NaVid-4D: Unleashing Spatial Intelligence in Egocentric RGB-D Videos for Vision-and-Language Navigation

Haoran Liu[1,2*], Weikang Wan[1,2*], Xiqian Yu[3,2*], Minghan Li[2*], Jiazhao Zhang[1,2], Bo Zhao[4]
Zhibo Chen[3], Zhongyuan Wang[5], Zhizheng Zhang[2,5†], He Wang[1,2,5†]

***Abstract*— Understanding and reasoning about the 4D space-time is crucial for Vision-and-Language Navigation (VLN). However, previous works lack in-depth exploration in this aspect, resulting in bottlenecked spatial perception and action precision of VLN agents. In this work, we introduce NaVid-4D, a Vision Language Model (VLM) based navigation agent taking the lead in explicitly showcasing the capabilities of spatial intelligence in the real world. Given natural language instructions, NaVid-4D requires only egocentric RGB-D video streams as observations to perform spatial understanding and reasoning for generating precise instruction-following robotic actions. NaVid-4D learns navigation policies using the data from simulation environments and is endowed with precise spatial understanding and reasoning capabilities using web data. Without the need to pre-train an RGB-D foundation model, we propose a method capable of directly injecting the depth features into the visual encoder of a VLM. We further compare the use of factually captured depth information with the monocularly estimated one and find NaVid-4D works well with both while using estimated depth offers greater generalization capability and better mitigates the sim-to-real gap. Extensive experiments demonstrate that NaVid-4D achieves state-of-the-art performance in simulation environment and makes impressive VLN performance with spatial intelligence happen in the real world.**

I. Introduction

Vision-and-language navigation (VLN) is a fundamental yet challenging task in embodied AI, requiring robots to navigate unseen environments based on visual input and natural language instructions [2, 23, 39, 41]. Recent advances have shifted VLN from a discrete simulator setting [4, 12, 20, 40, 51, 60, 69, 73] to a more realistic continuous setting in the real-world [28, 39, 50, 52, 55, 72], enabling robots to navigate more naturally, like humans, rather than merely transitioning between waypoints on a predefined navigation graph. Developing a truly generalizable real-world VLN system places greater demands on 3D spatial intelligence, particularly in reasoning about spatial relationships [18, 47, 70] (e.g., following instructions like "enter the farthest room"). Emerging large Vision-Language Models (VLMs) show great potential for addressing these challenges and shaping the future of VLN research due to their expansive perception and advanced language comprehension capabilities.

The latest VLN agents [11, 72] leverage Vision-Language Models (VLMs) to reason about spatial-temporal relationships during navigation by modeling historical and current observations through ViT-generated visual tokens. While these approaches benefit from the perceptual capabilities of current VLMs, they remain limited by a key drawback: existing VLMs are based on RGB vision foundation models that do not explicitly incorporate depth information, leading to insufficient 3D scene understanding and suboptimal performance when tasks require reasoning about spatial relationships.

Fig. 1: **4D spatio-temporal reasoning capabilities in vision-language navigation.** NaVid-4D can comprehend and reason about 3D spatial and 1D temporal relationships across diverse tasks and directly predict actions to follow given instructions. (The images with blue, grey, and yellow borders represent the start, middle, and end states of the task, respectively.)

Recent research has begun to address these limitations by improving spatial reasoning and understanding in VLMs. For instance, SpatialVLM [8] introduces a data generation pipeline that facilitates large-scale training on spatially-aware visual question-answering (VQA) tasks using 2D input. SpatialRGPT [14] and SpatialBot [6] further enhance the spatial reasoning capabilities of VLMs by integrating 3D inputs into their architectures. While these advancements have demonstrated spatial intelligence in digital question-answering tasks, a gap remains before they can be effectively applied to embodied Vision-Language-Action (VLA) models. This work seeks to bridge that gap by exploring the 3D spatial perception capabilities needed for robots to complete VLN tasks, demonstrating spatial intelligence in the physical

[1]CFCS, School of Computer Science, Peking University, [2]Galbot, [3]University of Science and Technology of China, [4]Shanghai Jiao Tong University, [5]Beijing Academy of Artificial Intelligence. *: equal contributions, †: corresponding authors (e-mail: zhangzz@galbot.com, hewang@pku.edu.cn).

world.

In this work, we introduce NaVid-4D, an end-to-end VLA model that solely requires egocentric RGB-D video stream and natural language instructions as model inputs to generate navigation actions in continuous environments. It makes the first endeavor to enhance spatial understanding and reasoning capabilities in VLN tasks and showcase spatial intelligence in the physical world. Like its predecessor NaVid [72], NaVid-4D is built upon a pre-trained VLM consisting of a vision foundation model and a large language model (LLM) to exploit the general-purpose knowledge acquired from large-scale pre-training. It is extended to an end-to-end VLA model for robotic navigation by learning the navigation policy on data from the simulation environment and further enhancing the perception with Internet data. Compared to NaVid, which is RGB-only, NaVid-4D further explores three essential questions: 1) Do we need explicit 3D features in VLN tasks? 2) How can 3D features be integrated into pre-trained foundation models efficiently? 3) What is the impact of different 3D information sources (captured vs. estimated) on the performance of VLA models?

