

---

# Supplementary Material of DreamVLA

---

Anonymous Author(s)

Affiliation

Address

email

## 1 Implementation Details

### 1.1 DreamVLA Architecture

**Text Encoder.** We use the CLIP ViT-B/32 text encoder [1] to process natural language task instructions. The encoder transforms each instruction into a fixed-length embedding that captures semantic intent. These embeddings are then projected into the shared latent space and used to condition the subsequent modules, enabling effective grounding of language into perception and action.

**Visual Encoder.** We employ an MAE-pretrained ViT-B [2] as the vision encoder. At each timestep, images are captured from two views: eye-on-hand and eye-on-base. Each image is processed by the vision encoder to produce 196 latent vectors, which represent local patch information, along with a [CLS] token that encodes the global representation of the image. Directly inputting all 197 tokens into the transformer backbone would create a significant computational burden, particularly when processing long histories. Moreover, many image details are redundant for accomplishing manipulation tasks. To address this, we utilize the Perceiver Resampler [3] to condense the image representations and extract task-relevant features. The Perceiver Resampler employs learnable latent vectors with a shape of (num latents, dim), where num latents is significantly smaller than the number of image tokens. Through Perceiver Attention, these latent vectors condense the input image features, along with the [CLS] token, to form the final image tokens.

**Robot State** The robot state consists of the arm and gripper state. The arm state includes the end-effector position and its rotation in Euler angles, resulting in a six-dimensional representation. The gripper state is a binary value indicating whether the gripper is open or closed. We tokenize the robot state using an MLP. Specifically, the gripper state is first converted into a one-hot encoding. The one-hot encoding of the gripper state and the arm state are then each passed through separate linear layers. The outputs are concatenated and passed through a final linear layer to produce the state token.

**Learnable Queries.** We introduce two sets of learnable query tokens, denoted as  $\langle \text{dream} \rangle$  and  $\langle \text{action} \rangle$ , to extract and integrate information from multimodal inputs for joint prediction.

The  $\langle \text{dream} \rangle$  queries provide structured supervision through comprehensive knowledge prediction tasks and consist of 64 tokens in total—organized as 16 queries for each of the four modalities: dynamic motion, depth estimation, DINOv2 feature recovery, and segmentation. These queries guide the model in reconstructing rich visual representations, enhancing the quality of the learned latent space.

The  $\langle \text{action} \rangle$  queries are dedicated to action sequence prediction. Their length is determined by the temporal prediction horizon, as defined in the action chunking strategy from [4].

Table 1: The parameter of the each module in DreamVLA.

	Hidden size	Number of layers	Number of heads
image encoder	768	12	12
perceiver resampler	768	3	8
LLM	1024	24	16
image decoder	1024	2	16
depth decoder	1024	2	16
DINO decoder	1024	2	16
segment decoder	1024	2	16

**Large Language Models.** We adopt GPT-2 Medium [5] as our language backbone. GPT-2 Medium is a 24-layer, 16-head Transformer decoder with a hidden size of 1,024 and a total of approximately 345 million parameters. It was pretrained on the WebText corpus ( $\sim 8$  million documents, 40 GB of text) using autoregressive language modeling to predict the next token with a byte-pair encoding vocabulary of 50,257 tokens.

**Output Heads.** To decode the *world embedding* into comprehensive world knowledge, we incorporate multiple task-specific output heads that predict dynamic motion regions, depth maps, and high-level semantics, including DINOv2 [6] and SAM-style segmentation features [7].

Each prediction head is implemented using a lightweight Vision Transformer (ViT) decoder, which operates on two types of tokens produced by the multimodal backbone: the latent embeddings associated with a specific modality, and a set of learnable mask tokens used for reconstruction.

To retain spatial correspondence, we inject fixed sine-cosine positional encodings into the token embeddings. These tokens are then processed through several Transformer encoder layers, followed by a modality-specific linear projection head that maps each patch token to its output space—such as per-pixel depth values or semantic logits—thereby reconstructing the expected visual signals of future observations.

The concrete detail of every module is shown in Table. 1

**Action Prediction with Diffusion Transformer** To generate future actions conditioned on multimodal context, we adopt a diffusion-based Transformer architecture, DiT-B [8], as our action decoder. DiT enables flexible modeling of complex action distributions by progressively denoising a sequence of latent action tokens through a series of Transformer layers, allowing the model to capture multimodal uncertainty in robot control.

We configure the DiT model with the base variant (DiT-B), using an action token embedding size equal to the hidden dimension of the fusion Transformer. The model predicts  $K$  future actions in an autoregressive-free manner, where each action is a 7-dimensional vector encoding end-effector pose displacement and gripper state. In our experiments, we set  $K = 2$ , corresponding to a 3-frame prediction window (current + 2 future steps). The model does not utilize past action context during generation (i.e., past window size is 0), focusing solely on predictive synthesis.

