2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

InSpire: Vision-Language-Action Models with Intrinsic Spatial Reasoning

Ji Zhang[1*] Shihan Wu[2*] Xu Luo[2] Hao Wu[2] Junlin Xie[2]
Lianli Gao[2] Heng Tao Shen[3] Jingkuan Song[3†]
[1]Southwest Jiaotong University
[2]University of Electronic Science and Technology of China
[3]Tongji University
{jizhang.jim,jingkuan.song}@gmail.com
https://koorye.github.io/Inspire

***Abstract*— Leveraging pretrained Vision-Language Models (VLMs) to map language instruction and visual observations to raw low-level actions, Vision-Language-Action models (VLAs) hold great promise for achieving general-purpose robotic systems. Despite their advancements, existing VLAs tend to spuriously correlate task-irrelevant visual features with actions, limiting their generalization capacity beyond the training data. To tackle this challenge, we propose Intrinsic Spatial Reasoning (InSpire), a simple yet effective approach that mitigates the adverse effects of spurious correlations by boosting the spatial reasoning ability of VLAs. Specifically, InSpire redirects the VLA's attention to task-relevant factors by prepending the question "In which direction is the [object] relative to the robot?" to the language instruction and aligning the model's output answer "right/left/up/down/front/back/grasped" and predicted actions with ground-truth. Notably, InSpire can be used as a *plugin* to enhance existing autoregressive VLAs, requiring no extra training data or interaction with other large models. Extensive experimental results in both simulation and real-world environments demonstrate the effectiveness and flexibility of InSpire.**

I. Introduction

In recent years, Vision-Language Models (VLMs) [35] have demonstrated remarkable abilities across a diverse set of tasks, including image captioning and visual question answering (VQA). These advances have paved the way for Vision-Language-Action (VLA) models, which use pretrained VLMs to directly map language commands and visual inputs to low-level motor actions, offering a promising pathway toward general-purpose robotic systems [5], [15], [23].

Despite their advancements, state-of-the-art VLAs tend to spuriously correlate task-irrelevant visual features with actions, overlooking vital elements such as language instructions and spatial relations in visual observations, which hinders their ability to generalize beyond the training data distribution[31]. As shown in Fig. 1 **(a)**, the VLA—trained via direct observation-to-action mapping—fails to accurately identify task-specific objects and model their spatial relationships with the robot. Instead, it allocates more attention to task-irrelevant regions when predicting actions. Understanding and reasoning about spatial relationships are crucial skills that empower humans to solve complex tasks: if asked the same question, they would first try to assess the directions

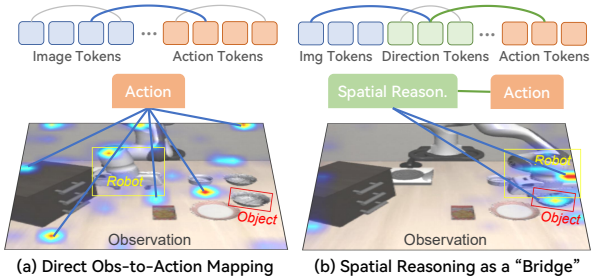

Fig. 1: **(a)** VLAs typically predict actions relying on *Spurious Correlations* learned by the direct observation-to-action mapping mechanism. **(b)** The core idea of our proposed InSpire method that tackles spurious correlations by boosting the spatial reasoning capabilities of VLAs.

(or coarse locations) of the "black bowl" and the "plate" relative to their hands. This underscores the importance of leveraging spatial reasoning as a bridge to capture the causal relationships between observations and actions, thereby allowing VLAs to produce more accurate and robust robot actions, as illustrated in Fig. 1 **(b)**. Prior efforts generally resort to auxiliary training data [39] or other large models [32], [25] to enhance chain-of-thought (or step-by-step) reasoning capabilities. While these approaches offer partial improvements in spatial reasoning, they often suffer from limitations in efficiency and generalizability. We thus raise the following question:

> *Without relying on extra data or interacting with other large models, can we enhance the spatial reasoning capabilities of VLAs to resolve spurious correlations?*

Our solution the question is **Intrinsic Spatial Reasoning (InSpire)**, a simple yet effective approach that boosts VLAs' spatial reasoning capabilities to mitigate the adverse effects of spurious correlations on their generalization in novel scenarios. As presented in Fig. 2, our proposed InSpire approach redirects the model's attention from spurious factors to task-relevant ones by simply appending the question "In which direction is the [object] relative to the robot?" before the language instruction and aligning the VLA's generated answer "right/left/up/down/front/back/grasped" and predicted actions with the ground-truth. By using this

* Equal contribution
† Corresponding author

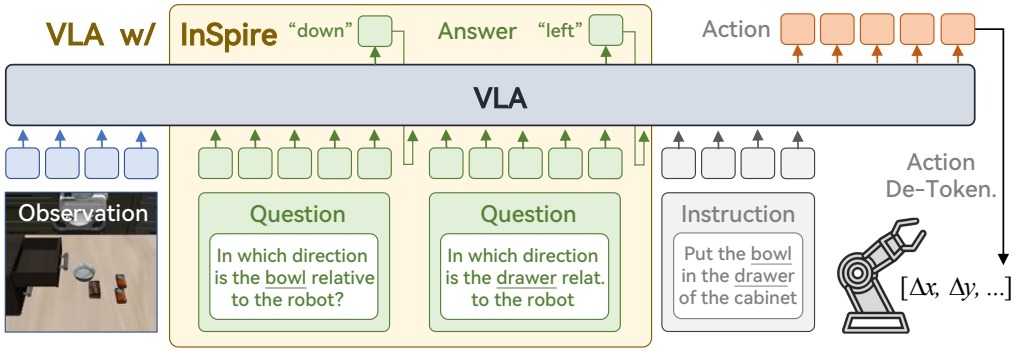

Fig. 2: **Overview of our InSpire approach**. InSpire boosts the VLA's spatial reasoning ability by appending the question "In which direction is the [object] relative to the robot?" before the language instruction and aligning the VLA's answer "right/left/up/down/front/back/grasped" and predicted actions with the ground-truth.

spatial reasoning VQA task as the bridging "language" between observations and actions, InSpire equips VLAs with the ability to understand and reason about spatial relationships without the need to collect auxiliary training data or interact with other large models. Notably, the InSpire approach is fully compatible with existing autoregressive VLAs and can be seamlessly integrated as a ***plugin*** to enhance their performance. We conduct extensive evaluations using both simulation benchmarks (LIBERO [19] and CALVIN [24]) and real-world tasks, the achieved results on two state-of-the-art VLAs—miniVLA-VQ[3] (a lightweight version of OpenVLA[15]) and π_0-FAST [28] (an upgraded version of π_0[4]), prove InSpire's effectiveness and flexibility.