A series of studies on VLMs [6, 8, 14] have demonstrated that depth information is necessary for digital tasks requiring spatial understanding and reasoning. Embodied intelligence tasks, which require generating actions in the physical world, have even higher demands for spatial perception. Therefore, we believe that explicitly utilizing depth information in VLA models is also crucial. So far, the biggest challenge lies in the lack of sufficiently powerful RGB-D vision foundation models, as acquiring large-scale RGB-D image-text pairs for pretraining is extremely costly. To address this, we propose a novel method to efficiently extract depth features and inject them into an off-the-shelf RGB-based vision foundation model as the RGB-D encoder. Given the domain shifts between navigation data and co-trained web data, as well as between simulation and real-world environments, we further compare captured depth with estimated depth in terms of their impacts on end-to-end performance. We find that NaVid-4D works well with both, while estimated depth offers greater generalization capability and better mitigates the sim-to-real gap. Besides, we collect 1.8M image-text pairs as [6] on digital question-answering tasks and adopt them in a co-training for NaVid-4D to enhance its spatial perception.

The contributions of this work can be summarized in three aspects: 1) We build NaVid-4D, an end-to-end VLA model that solely requires egocentric RGB-D video stream and natural language instructions as model inputs to generate navigation actions in continuous environments. 2) We propose a novel model paradigm and its corresponding training strategy to efficiently inject depth representations into the existing RGB-based foundation model, obviating the high cost of building RGB-D foundation models. 3) We demonstrate the benefits of explicitly integrating depth information for VLN in both simulation environments and sim-to-real generalization and compare the impacts of adopting captured or estimated depth information on the final end-to-end performance.

II. Related Work

Vision-and-Language Navigation. Visual-Language Navigation (VLN) [2] has attracted significant attention in recent years. A common approach in simulated environments is to discretize the scene [4, 20, 54, 62], where the robot agent makes decisions by aligning language and visual observations to move by teleportation between nodes on a predefined navigation graph [28, 35, 45, 53, 65, 66]. However, these approaches often perform poorly when directly transferring VLN models trained in discrete spaces to continuous 3D real-world robotic applications [19, 27]. To address this issue, [39] has utilized the Habitat simulator [58] and proposed a visual-language navigation benchmark in continuous environments (VLN-CE), allowing robot agents to navigate freely to any unobstructed space within the simulator. At each time step, the agent predicts actions based on vision observations and language instructions, using direct low-level control prediction [9, 10, 21, 56] or selecting navigable subgoals estimated by a waypoint predictor [27, 37, 38]. Recently, large language models (LLMs) and vision-language models (VLMs) [1, 42, 55], trained on internet-scale image, video, and text data, have demonstrated exceptional capabilities in multimodal reasoning and cross-domain generalization. Many VLN models have benefited from integrating these foundation models into their architectures [12, 13, 28, 72]. Our work is more related to the recent SOTA work NaVid [72], which is a video-based large VLN model with only RGB input. In this paper, we propose a VLM-based navigation model capable of utilizing RGB-D information which significantly improves the spatial reasoning capability and navigation performance.

Spatial Reasoning via Vision-Language Models. Recently, significant efforts have been made to improve the spatial reasoning capabilities of vision-language models (VLMs). Pioneering studies [29, 32, 67] have primarily concentrated on incorporating 3D representations, such as multi-view images or point clouds, to infuse spatial information into VLMs. However, the limited availability of multi-view images and point cloud data constrains the effectiveness of these approaches. Meanwhile, some approaches have sought to enhance spatial reasoning abilities without directly incorporating 3D representations. For instance, ConceptGraph [24] integrates scene graphs into VLMs to capture spatial relationships. SpatialVLM [8] constructs an Internet-scale 3D spatial reasoning dataset to train 2D VLMs, significantly improving their performance on spatial Visual Question Answering (VQA) tasks. Our work is related to SpatialRGPT [14], which integrates depth information for enhancing region-level spatial reasoning in VLMs. In this paper, we focus on leveraging depth information to enhance the spatial reasoning capabilities of our VLM-based navigation agent.

III. Method: NaVid-4D

A. Task Formulation

In this section, we present the formulation of Vision-and-Language Navigation in Continuous Environments (VLN-