During training, Gaussian noise is added to the future action trajectories, and the model learns to reverse this corruption process step by step. This module operates on top of the fused multimodal representation, enabling temporally coherent and semantically grounded action generation. The concrete detail of DiT is shown in Table. 2

## 1.2 Feature Extraction

To facilitate dynamic region prediction, we adopt a motion-based heuristic to generate coarse binary masks that highlight regions of interest. Given a sequence of consecutive RGB frames of resolution  $H \times W$ , we uniformly sample one keypoint every 8 pixels in both spatial dimensions, resulting in  $N = \lfloor H/8 \rfloor \times \lfloor W/8 \rfloor$  sampled locations per frame. For each sampled location, we compute inter-frame displacements  $(\Delta x, \Delta y)$  by tracking its position across adjacent frames using CoTracker [9].

Table 2: Configuration of the DiT-B model used for action prediction.

Parameter	Value
Model type	DiT-B
Token size	1024
Action prediction window	2 future steps (3-frame chunk)
Past context steps	0
Number of Transformer layers	12
Number of attention heads	12
Positional encoding	Learned (1D for time)
Diffusion timesteps (Train)	8
Diffusion timesteps (Inference)	10
Noise schedule	Linear
Loss function	Denosing Score Matching (L2 loss)
Precision	float32

73 The magnitude of displacement is converted into a scalar speed value:

$$s_{ij} = \sqrt{(\Delta x_{ij})^2 + (\Delta y_{ij})^2},$$

74 where  $(i, j)$  denotes the spatial coordinates of each sampled patch. We then apply a speed threshold  
75  $\tau$  (e.g.,  $\tau = 1$  pixel/frame) to obtain a binary motion mask. To account for small motions and ensure  
76 spatial connectivity, we perform a single-pixel morphological dilation, expanding each positive  
77 location to its eight-connected neighbors.

78 The resulting mask is flattened and reshaped into the form  $(B, 1, L)$ , where  $L = \lfloor H/8 \rfloor \cdot \lfloor W/8 \rfloor$  and  
79  $B$  is the batch size. We apply this binary mask element-wise to both predicted patch embeddings  
80  $\{\hat{p}_i\}$  and their corresponding ground-truth embeddings  $\{p_i\}$  during loss computation, encouraging  
81 accurate representation in dynamic regions.

82 For depth supervision, we use the ground-truth depth maps provided by datasets when available. In  
83 cases where depth annotations are not provided—such as in certain real-world robot datasets—we use  
84 monocular depth estimators, specifically Depth-Anything v2 [10], to generate pseudo-ground-truth  
85 depth labels.

86 In addition to depth and dynamic signals, we include from high-level features supervision. For DI-  
87 NOv2 [6], we extract features from the final transformer layer, capturing global semantic and structural  
88 representations. For SAM [7], we utilize the output of its image encoder as dense segmentation-aware  
89 features. These diverse modalities collectively provide comprehensive supervision signals to improve  
90 the quality and generalizability of our learned visual representations.

### 91 1.3 Training Detail

92 The total loss can be formulated as:

$$\mathcal{L} = \lambda_{dyn}\mathcal{L}_{dyn} + \lambda_{depth}\mathcal{L}_{depth} + \lambda_{sem}\mathcal{L}_{sem} + \lambda_{DiT}\mathcal{L}_{DiT} \quad (1)$$

93 where  $\lambda_{dyn} = 0.1$ ,  $\lambda_{depth} = 0.001$ ,  $\lambda_{sem} = 0.1$ ,  $\lambda_{DiT} = 1$ .

94 We train DreamVLA on 8 NVIDIA A800 GPUs. The main bottlenecks is the memory bandwidth to  
95 load large spatial feature tensors, for example, of  $256 \times 64 \times 64$  for SAM. We pre-compute the features  
96 from all teacher models instead of doing inference on the fly. This approach requires extra storage  
97 space to save all the features extracted from the VFMs, but significantly saves on training time and  
98 avoids loading models with high GPU memory usage during training. All training configurations are  
99 listed in Table. 3

Table 3: DreamVLA Training Configuration

Hyperparameters	Value
# GPUs	8
Batch size	8 / GPU (64 effective)
Learning rate (LR)	1e-3
LR Schedule	Constant
Weight decay	0.01
Optimizer	AdamW
Betas	[0.9, 0.999]
Epochs	20
Warm-up epochs	1
Warm-up LR schedule	Linear (1e-2 * LR)

## 2 Experiments

### 2.1 Simulation Benchmark and Settings

We evaluate DreamVLA on the CALVIN benchmark [11], a simulated robotic manipulation suite designed for studying long-horizon, language-conditioned tasks. CALVIN aims to facilitate the development of agents that operate solely based on onboard sensor inputs and free-form human instructions, without access to privileged information or external supervision. The tasks in CALVIN require agents to execute long sequences of low-level control commands in response to complex language goals, reflecting realistic robotic interaction scenarios.

The benchmark includes four structurally similar but visually distinct environments, referred to as Env A, B, C, and D. Each environment features a Franka Emika Panda arm with a parallel gripper and a tabletop workspace containing manipulable elements such as a sliding door, a drawer, and a light button. The textures, object placements, and scene layouts vary across environments to encourage generalization and robustness.

Observations consist of RGB images from both fixed and gripper-mounted cameras (resized to 224×224), as well as low-dimensional robot state inputs that include the end-effector’s position, orientation, and gripper status. The agent outputs a 7-dimensional continuous action vector: 6 dimensions control the spatial displacement of the gripper, and the final dimension governs the open/close state of the gripper.

