FLARE: Robot Learning with Implicit World Modeling

Ruijie Zheng[1,2*], **Jing Wang**[1,3*], **Scott Reed**[1*]
Johan Bjorck[1†], **Yu Fang**[1†], **Fengyuan Hu**[1†], **Joel Jang**[1†], **Kaushil Kundalia**[1†]
Zongyu Lin[1†], **Loic Magne**[1†], **Avnish Narayan**[1†], **You Liang Tan**[1†], **Guanzhi Wang**[1†]
Qi Wang[1†], **Jiannan Xiang**[1†], **Yinzhen Xu**[1†], **Seonghyeon Ye**[1†]
Jan Kautz[1], **Furong Huang**[2], **Yuke Zhu**[1,4‡], **Linxi Fan**[1‡]
[1]NVIDIA   [2]University of Maryland, College Park
[3]Nanyang Technological University   [4]University of Texas, Austin
[*]equal contribution   [†]alphabetical order   [‡]equal advising

https://research.nvidia.com/labs/gear/flare

Abstract: We introduce *Future LAtent REpresentation Alignment (***FLARE***)*, a novel framework that integrates predictive latent world modeling into robot policy learning. By aligning features from a diffusion transformer with latent embeddings of future observations, **FLARE** enables a diffusion transformer policy to anticipate latent representations of future observations, allowing it to reason about long-term consequences while generating actions. Remarkably lightweight, **FLARE** requires only minimal architectural modifications—adding a few tokens to standard vision-language-action (VLA) models—yet delivers substantial performance gains. Across two challenging multitask simulation imitation learning benchmarks spanning single-arm and humanoid tabletop manipulation, **FLARE** achieves state-of-the-art performance, outperforming prior policy learning baselines by up to 26%. Moreover, **FLARE** unlocks the ability to co-train with human egocentric video demonstrations without action labels, significantly boosting policy generalization to a novel object with unseen geometry with as few as a single robot demonstration. Our results establish **FLARE** as a general and scalable approach for combining implicit world modeling with high-frequency robotic control.

Keywords: World Model, VLA, Humanoid Robotics

1 Introduction

Human cognitive processes involve sophisticated predictive capabilities that operate largely implicitly. Consider a common action such as reaching for a coffee mug on a cluttered desk: without thinking about it, human brains could predict how the hand will move, what obstacles it might encounter, and how the mug will feel when grasped. This capacity to construct internal representations of future states, a form of world modeling, is fundamental to efficient human motor control and decision-making.

Several recent works [1, 2, 3, 4, 5, 6] have explored jointly learning world models and policies by generating future visual frames in parallel with actions. While intuitive, this approach faces notable practical and conceptual challenges. High-fidelity visual prediction typically requires large-scale generative models, introducing significant computational overhead and latency. Moreover, optimizing simultaneously for pixel-level reconstruction and action prediction places competing demands on model capacity: visual generation emphasizes detailed spatial fidelity and texture synthesis, whereas action modeling benefits from compact, abstract, task-relevant representations, often leading to diluted learning efficiency. In this work, we show that a surprisingly simple and flexible recipe, fully compatible with existing VLA architectures, can surpass prior VLA policy learning methods by a substantial margin.
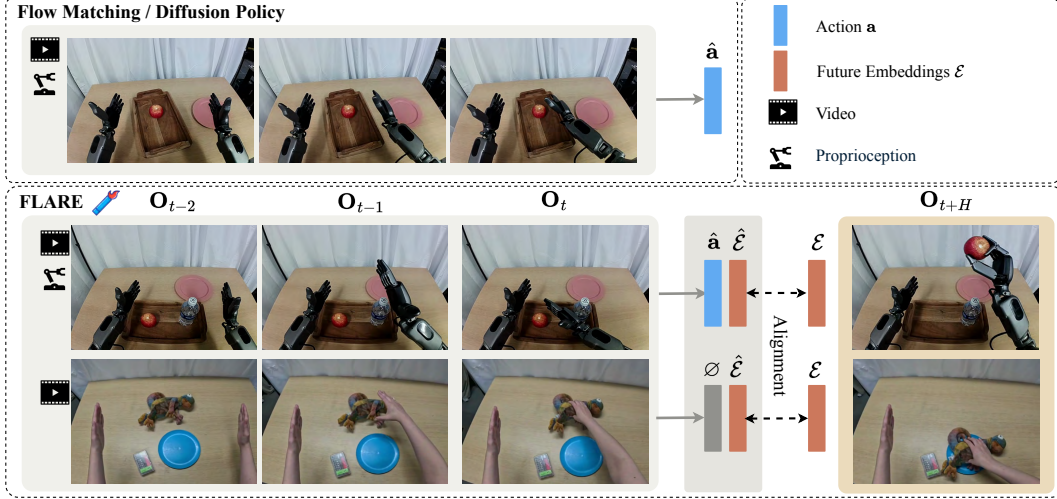
Figure 1: Comparison of **FLARE** to a conventional flow-matching (or diffusion) policy. **FLARE** can train using both action flow-matching and future latent alignment objectives, leading to improved performance as well as enabling learning from video-only data such as human ego-view demonstrations.

