. Therefore, visual grounding has been used to help language models understand and act on images more effectively. Visual grounding techniques help by pre-processing the image inputs by adding spatially relevant and easily comprehensible features. Set-of-Marks [19] allows GPT-4V [2] to correctly answer questions about an image by segmenting using SEEM/SAM [20, 12] and adding a numerical mark over each object, therefore helping the model to easily comprehend and refer to different objects. PIVOT [13] casts robotic control as a VQA problem by annotating an image with a visual representation of robot actions and iteratively querying a VLM to select the most fitting action. BYOVLA [8] uses GPT4-o [2] to select task-irrelevant regions for manipulation tasks and fully mask irrelevant regions that distract the VLA, which allows the model to focus on the relevant parts without being distracted by backgrounds and irrelevant objects.

III. Problem Formulation

This paper presents an evaluation of a visual grounding technique in the context of a vision-language-action model used for robot navigation. The problem is formulated as a trajectory generation task predicted from the current visual observation and a goal given either by a language instruction or a visual goal in the form of an image.

The trajectory generation is defined as:

$$\tau = VLA(i, g), \tag{1}$$

where the input i is the current egocentric observation image from the robot's camera, and $g \in \{g_l, g_i\}$ is either a language instruction g_l or a goal image g_i. The output

$\tau = \{(x_1, y_1), \dots (x_7, y_7)\}$ is a sequence of 7 waypoints in a 2D coordinate system that represents the predicted navigation path.

IV. Method

We propose a visual grounding pipeline that overlays semantic segmentation onto egocentric observations before feeding them to a VLA policy. The grounding has two variants: observation-only grounding, which segments only the current visual observation, and joint grounding, which additionally augments the goal modality. The overview of the method can be seen in Figure 1.

A. Segmentation-Based Visual Grounding

Given an egocentric RGB image from the robot's camera, we apply a semantic segmentation model to classify each pixel as either traversable or non-traversable. We use SegFormer [18], a transformer-based segmentation framework with a hierarchical encoder and a lightweight MLP decoder, for binary segmentation, producing a pixel-wise mask where traversable regions (floor) are distinguished from non-traversable obstacles (walls, furniture, stairs).

The resulting mask is overlaid onto the original image: traversable areas are highlighted in green, non-traversable areas in red. This produces the grounded observation i'. The input segmentation is shown in Figure 1 as the blue box.

For observation-only grounding, the VLA generates a trajectory as follows:

$$\tau_1 = VLA(i', g), \tag{2}$$

For joint grounding, the goal is also augmented. For image goals, the same segmentation overlay is applied to the goal image. For language goals, we append an explicit specification to the instruction: "keep the trajectory within green traversable areas and avoid red non-traversable areas". This yields the augmented goal $g' \in \{g'_l, g'_i\}$. The goal augmentation is shown in Figure 1 as the yellow box.

The trajectory for the joint grounding, where the input is segmented and goal is augmented, is then:

$$\tau_2 = VLA(i', g'), \tag{3}$$

The design choices were constrained by three requirements: real-time inference, generalization to unseen environments (no fixed object vocabulary), and multimodal goal support (language and images). LLM-based grounding methods such as Set-of-Marks [19] require chain-of-thought prompting, which increases the inference time from $0.5s$ to $5s$ per image, which is infeasible for real-time navigation. Object detection methods such as YOLO [17] are fast but rely on a fixed set of object classes, severely limiting generalization to novel obstacles in unseen environments. Segmentation-based grounding avoids both issues: SegFormer runs in real time and generalizes to any traversable/non-traversable distinction without a predefined vocabulary.

V. Experiments

This section covers the experimental setup, model and dataset choice, what experiments were conducted, and the results from all the experiments.

A. Experimental Setup

OmniVLA [9] has been chosen as the evaluated VLA model due to its competitive success rate among other contemporary VLA models and due to its flexibility when it comes to the goal modality, allowing the evaluation of visual grounding not only in the context of vision language navigation, but also in the context of vision goal navigation. The model predicts a trajectory consisting of 7 waypoints. The experiments were carried out on an NVIDIA Jetson Thor.