The dataset contains approximately 2.4 million interaction steps and 40 million short-horizon action windows. Environments A, B, and C provide language-free demonstrations for large-scale pretraining, while annotated instructions are available in a subset of the data for downstream policy learning. We hold out Env D for evaluation to assess zero-shot generalization to unseen combinations of instructions and environment variations.

Following standard protocol [11, 12], we evaluate performance on a set of 34 diverse tasks that include object pushing, placing, rotating, and other dexterous operations. In contrast to prior work, DreamVLA not only predicts actions conditioned on visual-language observations but also simultaneously learns to infer comprehensive future world knowledge including depth maps, dynamic saliency regions, DINOv2 features, and SAM-based segmentation maps. This multi-task supervision enables richer scene understanding and improves policy generalization. We report success rate (SR) as our primary evaluation metric, measuring whether the instructed task was completed correctly based on the final state of the environment.

### 2.2 Simulation Results

We evaluate our approach on the CALVIN ABC-D benchmark, where training is conducted on environments A, B, and C, and testing is performed exclusively in Environment D. This evaluation setting poses a strong challenge for generalization, as Environment D features novel textures, object arrangements, and visual configurations not seen during training.

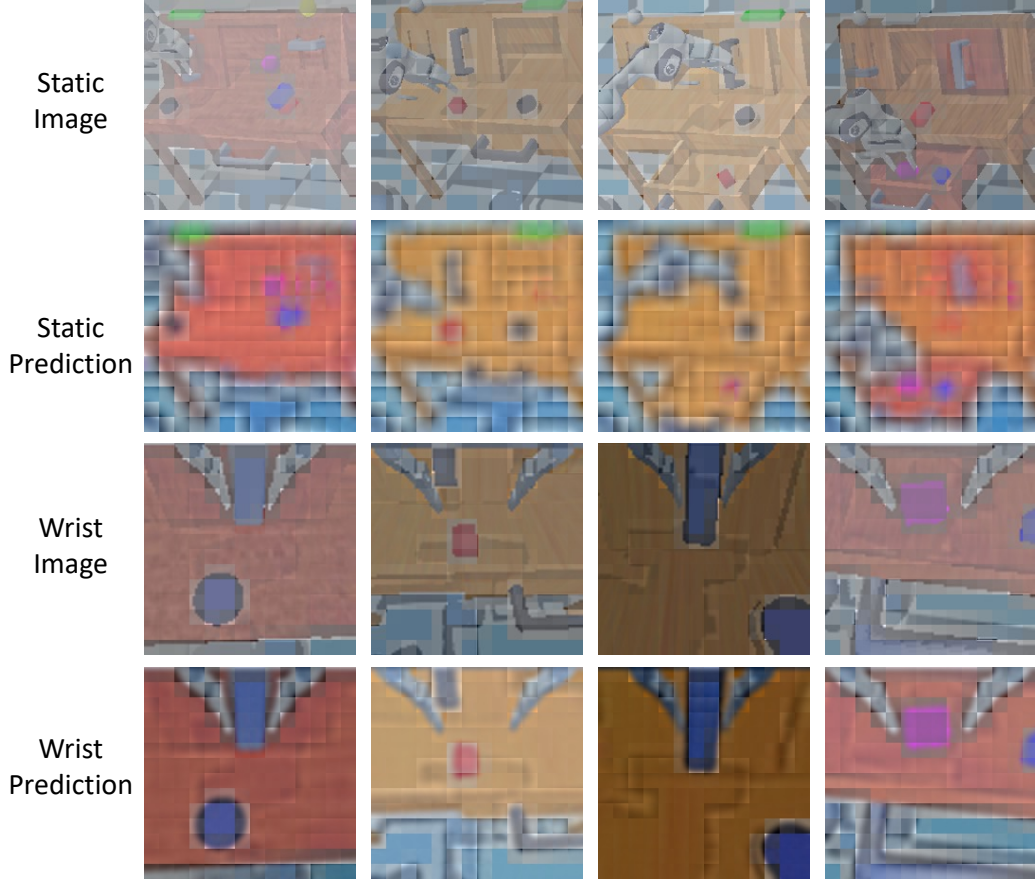


Figure 1: **Visualization results** of the dynamic region predictions.

To ensure a fair comparison and stable convergence, we initialize DreamVLA using the publicly released pre-trained checkpoint of Seer [12]. As reported in Table. 1 in the main manuscript, DreamVLA achieves superior performance across all tasks, substantially outperforming previous state-of-the-art methods.

In particular, our model significantly outperforms two-stage inverse dynamics approaches such as Susie [13], demonstrating the effectiveness of our end-to-end architecture that unifies multimodal prediction and action generation. Compared to CLOVER [14], UP-VLA [15], Seer [12], which also incorporates visual foresight, DreamVLA benefits from a more integrated design and joint optimization, resulting in consistently stronger execution accuracy. Furthermore, our method surpasses video generation-based pretraining approaches like GR-1 [16], highlighting the advantage of coupling visual world modeling with action planning in a single framework.

Notably, the large-scale variant of our model, DreamVLA-Large, achieves an average episode length of **4.45** on the ABC-D split, establishing a new state-of-the-art on the CALVIN benchmark and validating the benefits of predicting future knowledge.