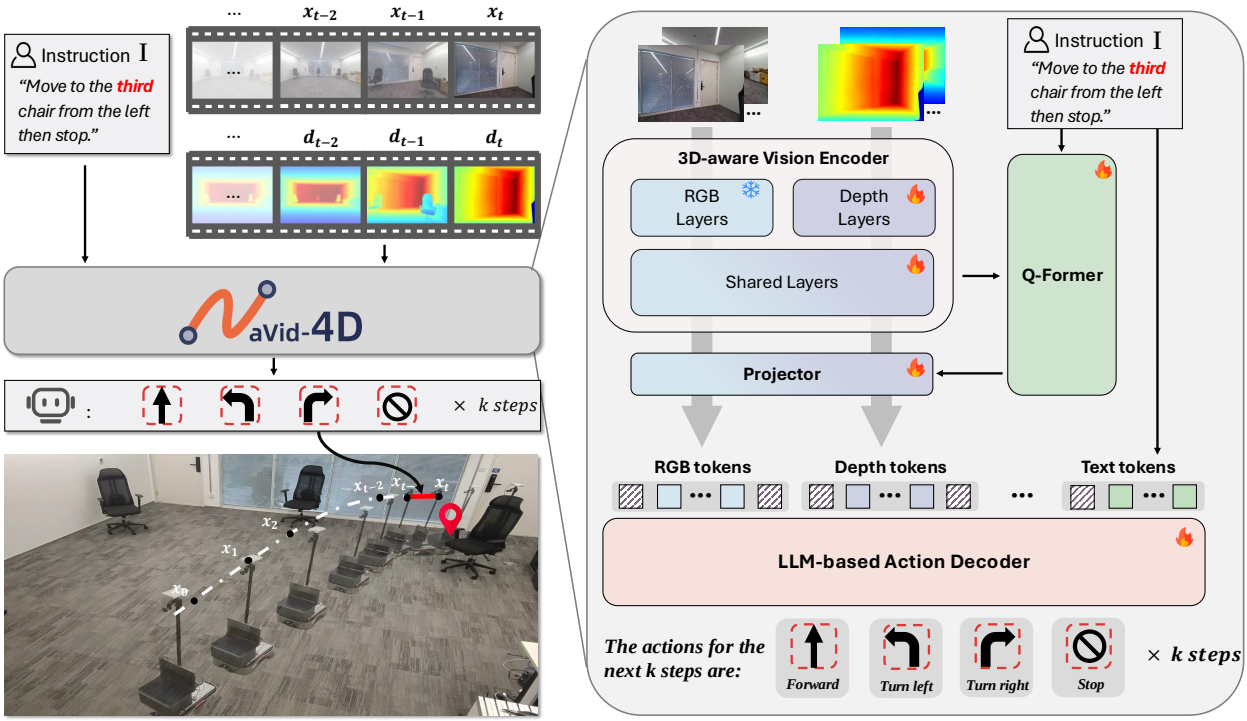

Fig. 2: **The framework of NaVid-4D.** At each time step, NaVid-4D takes as input an egocentric RGB-D video stream and the given instruction to generate low-level executable action for the next k steps in an end-to-end manner.

CE). At time step t, the agent is provided with a language instruction $\mathcal{I}$, comprising l words, and a video stream $\mathcal{O}_t = \{o_0, o_1, \ldots, o_t\}$, comprising a sequence of RGB frames $\{x_0, x_1, \ldots, x_t\}$ and their corresponding depth maps $\{d_0, d_1, \ldots, d_t\}$. At time t, the agent needs to plan low-level actions $\{a_t, a_{t+1}, \ldots, a_{t+k-1}\}$ for the next k steps based on the current observation o_t following the instruction $\mathcal{I}$. The actions at each timestep can be one of the four choices: forward, turn left, turn right, or stop. After $a_t, a_{t+1}, \ldots, a_{t+k-1}$ are executed, the agent will receive a new observation o_{t+k}. In this work, we build a VLA agent to generate $a_t, a_{t+1}, \ldots, a_{t+k-1}$ from an egocentric video stream $\mathcal{O}_t$ and the instruction $\mathcal{I}$ at each time step in an end-to-end manner. We dive into the role and methodology of explicitly utilizing depth information to unleash spatial intelligence in a 4D modeling of VLN-CE.

B. Model Paradigm

NaVid-4D is an end-to-end Vision-Language-Action (VLA) model designed for the VLN-CE task, extended from a pre-trained Vision Language Model (VLM). It takes as input a natural language instruction and an RGB-D video stream from a monocular egocentric camera for continuous navigation action generation in an end-to-end manner. The inputs of NaVid-4D include three fine-grained modalities, *i.e.*, RGB, depth, and language. Although RGB and depth both belong to the visual modality, they represent related yet distinct types of information. RGB is better suited for capturing texture information, while depth maps are more effective at conveying geometric information. Considering this nature, we design a novel model paradigm to align different modalities progressively. Specifically, the model first encodes RGB and depth information into a shared space. They are then projected, together with the language modality, into a more generalized feature space. It is worth noting that this model paradigm can be built on top of a classical pre-trained VLM, leveraging its general-purpose knowledge to enhance the learning of navigation policies.

As illustrated in Fig. 2, the model paradigm of NaVid-4D comprises a 3D-aware vision encoder, a projector coupled with a Q-Former, and an LLM-based action decoder. We elaborate on them below.

3D-aware Vision Encoder. We employ a classical ViT-based CLIP model [55, 64] and extend it to be the 3D-aware vision encoder in NaVid-4D. This encoder, initialized from the CLIP model weights pre-trained on large-scale RGB images, is responsible for learning visual representations from both RGB and depth images. Given the distinct characteristics of RGB and depth information discussed earlier, we propose to encode RGB and depth separately using different network parameters in the shallow layers of the vision encoder, referred to as the "RGB Layers" and "Depth Layers" in Fig. 2, respectively. These two share the same architecture and are initialized with the same pre-trained weights from the RGB-based CLIP model and then evolve individually with our proposed training strategy introduced later. After these layers, the RGB and depth information are represented in a relatively aligned feature space. We then further encode them with "Shared Layers" to get a set of visual tokens. These tokens encode visual information in a complementary way, whereas RGB and depth tokens encode texture and geometric

features cooperatively. Similar to NaVid [72], we compress the visual tokens when encoding historical frames. For each time step, experimentally, we recommend adopting 4 content tokens and 1 context token per historical frame while using 512 content tokens and 2 context tokens (257 RGB tokens and 257 depth tokens in total) for the current frame.

Token Projector. We adopt a projector coupled with a Q-Former to further transform visual tokens into a more generalized space with language aligned in it, making them compatible with a pre-trained LLM. Its design is similar to those in [43, 72], and readers can refer to them for details.

Action Decoder. The action decoder in NaVid-4D is extended from an open-source LLM, *i.e.*, Vicuna-7B [15], by adding a set of action tokens into its vocabulary dictionary. These tokens denote different low-level actions called "forward", "turn left", "turn right" and "stop". Different from NaVid [72], which simultaneously predicts an action type and a corresponding magnitude for execution in the next step, NaVid-4D instead predicts a sequence of discrete actions for the next k steps at each inference. We experimentally observe that this action modeling achieves performance comparable with that of NaVid but offers better sample efficiency. Similar to NaVid, we employ special tokens to organize multi-modal tokens as illustrated in Fig. 2 to facilitate the training.