To summarize, our contributions in this work are threefold.

- We propose InSpire, a novel approach designed to mitigate the negative impact of spurious correlations on the generalization performance of VLAs.
- Without employing extra data or interacting with other large models, InSpire endows VLAs with spatial reasoning capabilities in a plug-and-play manner.
- Comprehensive evaluations in both simulation and real-world environments demonstrate the effectiveness and flexibility of the proposed InSpire approach.

II. Proposed Approach

A. Preliminaries and Problem Statement

In the standard supervised or imitation learning framework, we consider an expert demonstration dataset $D = \{(o_i, l_i, a_i)\}_{i=1}^{N}$, where $o_i \in \mathcal{O}$ denotes an observation, $l_i \in \mathcal{L}$ a language instruction, and $a_i \in \mathcal{A} \subset \mathbb{R}^m$ an action. Each datapoint is sampled from the training distribution $p_{train}(o, l, a) = p(a|o, l)p_{train}(o, l)$, where the ground-truth expert policy $p(a|o, l)$ is independent of the specific choice of training distribution. The aim is to learn an action policy $\pi_\theta(a|o, l)$ that approximates $p(a|o, l)$. In this work, $\pi_\theta(a|o, l)$ are autoregressive VLAs, typically expressive neural networks like transformers pretrained from VLMs, which directly map visual observations o and language instructions l to action tokens a in an autoregressive manner [15], [5], [6], [28].

Following [11], we assume that the combined input $[o, l]$ is generated from underlying "observation factors". These factors are categorized into task-relevant factors $u \in \mathcal{U}$, which causally determine the action, and task-irrelevant factors $v \in \mathcal{V}$, which have no causal effect on the action. Consequently, the expert policy depends only on task-relevant factors, $p(a|o, l) = p(a|u)$, implying that in the true data-generating process, actions a are independent of task-irrelevant factors v (i.e., $p(a, v) = p(a)p(v)$). Spurious correlations arise when a and v become statistically dependent within the training distribution, such that $p_{train}(a, v) \neq p_{train}(a)p_{train}(v)$. This often occurs if the task-relevant factors u and task-irrelevant factors v are themselves correlated in the training data ($p_{train}(u, v) \neq p_{train}(u)p_{train}(v)$). Given the causal relationship $u \to a$, a correlation between u and v under p_{train} can induce a non-causal statistical association between v and a. Learned policies relying on such spurious correlations exhibit poor generalization to distributions beyond p_{train}, leading to unreliable performance in novel scenarios.

B. Intrinsic Spatial Reasoning

We hypothesize that models resort to spurious correlations when task-relevant factors are not explicit in the input. Such latent factors are difficult to learn, as neural networks preferentially learn simpler patterns first [1]. For instance, in a task such as "put the bowl in the drawer of the cabinet" (see Fig. 2), the small size of the bowl might lead the model to infer actions from irrelevant distractors or background elements rather than the object of interest. The central insight behind our method is to enable the model to first extract salient, task-relevant information from observations—a simpler pattern to discern—and then utilize this information as an additional input for action generation.

*1) **Method Overview**:* Formally, we denote $u' = f(u)$ as an extracted representation that processes and summarizes high-level information from the task-relevant factors u. The goal is to learn two policies: an extraction policy $\pi_{u'} : \mathcal{O} \times \mathcal{L} \to \mathcal{U}'$ which maps observation and task instruction to the extracted task-relevant representations, and an action policy $\pi_\theta : \mathcal{O} \times \mathcal{L} \times \mathcal{U}' \to \mathcal{A}$ which maps observation, task

instruction, and the extracted task-relevant representations to actions. Consequently, the model outputs actions via a two-step process instead of a single step:

$$u' = \pi_{u'}(o, l), \;\; a = \pi_\theta(u', o, l).$$

The extracted representation u' is designed to be more readily learnable, thereby guiding the model away from spurious correlations present in the original observations o.

*2) **Modeling Task-relevant Factors via Spatial Reasoning VQA**:* To establish a reliable representation u' of task-relevant factors, we use the extensive, text-based world knowledge inherent in VLMs—the models from which contemporary VLAs are pretrained. Exploiting text as this representation u', we propose Intrinsic Spatial Reasoning (InSpire) that performs explicit spatial reasoning regarding the robot and objects of interest prior to the inference of subsequent actions.

As shown in Fig. 2, we employ a VLA that dually functions as an extraction policy $\pi_{u'}$ and an action policy π_θ. Given an initial language instruction l, InSpire first introduces a textual question q to probe the spatial relationships between objects mentioned in l and the robot, e.g., "In which direction is the [object] relative to the robot?". Object names are identified within l using the natural language toolkit [21]. The VLA, operating as the extraction policy $\pi_{u'}$, takes q, the current visual observation o, and the language instruction l as input to produce a textual answer g, which is constrained to a predefined set of valid options of coarse-grained directions, as detailed in Fig. 3. The combination of the self-generated question and its corresponding answer constitutes an extracted textual representation $u' = [q, g]$. This representation u' is subsequently passed to the same VLA, now serving as the action policy π_θ, which outputs the final robot actions. Because this spatial reasoning VQA task is specifically formulated to correlate highly with objects of interest and the requisite actions, the resulting textual pair u' provides a distilled, abstract representation of task-relevant factors, simplifies the learning challenge for the action policy, and mitigates the learning of spurious correlations.