### 2.3 Visualization

As shown in Figure 1 and Figure 2, we visualize the model’s predictions of dynamic regions and depth maps. Although supervision is applied only to dynamic regions, DreamVLA is able to reconstruct semantically meaningful representations of the entire scene. This surprising generalization ability can be attributed to two factors. First, in long-horizon manipulation sequences, the robot arm is in constant motion and frequently interacts with various objects, causing most task-relevant regions to become dynamic at some point in time. This ensures that a large portion of the scene is eventually observed under dynamic supervision. Second, although static regions are not explicitly supervised,

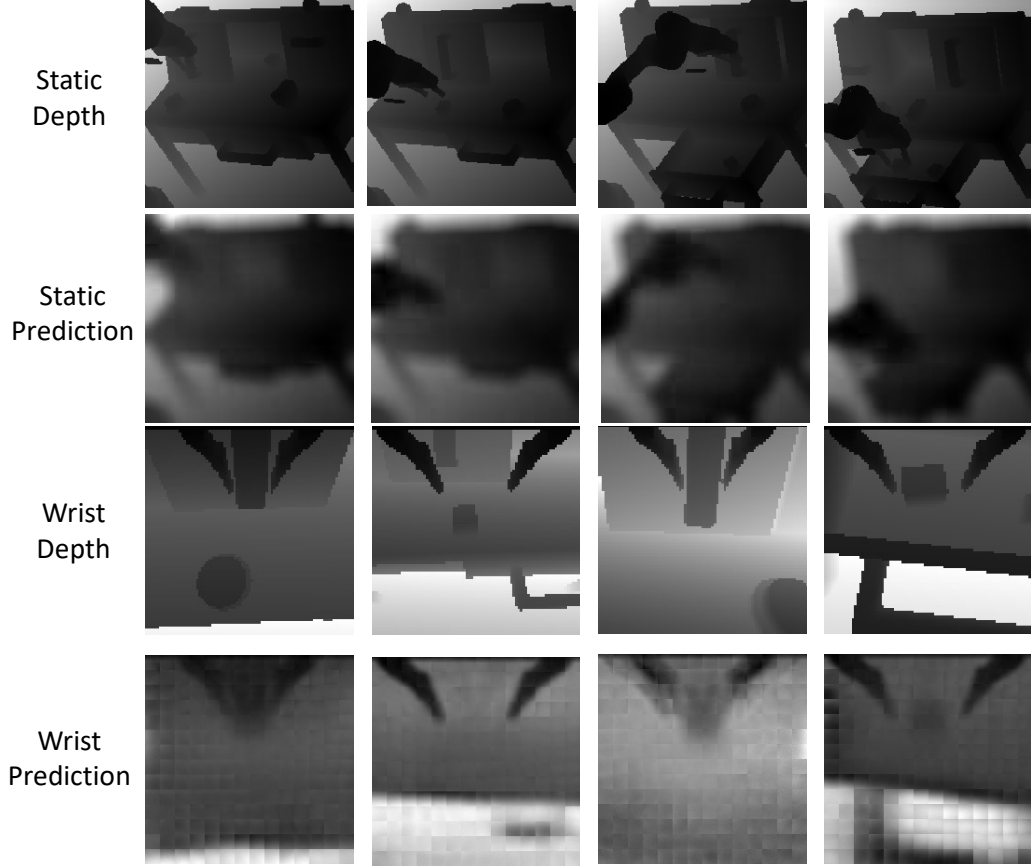


Figure 2: **Visualization results** of the depth maps.

the input frames inherently contain global visual context—including background structures, object appearances, and spatial layout—which the model can leverage to hallucinate and complete missing details. As a result, DreamVLA implicitly learns to integrate temporal dynamics with static priors, leading to coherent and accurate predictions beyond the explicitly labeled regions.

Although the predicted depth maps are relatively coarse due to the patch-level reconstruction inherent in MAE-style decoders [2], they still provide valuable guidance for downstream tasks. In particular, the model benefits from anticipating future depth, which helps refine action decisions and improves spatial awareness.

## 2.4 Real-world Settings

In our real-world training setup, we use a history length of 7, with the model jointly predicting the next 3 future visual representations and action steps. The visual backbone is initialized with a ViT-B model pre-trained using MAE [2], and inference is accelerated using bfloat16 mixed-precision without any observed degradation in task performance. This configuration strikes a balance between computational efficiency and policy stability in manipulation tasks.

For pretraining, we leverage a large-scale dataset such as DROID [17], which contains approximately 76,000 successful robot trajectories collected in diverse settings. For downstream adaptation, we fine-tune the model using 400 task-specific demonstrations. As shown in Figure. 3, we present the qualitative results of real-world experiments.