C. Training Strategy

The training process of NaVid-4D consists of four distinct stages: 1) *Projector Warmup*, 2) *Vision Encoder Training*, 3) *Instruction-tuning*, and 4) *DAgger Improvement*. In the first stage, considering that both the vision encoder and the LLM-based action decoder have undergone large-scale pretraining, we warm up the projector while keeping the remaining parts frozen. In this stage, all layers of the vision encoder are shared over RGB and depth information. When entering the second stage, we train the 20 "Depth Layers" along with the subsequent 20 "Shared Layers" while keeping the "RGB Layers" and the action decoder frozen. During the last two stages, we unfreeze the LLM-based action decoder to enhance the alignment of action with instruction and freeze the vision encoder. Throughout the first three stages, we adopt our constructed training data consisting of both navigation policy data and semantic understanding data introduced below. For the final stage, we further incorporate data generated by rolling out the policy using the DAgger [57] algorithm.

D. Dataset Construction

Training Data Composition. As introduced earlier, we collect 320K vision-language-action samples from the simulation environment Habitat [58] to learn navigation policies and generate 10K navigation instruction reasoning samples as NaVid [72] to enhance instruction understanding. Besides, we utilize 1.32M question-answering samples from [22, 26, 33, 34, 44, 48, 49, 59, 63], 98K question-answering samples from [5], and 40K text samples from [61] to endow NaVid-4D with general multi-modal LLM capabilities. Additionally, we incorporate 20K depth map understanding samples, 21K spatial understanding samples, and 7.5K robot scene understanding samples from SpatialBot [6] to further enhance the model's spatial understanding and reasoning capabilities. All these data are combined into a total of 1.84M training samples for co-training, with all visual information represented in RGB-D format. The acquisition of depth information is detailed in the following section.

Depth Information Acquisition. Depth information is easier to obtain compared to other 3D representations, making it more feasible to meet the data volume and diversity requirements of VLA models. In simulation environments, we can obtain depth information with ground-truth accuracy, but this is almost impossible in the real world as depth data collected in the real world inevitably contains noise due to hardware limitations. For example, captured depth images often exhibit inaccuracies at object boundaries and may fail entirely when dealing with transparent or specular objects. Prior works [3, 31, 68] that directly use ground-truth depth in simulators have encountered significant sim-to-real gaps due to the discrepancies in depth distributions between simulation and real-world environments. Besides, the internet data used for co-training does not include depth information originally. This suggests that while using collected depth data may achieve high accuracy in simulations, it introduces challenges in bridging the sim-to-real gap and improving generalization with different data sources. In this work, we conduct extensive experiments to compare these two methods for acquiring depth, *i.e.*, capturing vs. estimation, with respect to the final end-to-end performance and further study the best practice of injecting depth information in existing vision foundation models. Details are in the experiment section.

E. Implementation details

NaVid-4D is trained on 4 NVIDIA H800 GPUs for 3 days in the first two stages, and 8 NVIDIA H800 GPUs for 2.5 days in the last two stages. To improve obstacle avoidance performance, we adopt A* algorithm [25] to refine training trajectories wherein the distances between the robot and obstacles are considered in the cost function. Following NaVid [72], non-navigation video data is sampled at 1 FPS, while all frames are retained for navigation data. At each time step, the agent's action granularity is set to move forward for 25 cm, turn left/right for 15°, or decide to stop, which empirically ensures smooth and continuous movement in real-world experiments. We initialize Q-Former [16], BERT [17], and Vicuna-7B [15] using their default pre-trained weights and initialize all layers of the vision encoder with the pre-trained weights of EVA-CLIP [64]. Additionally, we initialize the projector with the corresponding weights of LLaMA-VID [43] trained in its first stage. For evaluation, observation images are transferred to a computer equipped with an NVIDIA GeForce RTX 3090 using ROS2 [46], and actions are extracted from the model's output using regular expression matching [36]. In the real-world experiments, Metric3Dv2 [30] takes 1.3s to estimate depth, and our model takes 2.9s to generate 4 actions at each time step.

IV. EXPERIMENTS

In this section, we conduct experiments to study the following essential questions for evaluating NaVid-4D:

- (1) How well does NaVid-4D perform against state-of-the-art VLN models?
- (2) How much benefit does the explicit use of depth information bring to the VLN task?
- (3) What is the proper way to inject depth features into existing vision foundational models?
- (4) What impact do different methods of obtaining depth have on the performance of NaVid-4D?
- (5) Is predicting multiple steps at each inference time effective for performance improvement?
- (6) Is NaVid-4D practical for real-world deployment?

A. Experimental Setup

Simulation Experiments. We conduct experiments on the VLN-CE benchmark, which offers 16,844 path-instruction pairs over 90 visually realistic scenes in the Matterport3D [7] dataset. For a fair comparison, all methods are trained on the training split containing 10,819 Room-to-Room (R2R) [39] samples and evaluated on the test split with 1,839 R2R val-unseen samples. To reduce the computation cost, we adopt DAgger only in Table I and adopt SpatialQA data only in Table I and Table V. Regarding the evaluation, we follow the standard VLN evaluation metrics [21, 39, 71] to report results, including success rate (SR), oracle success rate (OS), success weighted by path length (SPL), trajectory length (TL), and navigation error from goal (NE). An episode is considered a success if the STOP decision is taken within 3m of the goal position.