*3) **Automated Rule-based Object Direction Labeling**:* To facilitate model training, ground-truth spatial relationships are integrated into the training datasets. Each training datapoint is thus augmented to (o_i, a_i, u'_i), where u'_i encompasses self-generated questions and their corresponding ground-truth spatial relationship answers. These ground-truth relationships are derived by first determining the positions of the robot's end-effector/gripper and relevant objects, then inferring their relative spatial arrangement, as illustrated in Fig. 3. In simulations, object locations are readily available. In real-world settings, the end-effector's position at the instant of gripper closure/opening is recorded and used as a proxy for the object's position when establishing these relationships. Given an end-effector position $[x_i, y_i, z_i]$ and an object position $[x_0, y_0, z_0]$, the position difference is calculated as $\boldsymbol{d} = [x_i - x_0, y_i - y_0, z_i - z_0]$. The coarse-grained spatial relationship is then determined by the axis corresponding to the component of $\boldsymbol{d}$ with the largest absolute value. For instance, if $|x_i - x_0|$ is the maximum component magnitude in $\boldsymbol{d}$, the relationship is classified as "left" or "right" based on the sign of $x_i - x_0$. When the object is grasped, as indicated by gripper closure, it is directly labeled with the text "grasped". This rule-based strategy, applied from a third-person viewpoint, generates the ground-truth labels used for supervising the VLA's spatial reasoning. During training, the autoregressive loss is also applied to tokens corresponding to spatial reasoning answers, alongside the primary action tokens; this additional loss encourages the model to predict the correct textual descriptions of these spatial relationships. Our experiments show that this relatively coarse-grained spatial reasoning is nonetheless effective for guiding the action generation process (see Section III-D).

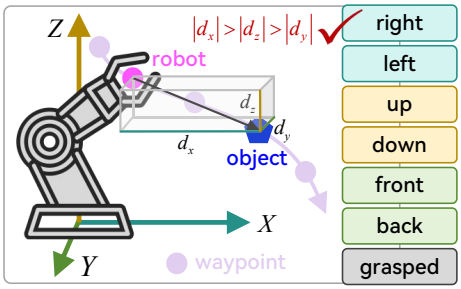

Fig. 3: **Automated rule-based object direction labeling.** At each waypoint of a trajectory, the 3D locations of the robot's gripper and target objects are obtained from the simulation environment or recorded positions where the robot interacts with objects in the real-world environment. These locations are used in a rule-based strategy to automatically compute the object's direction.

III. Experiments

In this section, we conduct extensive experiments to answer the following questions:

1) Can InSpire enhance VLAs on tasks from both simulation and real-world environments?
2) Can InSpire surpass other reasoning-based methods in improving VLAs?
3) What is the impact of various design decisions on InSpire's performance?
4) How does InSpire help resolve spurious correlations?

We answer the first question in Sections III-A and III-B, the second question in Section III-C, the third question in Section III-D, and the fourth question in Section III-E.

A. Simulation Experiments

We perform simulation experiments using LIBERO [19] and CALVIN [24] benchmarks. We train the model with random 3 seeds, and conduct 100 trials per task at inference.

*1) **LIBERO Evaluation**:* We use manipulation tasks from the five datasets—LIBERO-90, LIBERO-Spatial, LIBERO-Object, LIBERO-Goal and LIBERO-Long/10. The VLA model is jointly trained on the 90 tasks in the LIBERO-90 dataset, with each task consisting of 50 demonstrations.

TABLE I: **LIBERO Performance**. * models borrowed from [3]. † our models pretrained from scratch on LIBERO-90. ‡ our models finetuned on LIBERO-90.

Model	Seen	Unseen				
	LIBERO-90	-Spatial	-Object	-Goal	-Long	**Average**
OpenVLA [15]	61.4	-	-	-	-	-
miniVLA* [3]	62.0	0	0	0	1	0.25
miniVLA-VQ* [3]	77.0	0	0	3	1	1
miniVLA-hist.*[3]	82.0	0	1	2	7	2.5
miniVLA-wrist*[3]	82.1	0	0	0	0	0
SpatialVLA‡[38]	$46.2_{\pm 1.7}$	0	0	0.67	1.33	$0.5_{\pm 0.5}$
miniVLA-VQ† [3]	$83.3_{\pm 1.2}$	0.7	0	5.7	8.0	$3.6_{\pm 0.4}$
+InSpire (*Ours*)	**89.5**$_{\pm 1.5}$	**13.7**	**20.1**	**12.7**	8.0	**13.6**$_{\pm 3.9}$
Δ	+6.2	+13.0	+20.1	+7.0	+0	+10.0
π_0-FAST‡ [28]	$83.1_{\pm 1.0}$	5.7	6.0	6.0	5.0	$5.7_{\pm 0.8}$
+InSpire (*Ours*)	**84.1**$_{\pm 1.0}$	**7.3**	**15.7**	6.0	5.0	**8.5**$_{\pm 1.3}$
Δ	+1.0	+1.6	+9.7	+0	+0	+2.8

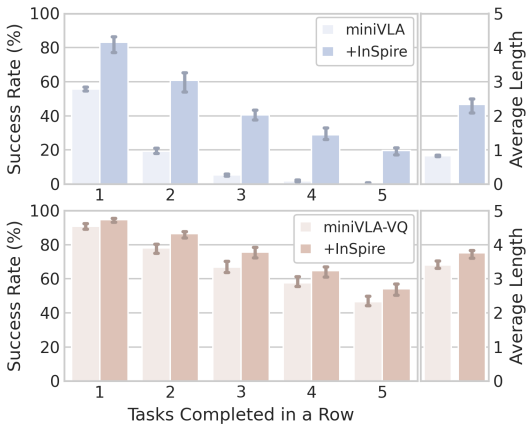

Fig. 4: **CALVIN Performance**.

TABLE II: **Comparison with state-of-the-art**. † independently pretrained from scratch using the four LIBERO datasets. ‡ jointly finetuned on the four LIBERO datasets. The results of SpatialVLA-4B are obtained from its original paper [38], while the results of other rivals are referred from [39].