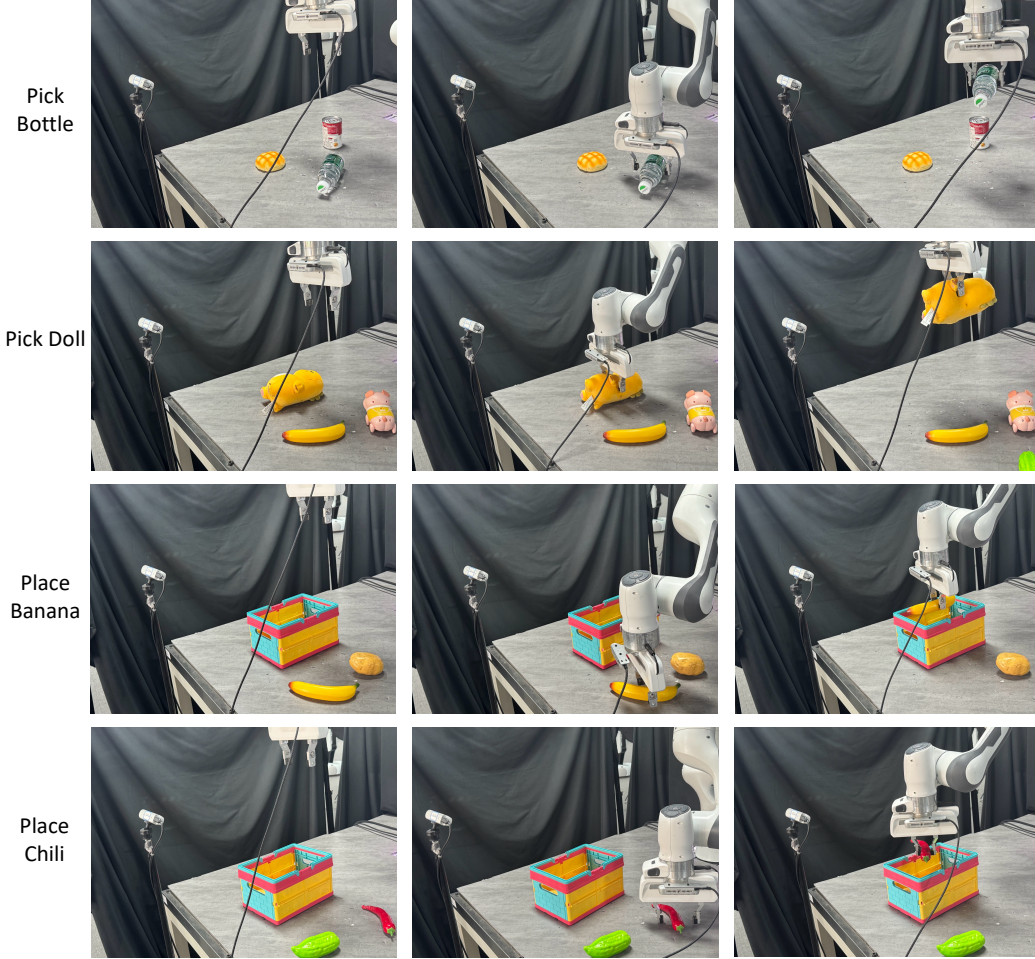


Figure 3: **Qualitative results** of real world language-grounded manipulation.

### 3 Additional Related Works

#### 3.1 Language-Grounded Robot Manipulation

Language-grounded robot Manipulation adopts the human language as a general instruction interface. Existing works can be categorized into two groups: **i)** *End-to-end* models like RT-series [18, 19, 20] built upon unified cross-modal Transformers with tokenized actions [4, 21, 22, 23, 24, 25], large vision-language-action (VLA) models built from VLMs [26], or 3D representations [27, 28, 29]. Training on robot data such as Open X-Embodiment [30] and DROID [17], a remarkable process has been made. However, the data scale is still limited compared to in-the-wild data for training VLMs. **ii)** *Decoupled* high-level reasoning and low-level actions in large VLMs and small off-the-shelf policy models, primitives [31, 32, 33, 34, 35, 36, 37, 38], or articulated priors [39, 40].

### 4 Additional Discussions and Future Work

**i. Scaling Laws.** A promising direction for future exploration involves investigating scaling behavior in DreamVLA. In particular, we plan to study how increasing the capacity of key components—such as the backbone visual encoder or the size of the language model—affects model performance. This includes replacing the current text encoder with larger-scale language models (e.g., LLaMA-2 or GPT variants) to assess the impact of richer linguistic understanding on multimodal reasoning and action generation.

**ii. Integration with Additional Baselines.** We also aim to evaluate DreamVLA in conjunction with more recent and diverse baselines. For example, RoboVLMs [41] incorporate a wide range of vision-language backbones and offer a unified framework for robotic policy learning. Combining DreamVLA with these baselines can help standardize performance comparisons and reveal architectural synergies between pretrained vision-language models and action-centric transformers.

**iii. Contribution of Multi-View Observations.** Our current framework leverages both fixed and egocentric camera views. In future work, we plan to conduct a detailed ablation study to quantify the contribution of each view modality to task performance. This analysis will provide insights into how multi-view information improves spatial reasoning and robustness, especially in occluded or ambiguous scenarios.

**iv. Extension to More Complex and Long-Horizon Tasks.** While DreamVLA demonstrates strong performance on the CALVIN benchmark, we are interested in extending the framework to more complex, long-horizon tasks that involve extended temporal dependencies, delayed rewards, and multi-stage subgoals. This includes evaluating on benchmarks that require sustained interaction, sequential tool use, or high-level planning. Addressing these challenges will require not only more powerful temporal modeling but also better integration of memory, goal abstraction, and hierarchical reasoning mechanisms.