Real-world Experiments. For evaluation in real-world environments, we use the Hexman Echo Plus as the robot base, equipped with a Kinect DK camera to capture RGB images, as shown in Fig. 4. To show the advantage of using depth information in normal navigation tasks, we conduct instruction-following experiments in diverse real-world environment scenes. In the instruction-following experiments, an episode is considered a success if the STOP decision is taken within 1.5m of the goal position. We also evaluate spatial reasoning capabilities by conducting experiments on the 5 categories shown in Fig. 1 separately, and the episode is considered a success if it reaches the target.

B. Quantitative Results

We study the questions (1) and (2) by quantitatively comparing NaVid-4D with representative RGB and RGB-D-based state-of-the-art baselines as follows: **Seq2Seq** [39]: Employs a recurrent policy to predict actions directly from RGB-D observations. **CMA** [39]: Utilizes cross-modal attention between instructions and RGB-D observations to predict action. **WS-MGMap** [10]: Leverages a multi-granularity map that incorporates object geometry, texture, and semantic information. **NaVid** [72]: Video-based large vision language navigation model with only RGB input.

The results in Table I show that NaVid-4D outperforms all baselines by a large margin in simulation, demonstrating its strong capability in this task. Compared to its predecessor NaVid, NaVid-4D is superior on all evaluation metrics, particularly achieving 17.1% improvement in SR. This clearly demonstrates the advantages of explicitly using depth information in VLN models, which makes it easier to find the landmarks and follow the instructions correctly. Note that VLN-CE R2R contains ambiguous descriptions regarding initial orientation [41], and we find our method can achieve significant performance enhancement when the initial orientation is properly aligned with instructions, showing a remarkable improvement of 31.9% in Success Rate (SR).

	TL	NE↓	OS↑	SR↑	SPL↑
Seq2Seq [39]	9.30	7.77	37.0	25.0	22.0
CMA [39]	8.64	7.37	40.0	32.0	30.0
WS-MGMap [10]	10.00	6.28	47.6	38.9	34.3
NaVid [72]	7.63	5.47	49.1	37.4	35.9
NaVid-4D (ours)	11.91	5.99	55.7	43.8	37.1
NaVid-4D (w. aligned)	9.32	**3.85**	**68.1**	**57.8**	**53.0**

TABLE I: **Comparing on VLN-CE R2R Val-Unseen.**

	TL	NE↓	OS↑	SR↑	SPL↑
w/o ViT	9.65	4.88	58.1	44.2	39.6
with ViT	9.81	4.62	60.9	47.3	41.9
Ours	8.84	**4.32**	**61.0**	**51.4**	**47.6**

TABLE II: **Ablation study on the depth encoding method.** In "w/o ViT", we adopt MLP layers to encode flattened depth patches. In "with ViT", we adopt non-shared ViT models for encoding RGB and depth separately.

Num. of shared layers	TL	NE↓	OS↑	SR↑	SPL↑
0 (non-shared ViT)	9.81	4.62	60.9	47.3	41.9
10	9.55	4.34	**63.9**	51.0	45.8
20 (Recommended)	8.84	**4.32**	61.0	**51.4**	**47.6**
30	9.14	4.40	61.2	50.2	45.8

TABLE III: **Ablation study on the position of depth feature injection.** The vision encoder consists of 40 layers in total. This experiment studies how varying the number of shared layers affects the performance of NaVid-4D.

Train	Test	TL	NE↓	OS↑	SR↑	SPL↑
w/o	w/o	9.68	4.72	59.0	45.9	41.1
captured	captured	9.03	4.34	60.5	51.1	46.5
captured	estimated	9.01	4.51	59.7	49.9	45.4
estimated	captured	9.28	4.36	**63.5**	**51.7**	47.0
estimated	estimated	8.84	**4.32**	61.0	51.4	**47.6**

TABLE IV: **Ablation study on the depth sources of navigation data.** In all experiments involving depth input, the co-training data use estimated depth, as only estimated depth is available.

Ablation Studies. For questions (3), (4), and (5), we conduct ablation studies to compare different alternatives for model design and configuration.

For question (3), we compare different depth encoding strategies in Table II. In the experiment "w/o ViT", we use an MLP to encode flattened depth patches, resulting in clearly worse performance than "with ViT". This result demonstrates that the ViT weights pre-trained on RGB could serve as a reasonable initialization of a depth encoder to facilitate the alignment between RGB and depth information. Considering that RGB and depth are correlated but possess different characteristics, we propose to adopt non-shared weights in the shallow layers of the ViT to learn to represent

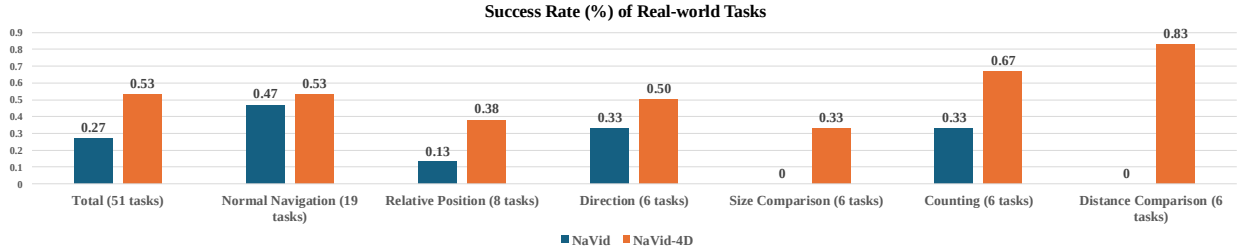

Fig. 3: **Real-world Experiment Results.** The result shows the comparison between NaVid [72] and our method in both instruction-following tasks and spatial intelligence tasks.