Model	-Spatial	-Object	-Goal	-Long	**Average**
Diffusion Policy† [9]	78.3	92.5	68.3	50.5	$72.4_{\pm 0.7}$
Octo‡ [23]	78.9	85.7	84.6	51.1	$75.1_{\pm 0.6}$
OpenVLA‡ [15]	84.7	88.4	79.2	53.7	$76.5_{\pm 0.6}$
SpatialVLA-4B‡ [38]	88.2	89.9	78.6	55.5	$78.1_{\pm 0.7}$
CoT-VLA-7B‡ [39]	87.5	91.6	87.6	69.0	$83.9_{\pm 0.6}$
InspireVLA-1B (*Ours*)‡	**90.7**	**94.3**	**88.3**	**73.3**	**86.7**$_{\pm 1.2}$

To comprehensively assess InSpire's capabilities, we evaluate the model on both seen tasks from LIBERO-90 and unseen tasks from the other four datasets. We perform evaluations using two state-of-the-art models: **miniVLA-VQ**[3] and π_0-**FAST** [28]. In our experiments, miniVLA-VQ is pretrained from scratch, whereas π_0-FAST undergoes full-finetuning on the LIBERO-90 dataset. To ensure fair comparisons, both models adopt the same hyperparameters (e.g., learning rate = 1e-5 and training step = 50000) as those employed for InSpire. Since the LIBERO-90 pretrained OpenVLA model has not been released, we borrow the reported results on seen tasks from the project homepage.

We have the following key observations from Table I. **1)** Our InSpire achieves the best or comparable performance compared to the two strong baseline VLAs on both seen and unseen tasks. Remarkably, InSpire enhances the absolute success rates of miniVLA-VQ by **6.2%** and **10%** on seen and unseen tasks, respectively. **2)** The established performance gains of InSpire on π_0-FAST are not as significant as that on miniVLA-VQ. The primary reason could be that we train miniVLA-VQ from scratch but finetune the pretrained π_0-FAST. With substantially less data than π_0-FAST's pretraining dataset, LIBERO-90 limits the ability of InSpire to enhance π_0-FAST's spatial reasoning capability. **3)** InSpire yields significantly better performance on LIBERO-Spatial and LIBERO-Object tasks compared to LIBERO-Goal and LIBERO-Long tasks, demonstrating its strength in handling spatial relationships and object interactions. **4)** SpatialVLA, and other VLAs, exhibit poor generalization performance on tasks from the four unseen datasets, indicating these models' inability to capture true causal relationships between observations and actions, leading to poor generalization beyond the distribution of their training data.

*2) **CALVIN Evaluation**:* CALVIN [24] is a challenging benchmark for long-horizon, language-conditioned robot manipulation. It features 34 tasks with unconstrained language instructions, performed by a Franka Panda robot in an interactive desk environment. The training dataset contains over 20k expert trajectories paired with language instruction labels We follow [24], [30] to train the model to predict delta XYZ positions and delta Euler angles for arm actions and binary gripper actions. We apply our InSpire approach to two baseline models, miniVLA and miniVLA-VQ. We use the ABC→D evaluation protocol, where A, B, C, and D stand for four different environments. We train the model on environment ABC with three random seeds and evaluate it on a distinct test environment D. The evaluation consists of 500 randomly generated task sequences, each comprising five subtasks. The model attempts the subtasks sequentially, proceeding to the next only upon success. We report the number of successfully completed subtasks in a row.

Quantitative results are shown in Fig.4. As observed, our InSpire consistently outperforms all baseline methods in completing 1 to 5 sequential tasks. This strong long-horizon capability is further highlighted by the average number of consecutively completed tasks, where InSpire again surpasses all competitors. This superior performance stems from our method's effectiveness in mitigating spurious correlations, leading to more accurate and robust robot actions.

B. Real-world Experiments

Due to the difficulty in collecting huge pools of real-world behavioral demonstrations, it is challenging to train a model that displays generalization and robustness on varied real-world tasks, whether starting from scratch or finetuning a pretrained model lacking exposure to substantial real-world manipulation data. Hence, we employ π_0-FAST, the current best-performing open-source VLA, as the baseline model for