**v. Application to Robotic Navigation and Humanoid.** Beyond tabletop manipulation, DreamVLA could be adapted to robot navigation tasks in indoor or semi-structured environments. By learning to predict dynamic regions, obstacles, and semantic scene components, the model could support instruction-driven navigation and path planning under multimodal supervision, especially in settings where map-based planning is infeasible.

Furthermore, another compelling extension is applying DreamVLA to humanoid robots, which require reasoning over whole-body motion, balance, and physically grounded interactions. The modularity of our framework allows for integration with additional proprioceptive inputs and more complex action spaces. This line of work would explore how multimodal predictive learning can scale to full-body motor control and human-like task execution.

## 5 Broader Impacts

DreamVLA proposes a new training paradigm for vision-language-action (VLA) modeling, going beyond the conventional mapping from visual observations and language to actions. Instead of directly predicting actions from high-dimensional input, our framework first encourages the model to predict comprehensive world knowledge, including depth, dynamic motion, segmentation, and semantic features, before generating actions. This intermediate representation improves action grounding and generalization.

A key strength of DreamVLA lies in its simplicity and efficiency: by adding only a lightweight decoder and a set of learnable queries, we significantly enhance the performance of existing VLA backbones with minimal parameter overhead. This makes the method both scalable and compatible with current VLM-based architectures, paving the way for more robust and transferable policies.

Practically, this design can benefit the development of assistive robots navigation and humanoid, where it is essential for agents to generalize across novel environments and language goals. Furthermore, since our method leverages unlabeled perceptual signals during training, it reduces reliance on curated language-instruction datasets, which are often expensive and domain-specific.

Overall, DreamVLA offers a practical, extensible, and training-efficient framework for improving VLA systems, and we hope it inspires further research into multimodal abstraction and low-cost robot learning.



## References

- [1] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. [1](#)
- [2] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022. [1](#), [6](#)
- [3] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35: 23716–23736, 2022. [1](#)
- [4] Tony Z Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. *arXiv preprint arXiv:2304.13705*, 2023. [1](#), [7](#)
- [5] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. [2](#)
- [6] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision. *Trans. Mach. Learn. Res.*, 2024, 2024. [2](#), [3](#)
- [7] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloé Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross B. Girshick. Segment anything. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 3992–4003. IEEE, 2023. [2](#), [3](#)
- [8] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023. [2](#)
- [9] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Cotracker: It is better to track together. In *European Conference on Computer Vision*, pages 18–35. Springer, 2024. [2](#)
- [10] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *Advances in Neural Information Processing Systems*, 37:21875–21911, 2024. [3](#)
- [11] Oier Mees, Lukas Hermann, Erick Rosete-Beas, and Wolfram Burgard. Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. *IEEE Robotics and Automation Letters*, 7(3):7327–7334, 2022. [4](#)
- [12] Yang Tian, Sizhe Yang, Jia Zeng, Ping Wang, Dahua Lin, Hao Dong, and Jiangmiao Pang. Predictive inverse dynamics models are scalable learners for robotic manipulation. *Int. Conf. Learn. Represent. (ICLR)*, 2024. [4](#), [5](#)
- [13] Kevin Black, Mitsuhiro Nakamoto, Pranav Atreya, Homer Walke, Chelsea Finn, Aviral Kumar, and Sergey Levine. Zero-shot robotic manipulation with pretrained image-editing diffusion models. *arXiv preprint arXiv:2310.10639*, 2023. [5](#)
- [14] Qingwen Bu, Jia Zeng, Li Chen, Yanchao Yang, Guyue Zhou, Junchi Yan, Ping Luo, Heming Cui, Yi Ma, and Hongyang Li. Closed-loop visuomotor control with generative expectation for robotic manipulation. *arXiv preprint arXiv:2409.09016*, 2024. [5](#)
- [15] Jianke Zhang, Yanjiang Guo, Yucheng Hu, Xiaoyu Chen, Xiang Zhu, and Jianyu Chen. Up-vla: A unified understanding and prediction model for embodied agent. *arXiv preprint arXiv:2501.18867*, 2025. [5](#)
- [16] Xinghang Li, Minghuan Liu, Hanbo Zhang, Cunjun Yu, Jie Xu, Hongtao Wu, Chilam Cheang, Ya Jing, Weinan Zhang, Huaping Liu, et al. Vision-language foundation models as effective robot imitators. In *The Twelfth International Conference on Learning Representations*, . [5](#)