	TL	NE↓	OS↑	SR↑	SPL↑
Ours	8.84	4.32	61.0	51.4	47.6
Ours (with SpatialQA [6])	9.25	**4.16**	**64.4**	**53.8**	**49.3**

TABLE V: **Ablation study on using SpatialQA [6] as co-training data.** DAgger is not used here to reduce the training cost.

Num. of steps	TL	NE↓	OS↑	SR↑	SPL↑
1	9.87	4.88	60.0	44.1	38.7
2	9.80	4.50	**63.5**	48.9	43.5
4 (Recommended)	8.84	**4.32**	61.0	**51.4**	**47.6**
8	8.95	4.34	59.8	50.5	46.5

TABLE VI: **Ablation study on the number of steps in each action chunk.**

their features in a unified space. Thus, we further conduct an ablation study on the number of shared layers, *i.e.*, the position of injecting depth features, in Table III. The results indicate that the optimal choice seems to share the first 20 layers of ViT, meaning that depth information should be injected at the midpoint of the ViT.

For question (4), we rigorously evaluate the benefits of explicitly using depth and compare different methods of acquiring depth, *i.e.*, captured vs. estimated, in Table IV. The results significantly demonstrate the necessity of explicitly using depth. Moreover, using estimated depth for training is advantageous even though the captured depth in simulators has ground-truth accuracy. This is because the co-training data includes only estimated depth, so the disparity between captured depth in navigation data and estimated depth in co-training data negatively affects performance. Moreover, when adopting estimated depth during training, we observe that estimated depth and captured depth deliver comparable performance in terms of evaluation performance. As for real-world applications, using estimated depth is better as well due to the noisy captured depth images. Thus, we recommend using estimated depth in both simulation and real-world scenarios. Besides, we also conduct an ablation experiment on the impact of using SpatialQA [6] data in Table V. The results show that SpatialQA [6] data enhances performance in R2R by helping the model better extract and interpret depth information with its depth-related data.

For question (5), we conduct ablations on the number of steps in action chunking in Table VI. The results indicate that predicting 4 steps at each inference yields the highest performance. This is because predicting multiple steps jointly encourages the model to learn longer-horizon knowledge, while predicting too many steps becomes more difficult for the model to handle. Additionally, chunking multiple steps

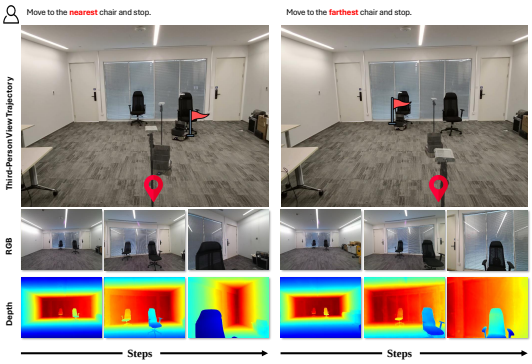

Fig. 4: **Real-world results.** From top to bottom are the third-person view trajectory, first-person view RGB images, and estimated depth.

reduces inference costs, making 4-step action chunking a good trade-off between performance and efficiency.

Real-world Results. We compare NaVid-4D with the SOTA baseline, NaVid [72], on real-world tasks, including instruction-following and spatial intelligence evaluations. For the instruction-following evaluation, as shown in Fig. 3, the results demonstrate that our model is comparable to NaVid [72] in real-world instruction-following tasks. Furthermore, we conducted a case study on the five categories of spatial intelligence abilities, as depicted in Fig. 1, with the results shown in Fig. 3. The findings indicate that our model with estimated depth as input significantly surpasses NaVid [72] in performance across those categories.

V. Conclusion

We introduce NaVid-4D, a VLM-based navigation agent capable of end-to-end generation of low-level actions following instructions with an egocentric RGB-D video stream. In NaVid-4D, we propose a novel model paradigm and training strategy to explicitly encode and exploit depth information. Thanks to this, NaVid-4D especially enhances the capability of spatial understanding and reasoning. The experimental results show improvements in both simulation and real-world settings for standard navigation tasks as well as those requiring spatial reasoning. Currently, NaVid-4D still faces challenges in long-range tasks due to the high GPU memory consumption of encoding depth information for historical frames. Furthermore, converting observations into point clouds, which more effectively capture 3D features, may offer the potential to improve the model's capabilities. In future work, we plan to explore these approaches for further improvements.

References

[1] J. Achiam *et al.*, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[2] P. Anderson *et al.*, "Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 3674–3683.

[3] P. Anderson *et al.*, "Sim-to-real transfer for vision-and-language navigation," in *Conference on Robot Learning*, PMLR, 2021, pp. 671–681.

[4] S. Banerjee, J. Thomason, and J. Corso, "The robotslang benchmark: Dialog-guided robot localization and navigation," in *Conference on Robot Learning*, PMLR, 2021, pp. 1384–1393.

[5] F. Caba Heilbron, V. Escorcia, B. Ghanem, and J. Carlos Niebles, "Activitynet: A large-scale video benchmark for human activity understanding," in *Proceedings of the ieee conference on computer vision and pattern recognition*, 2015, pp. 961–970.

[6] W. Cai *et al.*, "Spatialbot: Precise spatial understanding with vision language models," *arXiv preprint arXiv:2406.13642*, 2024.

[7] A. Chang *et al.*, "Matterport3d: Learning from rgb-d data in indoor environments," *arXiv preprint arXiv:1709.06158*, 2017.

[8] B. Chen *et al.*, "Spatialvlm: Endowing vision-language models with spatial reasoning capabilities," *arXiv preprint arXiv:2401.12168*, 2024.