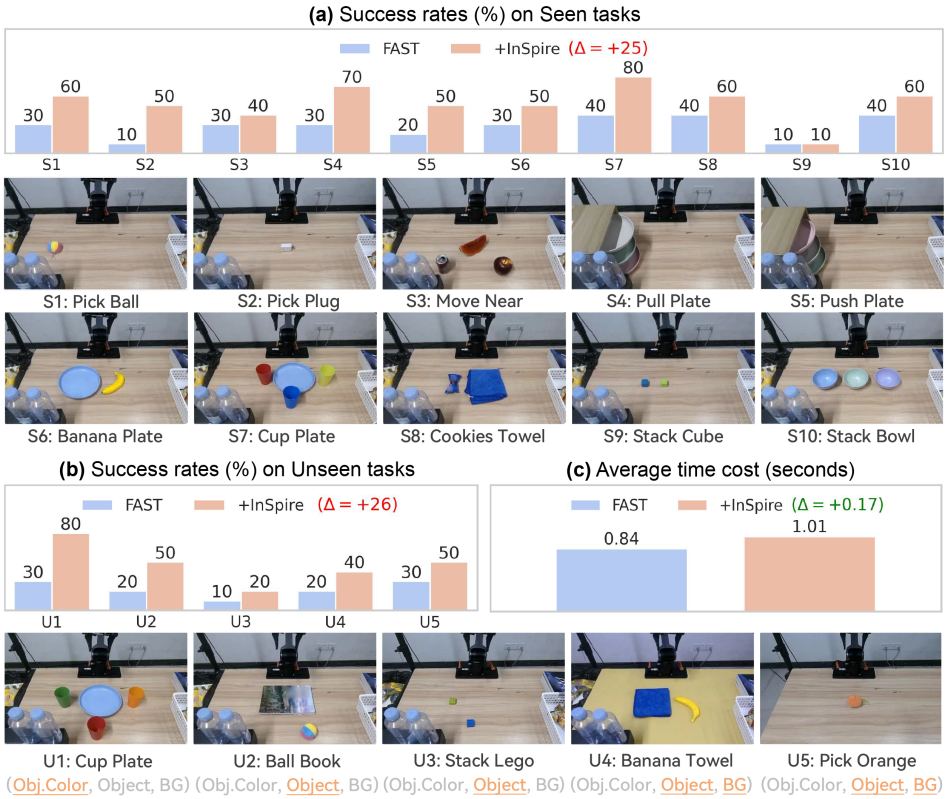

Fig. 5: **Real-world Performance**. **(a)(b)** Success rates (%) of the state-of-the-art model π_0-FAST [28] integrated w/ or w/o InSpire on seen and unseen real-world manipulation tasks. **(c)** Average time cost per step (in seconds) over all seen and unseen tasks. Δ: absolute improvement.

real-world evaluations. We carefully design 10 seen tasks and 5 unseen tasks that focus on evaluating the VLA's performance along multiple dimensions: spatial reasoning, interacting with novel objects and scenes, and following unknown instructions. The π_0-FAST model is full-finetuned using the 10 seen tasks, each consisting of training 10 training trajectories. During inference, we conduct 10 trials per task and randomize the configurations and orientations of task-specific objects for each trial. We use an AGILEX PiPER 6DOF robot arm. Other hyperparameters are consistent with those used in the simulation experiments.

Fig. 5 reports the success rates as well as time costs of π_0-FAST [28] integrated w/ or w/o our InSpire on 10 seen and 5 unseen real-world tasks. From the obtained results in the figure, we have several key observations. **1)** InSpire consistently enhances the success rates of the strong baseline VLA across 10 seen and 5 unseen tasks, achieving an average enhancement of **25**% and **26**%, respectively. **2)** InSpire achieves notable performance gains and exhibits robustness against variations in object color, objects, and backgrounds in unseen tasks. Particularly, InSpire increases the relative success rate of the baseline by **100**% on 4/5 of the unseen tasks. **3)** InSpire incurs an additional time cost of 0.18 seconds per step on average compared to the baseline VLA. The additional computational overhead is primarily attributed to the spatial reasoning VQA task introducing more tokens into the model's action generation process. However, considering the notable improvement in success rates, this trade-off is acceptable in a wide range of real-world robotic applications. **4)** The backgrounds of the 15 real-world tasks are significantly more complex than those of the simulation tasks from the LIBERO environment, with numerous distracting elements like plastic bottles, baskets, and trash cans. Nevertheless, our InSpire demonstrates outstanding performance, consistent with the results observed in simulation experiments.

C. *Comparison with State-of-the-art*

In this part, we first apply InSpire to the miniVLA-VQ model and pretrain a 1B-parameter VLA on the LIBERO-90 dataset, which we refer to as **InspireVLA-1B**. We then compare our InspireVLA-1B with state-of-the-art reasoning-based VLAs, including SpatialVLA-4B[38]—a 4B-parameter VLA that injects 3D information into the input observations to achieve spatial-aware action prediction, and CoT-VLA-7B [39]—a 7B-parameter VLA that enhances action prediction through CoT reasoning. We follow the common setup in [38], [39], leveraging tasks from LIBERO-Spatial, -Object, -Goal and -Long for evaluation. Each of SpatialVLA-4B, CoT-VLA-7B and our InspireVLA-1B is jointly fine-tuned on tasks from the four LIBERO datasets. We compare the performance on seen tasks from the four datasets and directly borrow the results of SpatialVLA-4B and CoT-VLA-7B from

TABLE III: Ablation of the formulation of the spatial reasoning VQA task.

Setting	Question	Answer	Seen	Unseen
Baseline	-	-	$83.3_{\pm 1.2}$	$3.6_{\pm 0.4}$
1D Direct.	In which direction is the [object] relative to the robot?	right/left/.../back/grasped	$\mathbf{89.5}_{\pm 1.5}$	$13.6_{\pm 3.9}$
3D Direct.	In which direction is [object] relative to the robot? x, y, z :	[right, up, front]/.../grasped	$87.9_{\pm 2.1}$	$\mathbf{15.0}_{\pm 1.9}$
Proximity	What is the distance between the robot and [object]?	far/middle/near/grasped	$88.7_{\pm 1.1}$	$10.3_{\pm 1.0}$
3D Locat.	What is the accurate posi. of [object] relat. to the robot? x, y, z :	$[1, -3, 4]/.../[2, 0, 1]$	$88.1_{\pm 0.8}$	$10.1_{\pm 2.6}$
Distance	What is the accurate distance between the robot and [object]?	0/1/2/3/.../9	$85.1_{\pm 2.2}$	$6.6_{\pm 0.6}$

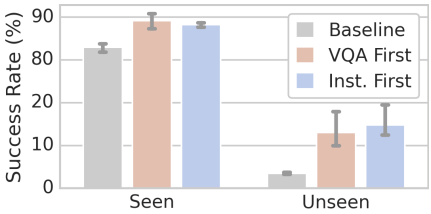

Fig. 6: Ablation study on the insertion position of the spatial reasoning VQA task.

the original papers. We perform LoRA fine-tuning [12] on InSpireVLA-1B using tasks from the four datasets. We report the average success rates and std errors over 3 random seeds. As in [39], we also compare InSpire with other three strong baselines: Diffusion Policy[9], Octo[23] and OpenVLA[15], their results are directly referred from [39][1].

From the obtained results in Table II, we have the following key observations. **1)** Our InspireVLA-1B model consistently outperforms the five VLAs across the four LIBERO datasets. Particularly, InspireVLA-1B improves the two reasoning-based models SpatialVLA-4B and CoT-VLA-1B by **8.6**% and **2.8**%, with **4**× and **7**× parameter efficiency, respectively. **2)** The three reasoning-based models (i.e., SpatialVLA-4B, CoT-VLA-7B and our InspireVLA-1B) achieve better performance than other three competitors, emphasizing the necessity of integrating an intermediate reasoning process into existing VLAs to capture the true causal relationships between observations and actions. **3)** InspireVLA-1B demonstrates remarkable advantages on LIBERO-Spatial and LIBERO-Object, highlighting its effectiveness in transferring knowledge about spatial information and objects.