The dataset used in the experimental evaluation is the ETH-2 episode from the Grand Tour dataset [6], which is the largest legged robotics dataset collected using an ANYmal D quadruped robot with multimodal sensors that capture images, depth, proprioception, and more. Out of the dataset a specific episode has been chosen that was indoor and had a sizable number of describable objects for language instructions. The episode has been split into segments, where for each segment the model was given a specific language instruction or the final image of the segment in the case of goal image experiments. All segments and their language goals can be seen in Table II.

The experiments included 3 full runs of the chosen episode, where 1) the default *omnivla-base* model checkpoint has been used, 2) the finetuned *omnivla-finetuned-cast* model checkpoint has been used, which has been finetuned with the CAST [7] dataset that aims to improve performance when using simpler, atomic instructions, and 3) the default *omnivla-base* model checkpoint, but using image goals as input instead of language goals. For each full run, the model generated 3 trajectories: the original one that used the original current visual observation and the goal, labelled in plots as *predicted_base*, and then two trajectories which use visual grounding; *predicted_segmented*, which uses the segmented current visual observation and the original goal, and *predicted_segmented_augmented*, which uses the segmented current visual observation and the augmented goal.

B. Experimental Results

We evaluated three prediction conditions across three model configurations: (1) base (original observation and goal), (2) segmented (grounded observation only), and (3) augmented (grounded observation + augmented goal). Performance was measured as the minimum Euclidean distance from each predicted waypoint to the ground-truth trajectory. Waypoints are indexed from 1 (nearest) to 7 (farthest). A small number of segments of the evaluated episode have been omitted in the final evaluations. Due to the limitation of the model to generate a trajectory only in 2-dimensional spaces and the nature of the segmentation technique used that tends to mark stairs as a non-traversable region, the beginning of the episode where the robot walks up the stairs has been omitted, together with all the segments where the robot stops, as explained later.

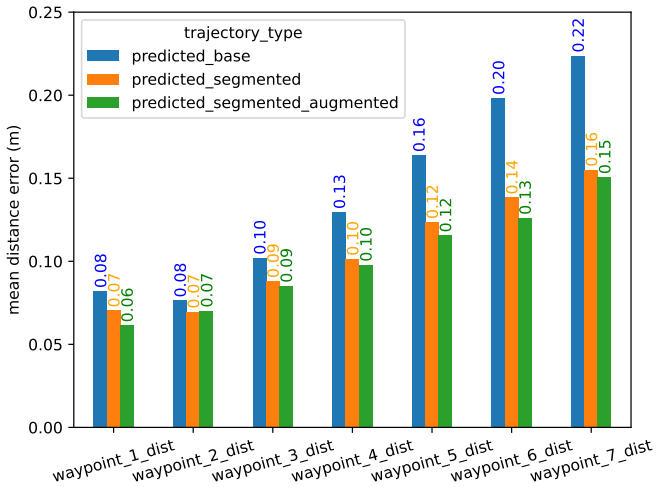

Fig. 2. The mean error for each waypoint for the base prediction (model = *omnivla-base*).

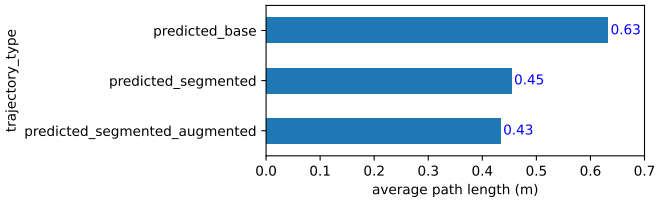

Fig. 3. The average length of the predicted trajectories (model = *omnivla-base*).

Visual grounding consistently reduced prediction error at all waypoints. For the *omnivla-base* model with language goals, the average error at the farthest waypoint decreased from 0.22 m (base) to 0.16 m (segmented) and 0.15 m (augmented); reductions of 27% and 32%, respectively. The effect is more pronounced at farther waypoints; at waypoint 1, the reduction in error was only $13-25\%$. Figure 2 visualizes this trend: the error-vs-distance curve is less steep for grounded predictions, indicating that grounding helps the most where the model otherwise struggles. The disaggregation by instruction length (Table I) shows that this far-waypoint benefit is greater for long instructions ($39-44\%$ error reduction at WP7) than for short instructions ($35-36\%$).

