

---

# Vision-Language-Action Model with Open-World Reasoning

---

Anonymous Author(s)

Affiliation

Address

email

## 1 A Demo Link

2 We provide an anonymous link to access our demo: [video link](#). The demo showcases our model's  
3 performance on math matching game and the toy placement task, illustrating the generalization ability  
4 of our ChatVLA-2.

## 5 B Limitation

6 Our work investigate to retain the pre-trained knowledge from the vision-language model in vision-  
7 language-action model. As such, the VLA are able to reasoning over the image observation and  
8 language instruction, and enforce the action model to follows such reasoning. Currently, we are  
9 unable to fully retain the pre-trained knowledge from VLM. We observe that it is inevitable that  
10 many capacity disappear during the fine-tuning with robot data. This is the most challenging part,  
11 and current approach cannot fully resolve this problem. We leave this to the future work. Also, our  
12 current method is mainly conducted on table top tasks. We aim to expand the embodiment to mobile  
13 manipulator to perform more long-horizon and complex real world tasks in the future.

## 14 C Implementation Details

### 15 C.1 Training details.

16 We utilize 8 NVIDIA H800 GPUs (80GB each) for training. We adopt mixed-precision training  
17 (FP16) and use the AdamW optimizer. For training stage 1, we co-train on image-text data and robot  
18 data, setting the initial learning rate to 2e-5 and training for 15k steps. For training stage 2, we freeze  
19 the VLM backbone. The model is trained for 50k steps, starting with a learning rate of 2e-5 and  
20 a warm-up phase over the first 3k steps. In both stages, we apply a cosine learning rate scheduler,  
21 scaling down the learning rate to 2e-6. The total training cost is 340 GPU hours.

### 22 C.2 Data details.

23 **Image-text data composition.** The image-text dataset used in our experiments integrates samples  
24 from multiple established benchmarks, including COCO, TextVQA, and GQA, alongside additional  
25 data specifically constructed to align with our task formulation. To ensure balanced representation,  
26 we incorporate approximately 32k samples from COCO, 20k from TextVQA, and 54k from GQA.  
27 These robotics-related image-text pairs employ the reasoning template used in the toy placement task,  
28 as illustrated in C.2. Furthermore, we utilize data from RoboPoint, comprising approximately 2k  
29 samples collected within a simulated environment. Although the RoboPoint data exhibits lower visual  
30 quality due to visual discrepancies and camera viewpoints, our experiments indicate that including  
31 this data enhances the visual-language alignment (VLA) model's spatial understanding capabilities.

Table 1: Ablation study on number of experts.

Expert numbers	Top-k numbers	OCR	Math
8	2	<b>3.58</b>	<b>1.73</b>
6	3	2.42	1.26
4	2	1.87	0.94

Table 2: Ablation study on reasoning-following enhancement module.

Method	Avg. success rate
<b>Latter-half-layer injection</b>	<b>43/52</b>
Full-layer injection	36/52
Former-half-layer injection	22/52

32 Additionally, we gathered 5k samples from real-world environments, covering both tabletop setups  
 33 and broader scenes. These samples follow a similar annotation format to the LLaVA dataset, utilizing  
 34 a question-answering structure. All collected data is combined and utilized collectively during  
 35 training in our method.

36 **Data pre-processing.** For the image-text data, we limit each example to a maximum of 5 dialogue  
 37 turns. If an instance originally contains more than 5 turns, we retain the first turn and randomly  
 38 sample four additional turns from the remainder. For the TextVQA dataset, we specifically select  
 39 samples that do not contain numeric OCR tokens or mathematical operators, as our goal is to utilize  
 40 pre-trained knowledge for open-world manipulation. We use the image resolution of  $320 \times 240$ .

41 **Reasoning templates of robot data.** All our robot data are annotated with sub-reasoning, similar  
 42 to the approach used in  $\pi_{0.5}$  and DexVLA. We initialize these reasoning annotations with fixed  
 43 templates and then augment them using GPT-4o, following a pipeline analogous to the one employed  
 44 in training large language models. This method allows us to keep our reasoning phrase flexible, such  
 45 that the action expert would not dominate by certain template.

## 46 D More Ablation Studies

47 We have discussed the importance of some key components in our ChatVLA-2 in the main text,  
 48 including the choice of mixture-of-experts and the two-stage training strategy. In this section, we will  
 49 further discuss the following questions:

### 50 D.1 Ablation study on number of experts.

51 We conduct experiments to check how many experts we should use to better obtain pretrained  
 52 knowledge from VLM while maintaining appropriate resource consumption. As is shown in Table  
 53 1, experimental results indicate that increasing both the total number of experts and the number of  
 54 experts selected during inference can enhance the model’s generalization ability in robotic scenarios.

55 A possible explanation for this phenomenon is that, a limited number of experts tend to develop  
 56 selection biases toward visually similar task images in such scenarios. This can lead to overfitting  
 57 on robot data and result in the neglect of the pretrained VLM knowledge, ultimately degrading  
 58 performance.

### 59 D.2 Ablation Study on Layers for Injecting Reasoning-Following Enhancement Module.

60 As shown in the main text, we replace the original observation embedding with reasoning tokens  
 61 and use them to condition the generation of scale and shift parameters in the latter half layers of the  
 62 action expert. This mechanism effectively injected reasoning context into the model. In this section,  
 63 we conduct experiments on the place of injecting reasoning. The results are shown in Table 2.

64 Experiments show that the former half layers of action expert significantly impacts action generation  
 65 stability. Introducing reasoning information into the former half layers actually increases instability  
 66 in the generated actions, which in turn significantly reduces task success rates. We hypothesize  
 67 that this effect may due to our design choice of replacing the original observation embedding with  
 68 reasoning information. One possible explanation is that the observations themselves may carry critical  
 69 information for action generation, and their removal could negatively affect performance.