D. *Ablations on Design Decisions*

In this section, we conduct ablation studies in the LIBERO environment to explore how the performance of InSpire varies with different design decisions. We employ miniVLA-VQ as the baseline VLA, and conduct 100 trials per task. Our ablation studies use a single fixed seed as in [39], [33], [26].

*1) **VQA Formulations**:* InSpire leverages a spatial reasoning VQA task as the bridging "language" between observations and actions. This implies that the formulation of the VQA task dictates the types of spatial information used by InSpire. Here, we investigate the impacts of different VQA formulations on InSpire's performance. As shown in Table III, we design five VQA tasks to introduce diverse spatial information, denoted as "1D Direction", "3D Direction", "Proximity", "3D Location" and "Distance". The performance of our InSpire approach integrated with each of these VQA tasks is reported in the table. We have the following observations from the results in the table. **1)** The four VQA formulations consistently outperform the baseline on both seen and unseen tasks. **2)** "1D Direction" achieves the best performance among all VQA formulations, surpassing the baseline by large margins. **3)** "3D Direction" demonstrates performance close to that of "1D Direction", with both significantly outperforming the other VQA formulations. This suggest that direction prediction-based VQA tasks are more effective in guiding the action generation process.

*2) **VQA Insertion Positions**:* In our developed InSpire approach, the tokens for the spatial reasoning VQA task are appended following the input vidsual observation tokens and preceding the language instruction tokens (Fig. 2). In this experiment, we investigate the impacts of different insertion positions of those VQA tokens on performance. Fig. 6 presents a quantitative comparison of InSpire's performance with two design decisions: inserting VQA tokens before the language instruction tokens ("VQA-First") and placing them after the language instruction tokens ("Instruct-First"). As seen, "VQA-First" achieves superior performance on seen tasks compared to "Instrut-First", but lags behind on unseen tasks. Moreover, InSpire substantially boosts the baseline's performance in both settings, highlighting the benefits of the auxiliary VQA task for enhancing spatial reasoning.

E. *How Does InSpire Help Resolve Spurious Correlations?*

In this section, we present qualitative results to explore how InSpire—and spatial reasoning more generally—help mitigate the adverse effects of spurious correlations. Fig. 7 illustrates some representative failure modes encountered by the baseline model miniVLA-VQ but successfully addressed by InSpire. The results in the figure reveal several strengths of our InSpire approach:

*1) **Addressing Shortcut Learning of VLAs**:* As shown in Fig. 7 (Left) **Task A**, the baseline VLA, upon observing the "drawer" frequently seen during training, directly approaches it based on shortcut learning without understanding the input language instruction and visual information. In other words, the VLA learns spurious correlations between task-irrelevant features of observations and actions, leading to overfitting to seen tasks and poor generalization on unseen tasks. Our

[1]Since the LIBERO-pretrained checkpoints of Diffusion Policy, Octo, OpenVLA and CoT-VLA-7B in [39] have not been open-sourced, we are unable to compare the performance of those models on unseen tasks.

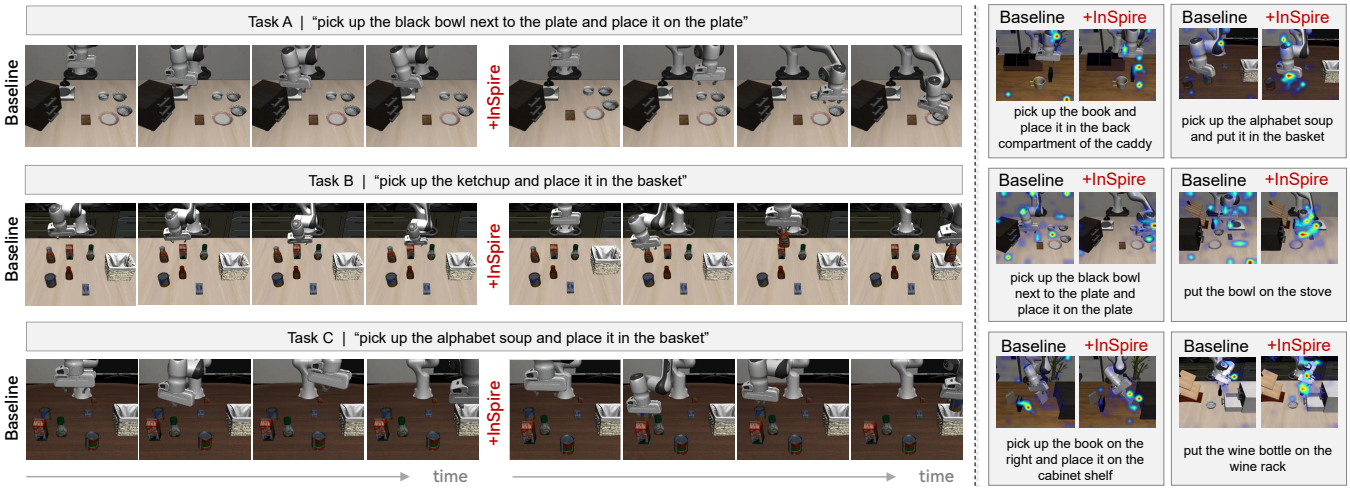

Fig. 7: **Qualitative Results**. (Left) Action sequences and (Right) attention maps produced by the baseline model (mini-VLA-VQ [3]) integrated w/ or w/o our proposed InSpire approach.

InSpire approach corrects the shortcut learning exhibited by the baseline, avoiding the neglect of essential elements like language instructions and spatial relations.