However, grounded predictions also produced shorter trajectories. The base predictions averaged 0.63 m in length, while the segmented and augmented predictions averaged 0.45 m and 0.43 m, respectively, as seen in Figure 3. When we normalize error by trajectory length, the advantage of grounding largely disappears. As shown in Figure 4, normalized errors converge across all three conditions. In fact, at waypoints $1-5$, base predictions show slightly lower normalized error than the grounded variants. This suggests that grounding primarily acts as a trajectory length regularizer rather than fundamentally improving the VLA's spatial reasoning per unit distance.

We compared performance across three model runs: *omnivla-base* with language goals, *omnivla-finetuned-cast* (trained on atomic instructions), and *omnivla-base* with image goals. The base + image configuration had the lowest base error of 0.13 m, consistent with the reported strength of OmniVLA in visual goal navigation. Consequently, grounding provided little to no benefit and, in fact, slightly increased error for some segments. This indicates that visual grounding is more effective when paired with language goals than with image goals, likely because language instructions require the model to map abstract descriptions to spatial regions, a task that explicit traversability cues directly support. Figure 5 summarizes the results for each model/goal modality. The base + image configuration achieved 0.13 m average error, and grounding provided a very small improvement $0.12-0.11$ m, compared to language goals where grounding reduced the error from $0.14-0.16$ m to $0.10-0.11$ m.

We observed a mild negative correlation $r \approx -0.54$ between the number of words in the language instruction and the trajec-

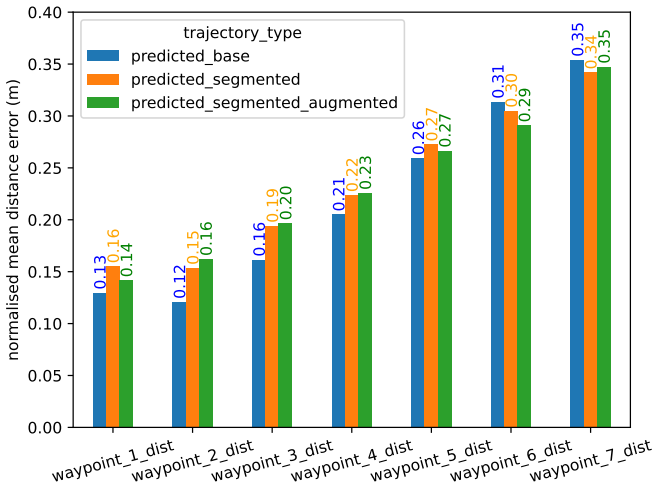

Fig. 4. The normalised mean error for each waypoint for the base prediction (model = *omnivla-base*).

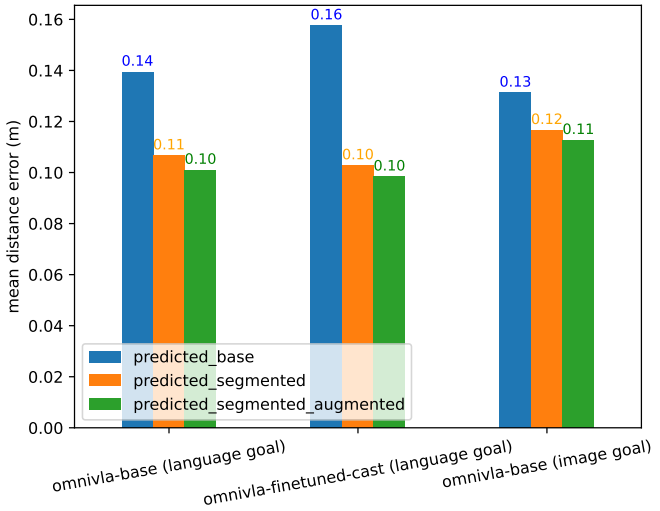

Fig. 5. The overall mean distance error for each model/goal modality.