[9] K. Chen, J. K. Chen, J. Chuang, M. Vázquez, and S. Savarese, "Topological planning with transformers for vision-and-language navigation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 11 276–11 286.

[10] P. Chen *et al.*, "Weakly-supervised multi-granularity map learning for vision-and-language navigation," *Advances in Neural Information Processing Systems*, vol. 35, pp. 38 149–38 161, 2022.

[11] P. Chen *et al.*, "A2 Nav: Action-aware zero-shot robot navigation by exploiting vision-and-language ability of foundation models," *arXiv preprint arXiv:2308.07997*, 2023.

[12] S. Chen, P.-L. Guhur, C. Schmid, and I. Laptev, "History aware multimodal transformer for vision-and-language navigation," *Advances in neural information processing systems*, vol. 34, pp. 5834–5847, 2021.

[13] S. Chen, P.-L. Guhur, M. Tapaswi, C. Schmid, and I. Laptev, "Think global, act local: Dual-scale graph transformer for vision-and-language navigation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 537–16 547.

[14] A.-C. Cheng *et al.*, "Spatialrgpt: Grounded spatial reasoning in vision language model," *arXiv preprint arXiv:2406.01584*, 2024.

[15] W.-L. Chiang *et al.*, "Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality," *See https://vicuna. lmsys. org (accessed 14 April 2023)*, vol. 2, no. 3, p. 6, 2023.

[16] W. Dai *et al.*, "Instructblip: Towards general-purpose vision-language models with instruction tuning," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[17] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[18] X. Fu *et al.*, "Blink: Multimodal large language models can see but not perceive," *arXiv preprint arXiv:2404.12390*, 2024.

[19] C. Gan *et al.*, "Threedworld: A platform for interactive multi-modal physical simulation," *arXiv preprint arXiv:2007.04954*, 2020.

[20] X. Gao, Q. Gao, R. Gong, K. Lin, G. Thattai, and G. S. Sukhatme, "Dialfred: Dialogue-enabled agents for embodied instruction following," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 10 049–10 056, 2022.

[21] G. Georgakis *et al.*, "Cross-modal map learning for vision and language navigation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 15 460–15 470.

[22] Y. Goyal, T. Khot, D. Summers-Stay, D. Batra, and D. Parikh, "Making the v in vqa matter: Elevating the role of image understanding in visual question answering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 6904–6913.

[23] J. Gu, E. Stefani, Q. Wu, J. Thomason, and X. E. Wang, "Vision-and-language navigation: A survey of tasks, methods, and future directions," *arXiv preprint arXiv:2203.12667*, 2022.

[24] Q. Gu *et al.*, "Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2024, pp. 5021–5028.

[25] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.

[26] M. He *et al.*, "Efficient multimodal learning from data-centric perspective," *arXiv preprint arXiv:2402.11530*, 2024.

[27] Y. Hong, Z. Wang, Q. Wu, and S. Gould, "Bridging the gap between learning in discrete and continuous environments for vision-and-language navigation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 15 439–15 449.

[28] Y. Hong, Q. Wu, Y. Qi, C. Rodriguez-Opazo, and S. Gould, "A recurrent vision-and-language bert for navigation," *arXiv preprint arXiv:2011.13922*, 2020.

[29] Y. Hong *et al.*, "3d-llm: Injecting the 3d world into large language models," *Advances in Neural Information Processing Systems*, vol. 36, pp. 20 482–20 494, 2023.

[30] M. Hu *et al.*, "Metric3d v2: A versatile monocular geometric foundation model for zero-shot metric depth and surface normal estimation," *arXiv preprint arXiv:2404.15506*, 2024.

[31] 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)*, IEEE, 2023, pp. 10 608–10 615.

[32] J. Huang *et al.*, "An embodied generalist agent in 3d world," *arXiv preprint arXiv:2311.12871*, 2023.

[33] D. A. Hudson and C. D. Manning, "Gqa: A new dataset for real-world visual reasoning and compositional question answering," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 6700–6709.

[34] S. Kazemzadeh, V. Ordonez, M. Matten, and T. Berg, "Referitgame: Referring to objects in photographs of natural scenes," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 787–798.

[35] L. Ke *et al.*, "Tactical rewind: Self-correction via backtracking in vision-and-language navigation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 6741–6749.

[36] S. M. Kearns, "Extending regular expressions with context operators and parse extraction," *Software: Practice and Experience*, vol. 21, no. 8, pp. 787–804, 1991.

[37] J. Krantz, A. Gokaslan, D. Batra, S. Lee, and O. Maksymets, "Waypoint models for instruction-guided navigation in continuous environments," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 15 162–15 171.

[38] J. Krantz and S. Lee, "Sim-2-sim transfer for vision-and-language navigation in continuous environments," in *European Conference on Computer Vision*, Springer, 2022, pp. 588–603.

[39] J. Krantz, E. Wijmans, A. Majumdar, D. Batra, and S. Lee, "Beyond the nav-graph: Vision-and-language navigation in continuous environments," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVIII 16*, Springer, 2020, pp. 104–120.

[40] J. Krantz *et al.*, "Iterative vision-and-language navigation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 14 921–14 930.

[41] A. Ku, P. Anderson, R. Patel, E. Ie, and J. Baldridge, "Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding," *arXiv preprint arXiv:2010.07954*, 2020.

[42] L. H. Li, M. Yatskar, D. Yin, C. Hsieh, and K. Chang, "Visualbert: A simple and performant baseline for vision and language. arxiv 2019," *arXiv preprint arXiv:1908.03557*, vol. 3, 1908.