*2) **Improving Model Robustness to Distractors**:* As illustrated in Fig. 7 (Left) **Task B**, the baseline VLA struggles to identify task-specific objects in complex scenarios with various distractors, limiting its ability to produce correct actions. The primary reason may be that the direct observation-to-action mapping mechanism prevents the learned model from capturing causal relationships between observations and actions, making it sensitive to familiar distractors when predicting actions. By employing a spatial reasoning VQA task as the bridging "language" between observations and actions, InSpire equips the VLA with spatial reasoning capability, thereby helping the VLA differentiate task-relevant objects from distractors (as shown in Fig. 7 (Right)).

*3) **Enabling Continual Action Correction**:* Continual correction of incorrect actions is an essential skill for both humans and robots. Yet, as shown in Fig. 7 (Left) **Task C**, the baseline VLA fails to correct the incorrect action and proceeds with following actions. This suggests the learned VLA uses perceptual data for decision-making in a very different way from how humans do. Conversely, InSpire assists the model in continuously correcting errors until the task is completed. This is primarily because InSpire performs spatial reasoning for each visual observation along the trajectory, which enhances the model's awareness of task execution status (e.g., the relative direction of the target object to the robot, the status of the gripper) and assists in correcting incorrect actions.

IV. Related Work

*1) **Vision-Language-Action Models (VLAs)**:* VLAs, built upon pretrained Vision-Language Models (VLMs) [13], [36], demonstrate strong generalization in robotics [6], [15]. This line of research, initiated by transformer-based frameworks like RT-1 [5] and advanced by transferring web-scale knowledge in RT-2 [6], has scaled to large datasets such as Open X-Embodiment [10]. Recent architectures leverage powerful foundation models (OpenVLA [16]), target computational efficiency (Octo [23]), or refine training and control strategies (π_0[4], π_0-FAST [28]).

*2) **Chain-of-Thought (CoT) Reasoning**:* CoT reasoning has been adapted from NLP to high-level robotic task planning [34], [29], [17], [2], [14]. Applications include sub-goal planning via diffusion or video generation [27] and reward shaping in reinforcement learning [37]. Models like CoT-VLA [39] and ECoT [33] use visual goal prediction and iterative reasoning, respectively, but can be inefficient by depending on complex intermediate computations or external models. The most related work, RT-H [2], uses language motion prediction as an intermediate step. In contrast, our InSpire framework employs a spatial reasoning VQA task to explicitly associate objects of interest with corresponding actions, reducing the impact of spurious correlations.

*3) **Learning Spatial Reasoning**:* Leveraging the spatial reasoning of large models is a recent focus in multimodal learning [22], [7], [18], [20]. Methods range from using 2D VLMs for spatial understanding without explicit 3D data (SpatialVLM [8]) to integrating 3D scene graphs and depth information (SpatialRGPT [18]). Our work is most related to SpatialVLA [38], which injects 3D information and uses discretized action grids. However, instead of relying on encoded 3D positions, our approach employs a VQA task as a bridging "language" to enable simultaneous spatial reasoning and action prediction, endowing VLAs with this capability without extra data or models.

V. Conclusion, Limitations and Future Work

This work proposes Intrinsic Spatial Reasoning (InSpire) to mitigate the adverse effects of spurious correlations on the action prediction of VLAs. By incorporating a spatial reasoning VQA task as the bridging "language" between visual observations and low-level actions, InSpire endows VLAs with spatial reasoning capabilities without employing

extra data or interacting with other large models. Notably, InSpire can be used as a plugin to improve existing autoregressive VLAs. Comprehensive experiments demonstrate InSpire's effectiveness and flexibility, achieving consistent improvements on two state-of-the-art VLAs on both seen and unseen tasks in simulation and real-world environments.

Although the results are encouraging, two key limitations remain in this work: **1)** Although InSpire has potential for integration with any LLM-based VLAs, its effectiveness in enhancing diffusion policy-based VLAs, such as Octo [23] and π_0 [4], remains to be explored. Future work will focus on exploring strategies to facilitate seamless integration of InSpire into such models. **2)** VLAs pretrained with diverse paradigms and datasets likely acquire varied spurious correlations; however, the characters of these differences across models are not yet understood. A deeper empirical study of these factors is a promising direction for bolstering the robustness of our approach in more complex scenarios.

ACKNOWLEDGMENT

This study is supported by grants from the National Natural Science Foundation of China (Grant No. U22A2097, No.82441006, No. U23A20315, No. 62425208) and the China Postdoctoral Science Foundation (No. 2025M781517).

REFERENCES

[1] D. Arpit, S. Jastrzębski, N. Ballas, D. Krueger, E. Bengio, M. S. Kanwal, T. Maharaj, A. Fischer, A. Courville, Y. Bengio, et al. A closer look at memorization in deep networks. In *ICML*, pages 233–242. PMLR, 2017.

[2] S. Belkhale, T. Ding, T. Xiao, P. Sermanet, Q. Vuong, J. Tompson, Y. Chebotar, D. Dwibedi, and D. Sadigh. Rt-h: Action hierarchies using language. *arXiv preprint arXiv:2403.01823*, 2024.

[3] S. Belkhale and D. Sadigh. Minivla: A better vla with a smaller footprint. `https://ai.stanford.edu/blog/minivla/`, 2024.

[4] K. Black, N. Brown, D. Driess, et al. Pi-0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

[5] A. Brohan, N. Brown, J. Carbajal, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

[6] A. Brohan, N. Brown, J. Carbajal, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.

[7] B. Chen, Z. Xu, S. Kirmani, B. Ichter, D. Sadigh, L. Guibas, and F. Xia. Learning to localize objects improves spatial reasoning in visual-llms. *arXiv preprint arXiv:2403.12345*, 2024.

[8] B. Chen, Z. Xu, S. Kirmani, B. Ichter, D. Sadigh, L. Guibas, and F. Xia. Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. In *CVPR*, pages 14455–14465, 2024.

[9] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 42(5-6):368–387, 2023.

[10] O. X.-E. C. et al. Open X-Embodiment: Robotic learning datasets and RT-X models. `https://arxiv.org/abs/2310.08864`, 2023.