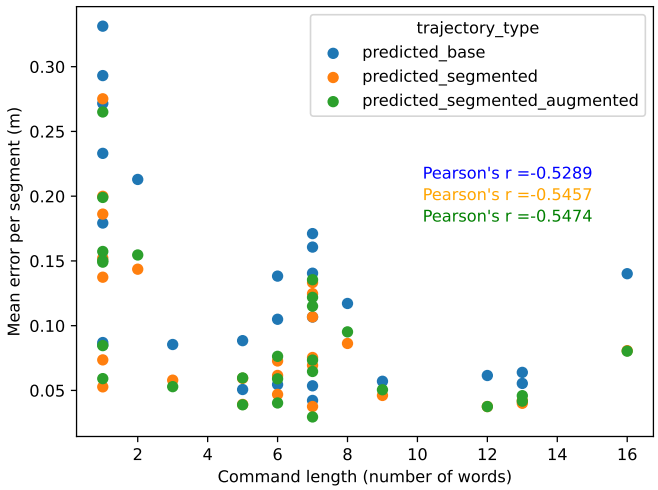

Fig. 6. Mean error per segment vs instruction length of that segment (model = *omnivla-finetuned-cast*).

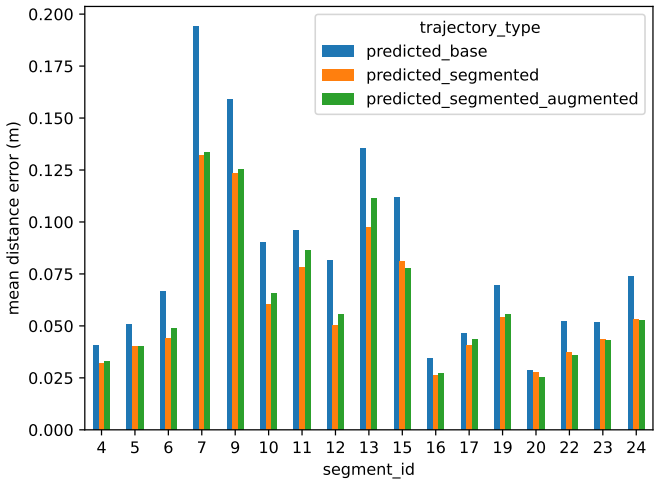

Fig. 7. The mean error for each segment by trajectory type, stops and stair segments removed (model = *omnivla-base*).

Instruction length	Waypoint	Base	Segmented	Augmented
"Stop" only	1	0.204	0.149	**0.100**
	1-7	0.274	0.174	**0.161**
Short (2 - 7 words)	1 (near)	0.081	0.069	**0.060**
	3	0.053	0.048	**0.047**
	5	0.108	0.074	**0.071**
	7 (far)	0.187	**0.120**	0.122
Long (>= 8 words)	1	0.074	0.050	**0.048**
	3	0.054	**0.045**	**0.045**
	5	0.098	**0.060**	0.061
	7	0.160	**0.090**	0.098
Overall average (without stop)	1-7	0.095	**0.064**	**0.064**

TABLE I
PER-WAYPOINT PREDICTION ERROR (METERS) BY GROUNDING TYPE, DISAGGREGATED BY INSTRUCTION LENGTH. (VALUES SHOW MEAN EUCLIDEAN DISTANCE FROM PREDICTED TO GROUND-TRUTH WAYPOINTS. SHORT INSTRUCTIONS (13 SEGMENTS) INCLUDE ATOMIC INSTRUCTIONS SUCH AS "TURN LEFT"; LONG INSTRUCTIONS (6 SEGMENTS) ARE DESCRIPTIVE NAVIGATION GOALS. BEST-PERFORMING GROUNDING CONDITION PER ROW IS BOLDED.)

tory error (Figure 6). Instructions with 8 to 16 words produced a consistently lower error than single-word instructions. An extra outlier was the instruction "stop" (1 word), which the model failed to execute entirely.

The model never generated a zero-displacement trajectory when given the language goal "stop" or an image goal identical to the current observation. Instead, it consistently predicted forward motion. We hypothesize that the VLA's training data lacked stationary instructions, making this an out-of-distribution input. This failure persists across all three model variants and both grounding conditions, indicating that perceptual augmentation alone cannot compensate for the missing training signal. As quantified in the "stop"-only row of Table I, the model's error at WP1 ranges from 0.1 m (augmented) to 0.204 m (base) instead of the ideal 0, with the average showing persistent error between 0.161 m and 0.274 m across all grounding conditions.