- [17] Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024. 6, 7
- [18] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alexander Herzog, Jasmine Hsu, Julian Ibarz, Brian Ichter, Alex Irpan, Tomas Jackson, Sally Jesmonth, Nikhil J. Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Kuang-Huei Lee, Sergey Levine, Yao Lu, Utsav Malla, Deeksha Manjunath, Igor Mordatch, Ofir Nachum, Carolina Parada, Jodilyn Peralta, Emily Perez, Karl Pertsch, Jornell Quiambao, Kanishka Rao, Michael S. Ryoo, Grecia Salazar, Pannag R. Sanketi, Kevin Sayed, Jaspiar Singh, Sumedh Sontakke, Austin Stone, Clayton Tan, Huong T. Tran, Vincent Vanhoucke, Steve Vega, Quan Vuong, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. RT-1: robotics transformer for real-world control at scale. In *Robotics: Science and Systems XIX, Daegu, Republic of Korea, July 10-14, 2023*, 2023. 7
- [19] Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayyaan Wahid, Quan Vuong, Vincent Vanhoucke, Huong T. Tran, Radu Soricut, Anikait Singh, Jaspiar Singh, Pierre Sermanet, Pannag R. Sanketi, Grecia Salazar, Michael S. Ryoo, Krista Reymann, Kanishka Rao, Karl Pertsch, Igor Mordatch, Henryk Michalewski, Yao Lu, Sergey Levine, Lisa Lee, Tsang-Wei Edward Lee, Isabel Leal, Yuheng Kuang, Dmitry Kalashnikov, Ryan Julian, Nikhil J. Joshi, Alex Irpan, Brian Ichter, Jasmine Hsu, Alexander Herzog, Karol Hausman, Keerthana Gopalakrishnan, Chuyuan Fu, Pete Florence, Chelsea Finn, Kumar Avinava Dubey, Danny Driess, Tianli Ding, Krzysztof Marcin Choromanski, Xi Chen, Yevgen Chebotar, Justice Carbajal, Noah Brown, Anthony Brohan, Montserrat Gonzalez Arenas, and Kehang Han. RT-2: vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning, CoRL 2023, 6-9 November 2023, Atlanta, GA, USA*, volume 229 of *Proceedings of Machine Learning Research*, pages 2165–2183. PMLR, 2023. 7
- [20] Suneel Belkhale, Tianli Ding, Ted Xiao, Pierre Sermanet, Quon Vuong, Jonathan Tompson, Yevgen Chebotar, Debidatta Dwibedi, and Dorsa Sadigh. RT-H: action hierarchies using language. *CoRR*, abs/2403.01823, 2024. 7
- [21] Hao Liu, Lisa Lee, Kimin Lee, and Pieter Abbeel. Instruction-following agents with jointly pre-trained vision-language models. *CoRR*, abs/2210.13431, 2022. 7
- [22] Markus Grotz, Mohit Shridhar, Tamim Asfour, and Dieter Fox. Peract2: Benchmarking and learning for robotic bimanual manipulation tasks. *CoRR*, abs/2407.00278, 2024. 7
- [23] Atharva Mete, Haotian Xue, Albert Wilcox, Yongxin Chen, and Animesh Garg. Quest: Self-supervised skill abstractions for learning continuous control. *CoRR*, abs/2407.15840, 2024. 7
- [24] Xinghang Li, Minghuan Liu, Hanbo Zhang, Cunjun Yu, Jie Xu, Hongtao Wu, Chilam Cheang, Ya Jing, Weinan Zhang, Huaping Liu, et al. Vision-language foundation models as effective robot imitators. In *The Twelfth International Conference on Learning Representations*, . 7
- [25] Konstantinos Bousmalis, Giulia Vezzani, Dushyant Rao, Coline Manon Devin, Alex X. Lee, Maria Bauzá Villalonga, Todor Davchev, Yuxiang Zhou, Agrim Gupta, Akhil Raju, Antoine Laurens, Claudio Fantacci, Valentin Dalibard, Martina Zambelli, Murilo Fernandes Martins, Rugile Pevceviciute, Michiel Blokzijl, Misha Denil, Nathan Batchelor, Thomas Lampe, Emilio Parisotto, Konrad Zolna, Scott E. Reed, Sergio Gómez Colmenarejo, Jon Scholz, Abbas Abdolmaleki, Oliver Groth, Jean-Baptiste Regli, Oleg Sushkov, Thomas Rothörl, José Enrique Chen, Yusuf Aytar, Dave Barker, Joy Ortiz, Martin A. Riedmiller, Jost Tobias Springenberg, Raia Hadsell, Francesco Nori, and Nicolas Heess. Robocat: A self-improving generalist agent for robotic manipulation. *Trans. Mach. Learn. Res.*, 2024, 2024. 7
- [26] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024. 7
- [27] Shizhe Chen, Ricardo Garcia Pinel, Cordelia Schmid, and Ivan Laptev. Polarnet: 3d point clouds for language-guided robotic manipulation. In *Conference on Robot Learning, CoRL 2023, 6-9 November 2023, Atlanta, GA, USA*, volume 229 of *Proceedings of Machine Learning Research*, pages 1761–1781. PMLR, 2023. 7