[11] I. Higgins, A. Pal, A. Rusu, L. Matthey, C. Burgess, A. Pritzel, M. Botvinick, C. Blundell, and A. Lerchner. Darla: Improving zero-shot transfer in reinforcement learning. In *ICML*, pages 1480–1490. PMLR, 2017.

[12] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. LoRA: Low-rank adaptation of large language models. In *ICLR*, 2022.

[13] S. Karamcheti, S. Nair, A. Balakrishna, et al. Prismatic vlms: Investigating the design space of visually-conditioned language model. In *ICML*, page 235, 2024.

[14] M. J. Kim, C. Finn, and P. Liang. Fine-tuning vision-language-action models: Optimizing speed and success. *arXiv preprint arXiv:2502.19645*, 2025.

[15] M. J. Kim, K. Pertsch, S. Karamcheti, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

[16] M. J. Kim, K. Pertsch, S. Karamcheti, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

[17] Y. Li, Y. Deng, J. Zhang, J. Jang, M. Memmel, R. Yu, C. R. Garrett, F. Ramos, D. Fox, A. Li, et al. Hamster: Hierarchical action models for open-world robot manipulation. *ICLR*, 2025.

[18] Y. Li, J. Liu, L. Zhang, L. Yang, J. Zhang, K. Keutzer, and H. Zhao. Spatialrgpt: Grounded spatial reasoning in vision language models. *NeurIPS*, 36:12345–12358, 2023.

[19] B. Liu, Y. Zhu, C. Gao, Y. Feng, Q. Liu, Y. Zhu, and P. Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *NeurIPS*, 36:44776–44791, 2023.

[20] H. Liu, C. Li, Y. Li, B. Li, Y. Zhang, S. Shen, and Y. J. Lee. Large language models are visual reasoning coordinators. *NeurIPS*, 36:26332–26345, 2023.

[21] E. Loper and S. Bird. Nltk: The natural language toolkit. *arXiv preprint cs/0205028*, 2002.

[22] P. Lu, H. Bansal, T. Xia, J. Liu, J. May, A. Tanwani, S. Savarese, B. Wu, S.-C. Zhu, L. Fei-Fei, et al. Mind's eye of llms: Visualization-of-thought elicits spatial reasoning in large language models. *arXiv preprint arXiv:2310.08594*, 2023.

[23] T. O. M., D. Ghosh, H. Walke, et al. Octo: An open-source generalist robot policy. *arXiv preprint arXiv:2405.12213*, 2024.

[24] O. Mees, L. Hermann, E. Rosete-Beas, and W. Burgard. Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. *IEEE Robotics and Automation Letters (RA-L)*, 7(3):7327–7334, 2022.

[25] V. Myers, B. C. Zheng, O. Mees, S. Levine, and K. Fang. Policy adaptation via language optimization: Decomposing tasks for few-shot imitation. In *CoRL*, 2024.

[26] M. Nakamoto, O. Mees, A. Kumar, and S. Levine. Steering your generalists: Improving robotic foundation models via value guidance. In *CoRL*, 2024.

[27] F. Ni, J. Hao, S. Wu, L. Kou, J. Liu, Y. Zheng, B. Wang, and Y. Zhuang. Generate subgoal images before act: Unlocking the chain-of-thought reasoning in diffusion model for robot manipulation with multimodal prompts. In *CVPR*, 2024.

[28] K. Pertsch, K. Stachowicz, B. Ichter, D. Driess, S. Nair, Q. Vuong, O. Mees, C. Finn, and S. Levine. Fast: Efficient action tokenization for vision-language-action models. *arXiv preprint arXiv:2501.09747*, 2025.

[29] L. X. Shi, B. Ichter, M. Equi, L. Ke, K. Pertsch, Q. Vuong, J. Tanner, A. Walling, H. Wang, N. Fusai, et al. Hi robot: Open-ended instruction following with hierarchical vision-language-action models. *arXiv preprint arXiv:2502.19417*, 2025.

[30] H. Wu, Y. Jing, C. Cheang, G. Chen, J. Xu, X. Li, M. Liu, H. Li, and T. Kong. Unleashing large-scale video generative pre-training for visual robot manipulation. In *ICLR*, 2024.

[31] S. Wu, X. Luo, J. Zhang, J. Xie, J. Song, H. T. Shen, and L. Gao. Policy contrastive decoding for robotic foundation models. *arXiv preprint arXiv:2505.13255*, 2025.

[32] M. Zawalski, W. Chen, K. Pertsch, O. Mees, C. Finn, and S. Levine. Robotic control via embodied chain-of-thought reasoning. *arXiv preprint arXiv:2407.08693*, 2024.

[33] M. Zawalski, W. Chen, K. Pertsch, O. Mees, C. Finn, and S. Levine. Robotic control via embodied chain-of-thought reasoning. *arXiv preprint arXiv:2407.08693*, 2024.

[34] M. Zawalski, W. Chen, K. Pertsch, O. Mees, C. Finn, and S. Levine. Robotic control via embodied chain-of-thought reasoning. In *CoRL*, 2024.

[35] X. Zhai, B. Mustafa, A. Kolesnikov, and L. Beyer. Siglip: Sigmoid loss for language image pre-training. *arXiv preprint arXiv:2303.15343*, 2023.

[36] X. Zhai, B. Mustafa, A. Kolesnikov, et al. Sigmoid loss for language image pre-training. In *ICCV*, pages 11975–11986, 2023.

[37] K. Zhang, Z.-H. Yin, W. Ye, and Y. Gao. Learning manipulation skills through robot chain-of-thought with sparse failure guidance. *arXiv preprint arXiv:2405.13573*, 2024.

[38] Y. Zhang, C. Li, X. Wang, Y. Yang, A. Gupta, and T. Darrell. SpatialVLA: Exploring spatial representations for visual-language-action models. In *CVPR*, pages 14567–14578, 2024.

[39] Q. Zhao, Y. Lu, M. J. Kim, Z. Fu, Z. Zhang, Y. Wu, Z. Li, Q. Ma, S. Han, C. Finn, et al. Cot-vla: Visual chain-of-thought reasoning for vision-language-action models. *CVPR*, 2025.