For approximately half of the segments, joint grounding (augmented goals) did not perform better than observation-only grounding (Figure 7). Augmentation helped most for ambiguous instructions, but offered little advantage when instructions were already descriptive. This suggests that the appended instruction is redundant when the language goal already implies traversability constraints. See Table II in the Appendix for all language instructions.

VI. LIMITATIONS

Despite promising results, our approach has inherent limitations. First, the segmentation method assumes static, 2D traversability, therefore dynamic obstacles and 3D structures (e.g. stairs) are not handled. Second, the red/green overlay is a simple heuristic; optimal color mapping or segmentation encoding for VLA inputs remain unexplored. Third, our grounding requires the VLA to interpret visual overlays correctly, and there is no guarantee that other VLAs with different vision encoders would respond similarly. Fourth, the "stop" instruction failure is not solved by grounding, indicating that certain atomic instructions may require explicit training data rather than perceptual augmentation.

Ultimately, the results of this preliminary study are limited to a single indoor episode and should be interpreted with caution.

VII. CONCLUSION

To our knowledge, this paper presents the first empirical evaluation of visual grounding for vision-language-action navigation policies. We introduced a real-time, segmentation-based grounding method using SegFormer that satisfies the three constraints for navigation: real-time inference, generalization to unseen environments, and multimodal goal support. Through ablative analysis on the Grand Tour dataset using OmniVLA, we demonstrated that visual grounding consistently reduced waypoint prediction error, particularly at the farthest waypoint

($27-44\%$ reduction depending on instruction length), and is substantially more effective for language goals than for image goals.

Our results reveal several important insights. First, while grounding reduces the absolute far-waypoint error by $27-44\%$, normalized error becomes comparable to baseline, and grounded trajectories are 30% shorter, confirming that grounding primarily acts as a trajectory length regularizer rather than fundamentally improving spatial reasoning. Second, the mild negative correlation between instruction verbosity and error indicates that VLA models benefit from more descriptive language goals. Third, the consistent failure to generate zero-displacement "stop" trajectories reveals a critical gap in current VLA training paradigms, as the model cannot represent or execute stationary instructions even when explicitly instructed.

Within the boundaries of our limitations, we have shown that visual grounding offers a simple, computationally inexpensive, and effective method to improve VLA navigation performance without model retraining.

Future work can extend this approach in several directions. Online closed-loop evaluation would validate whether the observed open-loop error reductions translate to improved task completion rates. Extending grounding to handle dynamic obstacles and 3D structures (e.g. stairs) would address the most significant limitation of the current method. Finally, the "stop" instruction failure suggests that combining perceptual grounding with targeted fine-tuning on atomic instructions may be necessary for full coverage of primitive navigation commands.

Acknowledgments

This work was supported by the CDT in Cybersecurity [EP/S022503/1] and the UKRI FLF [MR/V025333/1] (RoboHike). For the purpose of Open Access, the author has applied a CC BY public copyright license to any Author Accepted Manuscript version arising from this submission.

References

[1] Abubakar Abid and James Zou. Contrastive variational autoencoder enhances salient features. *arXiv preprint arXiv:1902.04601*, 2019.

[2] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[3] An-Chieh Cheng, Yandong Ji, Zhaojing Yang, Zaitian Gongye, Xueyan Zou, Jan Kautz, Erdem Bıyık, Hongxu Yin, Sifei Liu, and Xiaolong Wang. Navila: Legged robot vision-language-action model for navigation. *arXiv preprint arXiv:2412.04453*, 2024.

[4] Hao-Tien Lewis Chiang, Zhuo Xu, Zipeng Fu, Mithun George Jacob, Tingnan Zhang, Tsang-Wei Edward Lee, Wenhao Yu, Connor Schenck, David Rendleman, Dhruv Shah, et al. Mobility vla: Multimodal instruction navigation with long-context vlms and topological graphs. *arXiv preprint arXiv:2407.07775*, 2024.

[5] Luciano Floridi and Massimo Chiriatti. Gpt-3: Its nature, scope, limits, and consequences. *Minds and machines*, 30(4):681–694, 2020.