- [28] Haoyu Zhen, Xiaowen Qiu, Peihao Chen, Jincheng Yang, Xin Yan, Yilun Du, Yining Hong, and Chuang Gan. 3d-vla: A 3d vision-language-action generative world model. *arXiv preprint arXiv:2403.09631*, 2024. 7
- [29] Wentao Yuan, Jiafei Duan, Valts Blukis, Wilbert Pumacay, Ranjay Krishna, Adithyavairavan Murali, Arsalan Mousavian, and Dieter Fox. Robopoint: A vision-language model for spatial affordance prediction for robotics. *CoRR*, abs/2406.10721, 2024. 7
- [30] Abby O’Neill, Abdul Rehman, Abhinav Gupta, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, et al. Open x-embodiment: Robotic learning datasets and rt-x models. *arXiv preprint arXiv:2310.08864*, 2023. 7
- [31] Brian Ichter, Anthony Brohan, Yevgen Chebotar, Chelsea Finn, Karol Hausman, Alexander Herzog, Daniel Ho, Julian Ibarz, Alex Irpan, Eric Jang, Ryan Julian, Dmitry Kalashnikov, Sergey Levine, Yao Lu, Carolina Parada, Kanishka Rao, Pierre Sermanet, Alexander Toshev, Vincent Vanhoucke, Fei Xia, Ted Xiao, Peng Xu, Mengyuan Yan, Noah Brown, Michael Ahn, Omar Cortes, Nicolas Sievers, Clayton Tan, Sichun Xu, Diego Reyes, Jarek Rettinghouse, Jornell Quiambao, Peter Pastor, Linda Luu, Kuang-Huei Lee, Yuheng Kuang, Sally Jesmonth, Nikhil J. Joshi, Kyle Jeffrey, Rosario Jauregui Ruano, Jasmine Hsu, Keerthana Gopalakrishnan, Byron David, Andy Zeng, and Chuyuan Kelly Fu. Do as I can, not as I say: Grounding language in robotic affordances. In *Conference on Robot Learning, CoRL 2022, 14-18 December 2022, Auckland, New Zealand*, volume 205 of *Proceedings of Machine Learning Research*, pages 287–318. PMLR, 2022. 7
- [32] Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, Pierre Sermanet, Tomas Jackson, Noah Brown, Linda Luu, Sergey Levine, Karol Hausman, and Brian Ichter. Inner monologue: Embodied reasoning through planning with language models. In *Conference on Robot Learning, CoRL 2022, 14-18 December 2022, Auckland, New Zealand*, volume 205 of *Proceedings of Machine Learning Research*, pages 1769–1782. PMLR, 2022. 7
- [33] Jacky Liang, Wenlong Huang, Fei Xia, Peng Xu, Karol Hausman, Brian Ichter, Pete Florence, and Andy Zeng. Code as policies: Language model programs for embodied control. In *IEEE International Conference on Robotics and Automation, ICRA 2023, London, UK, May 29 - June 2, 2023*, pages 9493–9500. IEEE, 2023. 7
- [34] Wenlong Huang, Chen Wang, Ruohan Zhang, Yunzhu Li, Jiajun Wu, and Li Fei-Fei. Voxposer: Composable 3d value maps for robotic manipulation with language models. In *Annu. Conf. Robot. Learn. (CoRL)*, 2023. 7
- [35] Wenlong Huang, Fei Xia, Dhruv Shah, Danny Driess, Andy Zeng, Yao Lu, Pete Florence, Igor Mordatch, Sergey Levine, Karol Hausman, and Brian Ichter. Grounded decoding: Guiding text generation with grounded models for embodied agents. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. 7
- [36] Haoxu Huang, Fanqi Lin, Yingdong Hu, Shengjie Wang, and Yang Gao. Copa: General robotic manipulation through spatial constraints of parts with foundation models. *CoRR*, abs/2403.08248, 2024. 7
- [37] Kuan Fang, Fangchen Liu, Pieter Abbeel, and Sergey Levine. MOKA: Open-World Robotic Manipulation through Mark-Based Visual Prompting. In *Proceedings of Robotics: Science and Systems*, Delft, Netherlands, July 2024. 7
- [38] Zekun Qi, Wen Yao Zhang, Yufei Ding, Runpei Dong, Xinqiang Yu, Jingwen Li, Lingyun Xu, Baoyu Li, Xialin He, Guofan Fan, et al. Sofar: Language-grounded orientation bridges spatial reasoning and object manipulation. *arXiv preprint arXiv:2502.13143*, 2025. 7
- [39] Siyuan Huang, Haonan Chang, Yuhan Liu, Yimeng Zhu, Hao Dong, Abdeslam Boularias, Peng Gao, and Hongsheng Li. A3VLM: actionable articulation-aware vision language model. In Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard, editors, *Conference on Robot Learning, 6-9 November 2024, Munich, Germany*, volume 270 of *Proceedings of Machine Learning Research*, pages 1675–1690. PMLR, 2024. 7
- [40] Xiaoqi Li, Mingxu Zhang, Yiran Geng, Haoran Geng, Yuxing Long, Yan Shen, Renrui Zhang, Jiaming Liu, and Hao Dong. Manipllm: Embodied multimodal large language model for

- 401 object-centric robotic manipulation. In *IEEE/CVF Conference on Computer Vision and Pattern*  
402 *Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 18061–18070. IEEE,  
403 2024. [7](#)
- 404 [41] Xinghang Li, Peiyan Li, Minghuan Liu, Dong Wang, Jirong Liu, Bingyi Kang, Xiao Ma, Tao  
405 Kong, Hanbo Zhang, and Huaping Liu. Towards generalist robot policies: What matters in  
406 building vision-language-action models. *arXiv preprint arXiv:2412.14058*, 2024. [8](#)