[43] Y. Li, C. Wang, and J. Jia, "Llama-vid: An image is worth 2 tokens in large language models," *arXiv preprint arXiv:2311.17043*, 2023.

[44] H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Visual instruction tuning," *Advances in neural information processing systems*, vol. 36, 2024.

[45] C.-Y. Ma *et al.*, "Self-monitoring navigation agent via auxiliary progress estimation," *arXiv preprint arXiv:1901.03035*, 2019.

[46] S. Macenski, T. Foote, B. Gerkey, C. Lalancette, and W. Woodall, "Robot operating system 2: Design, architecture, and uses in the wild," *Science Robotics*, vol. 7, no. 66, eabm6074, 2022.

[47] A. Majumdar *et al.*, "Openeqa: Embodied question answering in the era of foundation models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 16 488–16 498.

[48] J. Mao, J. Huang, A. Toshev, O. Camburu, A. L. Yuille, and K. Murphy, "Generation and comprehension of unambiguous object descriptions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 11–20.

[49] A. Mishra, S. Shekhar, A. K. Singh, and A. Chakraborty, "Ocr-vqa: Visual question answering by reading text in images," in *2019 international conference on document analysis and recognition (ICDAR)*, IEEE, 2019, pp. 947–952.

[50] A. Moudgil, A. Majumdar, H. Agrawal, S. Lee, and D. Batra, "Soat: A scene-and object-aware transformer for vision-and-language navigation," *Advances in Neural Information Processing Systems*, vol. 34, pp. 7357–7367, 2021.

[51] B. Pan *et al.*, "Langnav: Language as a perceptual representation for navigation," *arXiv preprint arXiv:2310.07889*, 2023.

[52] Y. Qi, Z. Pan, Y. Hong, M.-H. Yang, A. Van Den Hengel, and Q. Wu, "The road to know-where: An object-and-room informed sequential bert for indoor vision-language navigation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 1655–1664.

[53] Y. Qi, Z. Pan, S. Zhang, A. van den Hengel, and Q. Wu, "Object-and-action aware model for visual language navigation," in *European Conference on Computer Vision*, Springer, 2020, pp. 303–317.

[54] Y. Qi *et al.*, "Reverie: Remote embodied visual referring expression in real indoor environments," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9982–9991.

[55] A. Radford *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*, PMLR, 2021, pp. 8748–8763.

[56] S. Raychaudhuri, S. Wani, S. Patel, U. Jain, and A. X. Chang, "Language-aligned waypoint (law) supervision for vision-and-language navigation in continuous environments," *arXiv preprint arXiv:2109.15207*, 2021.

[57] S. Ross, G. Gordon, and D. Bagnell, "A reduction of imitation learning and structured prediction to no-regret online learning," in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, JMLR Workshop and Conference Proceedings, 2011, pp. 627–635.

[58] M. Savva *et al.*, "Habitat: A platform for embodied ai research," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 9339–9347.

[59] D. Schwenk, A. Khandelwal, C. Clark, K. Marino, and R. Mottaghi, "A-okvqa: A benchmark for visual question answering using world knowledge," in *European conference on computer vision*, Springer, 2022, pp. 146–162.

[60] D. Shah, B. Osiński, S. Levine, *et al.*, "Lm-nav: Robotic navigation with large pre-trained models of language, vision, and action," in *Conference on robot learning*, PMLR, 2023, pp. 492–504.

[61] *Sharegpt*, `https://sharegpt.com/`, 2023.

[62] M. Shridhar *et al.*, "Alfred: A benchmark for interpreting grounded instructions for everyday tasks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 10 740–10 749.

[63] O. Sidorov, R. Hu, M. Rohrbach, and A. Singh, "Textcaps: A dataset for image captioning with reading comprehension," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, Springer, 2020, pp. 742–758.

[64] Q. Sun, Y. Fang, L. Wu, X. Wang, and Y. Cao, "Eva-clip: Improved training techniques for clip at scale," *arXiv preprint arXiv:2303.15389*, 2023.

[65] J. Thomason, M. Murray, M. Cakmak, and L. Zettlemoyer, "Vision-and-dialog navigation," in *Conference on Robot Learning*, PMLR, 2020, pp. 394–406.

[66] X. Wang *et al.*, "Reinforced cross-modal matching and self-supervised imitation learning for vision-language navigation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 6629–6638.

[67] Z. Wang, H. Huang, Y. Zhao, Z. Zhang, and Z. Zhao, "Chat-3d: Data-efficiently tuning large language model for universal dialogue of 3d scenes," *arXiv preprint arXiv:2308.08769*, 2023.

[68] Z. Wang, X. Li, J. Yang, S. Jiang, *et al.*, "Sim-to-real transfer via 3d feature fields for vision-and-language navigation," *arXiv preprint arXiv:2406.09798*, 2024.

[69] Z. Wang *et al.*, "Scaling data generation in vision-and-language navigation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 12 009–12 020.

[70] J. Zhang, M. Cai, T. Xie, and Y. J. Lee, "Countercurate: Enhancing physical and semantic visio-linguistic compositional reasoning via counterfactual examples," *arXiv preprint arXiv:2402.13254*, 2024.

[71] J. Zhang *et al.*, "3d-aware object goal navigation via simultaneous exploration and identification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 6672–6682.

[72] J. Zhang *et al.*, "Navid: Video-based vlm plans the next step for vision-and-language navigation," *arXiv preprint arXiv:2402.15852*, 2024.

[73] G. Zhou, Y. Hong, and Q. Wu, "Navgpt: Explicit reasoning in vision-and-language navigation with large language models," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, 2024, pp. 7641–7649.