[6] Jonas Frey, Turcan Tuna, Frank Fu, Katharine Patterson, Tianao Xu, Maurice Fallon, Cesar Cadena, and Marco Hutter. Grandtour: A legged robotics dataset in the wild for multi-modal perception and state estimation. *arXiv preprint arXiv:2602.18164*, 2026.

[7] Catherine Glossop, William Chen, Arjun Bhorkar, Dhruv Shah, and Sergey Levine. Cast: Counterfactual labels improve instruction following in vision-language-action models. *arXiv preprint arXiv:2508.13446*, 2025.

[8] Asher J Hancock, Allen Z Ren, and Anirudha Majumdar. Run-time observation interventions make vision-language-action models more visually robust. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9499–9506. IEEE, 2025.

[9] Noriaki Hirose, Catherine Glossop, Dhruv Shah, and Sergey Levine. Omnivla: An omni-modal vision-language-action model for robot navigation. *arXiv preprint arXiv:2509.19480*, 2025.

[10] Chenguang Huang, Oier Mees, Andy Zeng, and Wolfram Burgard. Visual language maps for robot navigation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10608–10615. IEEE, 2023.

[11] Shaohan Huang, Li Dong, Wenhui Wang, Yaru Hao, Saksham Singhal, Shuming Ma, Tengchao Lv, Lei Cui, Owais Khan Mohammed, Barun Patra, et al. Language is not all you need: Aligning perception with language models. *Advances in Neural Information Processing Systems*, 36:72096–72109, 2023.

[12] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023.

[13] Soroush Nasiriany, Fei Xia, Wenhao Yu, Ted Xiao, Jacky Liang, Ishita Dasgupta, Annie Xie, Danny Driess, Ayzaan Wahid, Zhuo Xu, et al. Pivot: Iterative visual prompting elicits actionable knowledge for vlms. *arXiv preprint arXiv:2402.07872*, 2024.

[14] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

[15] Dhruv Shah, Błażej Osiński, Sergey Levine, et al. Lm-nav: Robotic navigation with large pre-trained models of language, vision, and action. In *Conference on robot learning*, pages 492–504. pmlr, 2023.

[16] Daeun Song, Jing Liang, Xuesu Xiao, and Dinesh Manocha. Vl-tgs: Trajectory generation and selection

using vision language models in mapless outdoor environments. *IEEE Robotics and Automation Letters*, 2025.

[17] Juan Terven, Diana-Margarita Córdova-Esparza, and Julio-Alejandro Romero-González. A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas. *Machine learning and knowledge extraction*, 5(4):1680–1716, 2023.

[18] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in neural information processing systems*, 34:12077–12090, 2021.

[19] Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv preprint arXiv:2310.11441*, 2023.

[20] Xueyan Zou, Jianwei Yang, Hao Zhang, Feng Li, Linjie Li, Jianfeng Wang, Lijuan Wang, Jianfeng Gao, and Yong Jae Lee. Segment everything everywhere all at once. *Advances in neural information processing systems*, 36:19769–19782, 2023.

APPENDIX

segment_id	start_s	language_goal
0	0	Stop
1	6	Go up the middle of the stairs
2	16	Stop
3	33	Go up the middle of the stairs
4	37	Go forward through the middle archway
5	54	Turn left after passing the left pillar and go towards the gray cabinet
6	74	Keep going straight down the corridor
7	88	Turn right
8	93	Stop
9	124	Turn left; go forward; and turn right
10	130	Go forward and towards the right pillar
11	139	Go between the two pillars; then turn right
12	149	Go forward past the white boxes
13	165	Turn right 45 degrees and go forward
14	171	Stop
15	182	Turn right and go straight down the hallway and to the right of the right pillar
16	221	Go forward between the pillars
17	236	Go forward and around the fountain on the left
18	256	Stop
19	262	Go forward between the pillar
20	283	Go to the middle of the room
21	291	Stop
22	299	Go to the other side of the room along the right pillars
23	322	Go to the other side of the room and through the middle archway
24	338	Keep going forward
25	346	Stop

TABLE II
TABLE OF ALL THE LANGUAGE GOALS FOR EACH SEGMENT AND THEIR CORRESPONDING SEGMENT ID AND STARTING SECOND IN THE GRAND TOUR EPISODE ETH-2.