We introduce **F**uture **LA**tent **RE**presentation Alignment (**FLARE**), a lightweight yet highly effective extension to diffusion or flow-matching policies that introduces latent-space world modeling via a future alignment objective, eliminating the need for full-frame reconstruction. At its core, **FLARE** predicts a compact representation of the robot's future observation from the hidden states of the action denoising network. **FLARE** operates in two key stages. First, we pretrain a compact, action-aware observation embedding model. While general-purpose embedding models could be used for the target future embeddings, we find that an action-aware embedding explicitly optimized for downstream control tasks offers superior performance and efficiency due to its compactness and task alignment. Next, we co-train the diffusion transformer by introducing a minimal set of additional tokens, which are optimized to predict the future observation embeddings. This approach requires minimal modifications to existing VLA architectures [7, 8], making it broadly applicable and easy to deploy.

Despite its simplicity, **FLARE** achieves state-of-the-art performance across two multitask imitation-learning benchmarks spanning single-arm and humanoid tabletop manipulation. Notably, when trained on diverse cross-embodiment robot data, our action-aware embedding model generalizes effectively to unseen embodiment and tasks. With just 100 trajectories per task collected on a real GR1 humanoid posttrained from our pretrained action-aware observation embedding model, the **FLARE** policy achieves a 95% success rate in real-world evaluations. Finally, **FLARE** enables learning from action-free data sources, such as human videos. By leveraging GoPro-collected human egocentric video demonstrations and only a single real robot demonstration per object, **FLARE** successfully learns novel grasping strategies, highlighting its potential for scalable robot learning from less structured data sources.

2   Background

In this work, following π_0 and GR00T N1 [7, 8], we adopt **flow-matching** [9] as the learning objective for fitting actions from human demonstrations. Let o_t denote the robot's observation, which includes image inputs (potentially from multiple views) and a language instruction; let q_t be the robot's proprioceptive state; and let $A_t = (a_t, \ldots, a_{t+H})$ be an action chunk drawn from expert demonstrations. We define $\phi_t = VL(o_t)$ as the vision-language embedding of the observation.

Given the VL embedding ϕ_t, an action chunk A_t, a flow-matching timestep $\tau \in [0, 1]$, and sampled noise $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, we construct the noised action chunk as:
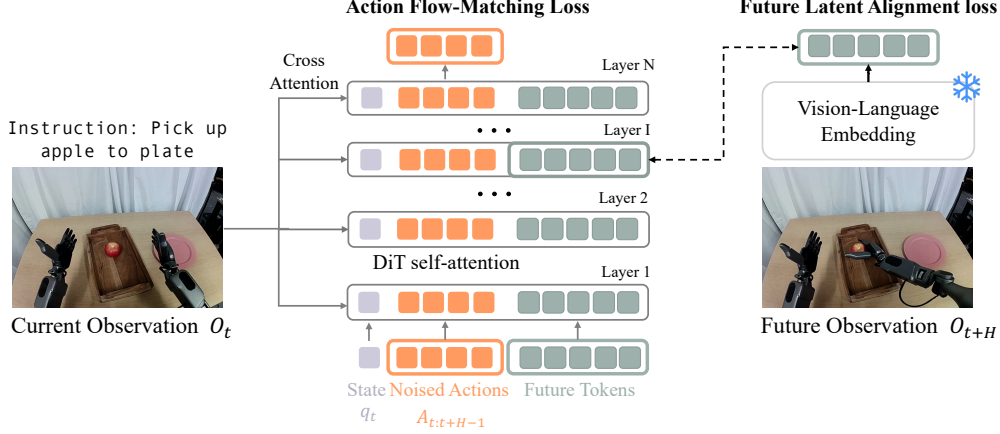
$$A_t^\tau = \tau A_t + (1 - \tau)\epsilon.$$

Figure 2: FLARE architecture. State and action token embeddings are concatenated into a sequence with learnable future token embeddings. The flow matching DiT blocks perform self-attention on this sequence, and cross-attention to the current vision and text observation embeddings. At a middle layer, the activations corresponding to the future token embeddings are used to compute a future latent alignment loss, which is the cosine similarity with vision-language embeddings from a future observation.

Then the model prediction $V_\theta(\phi_t, A_t^\tau, q_t)$ is trained to approximate the denoising direction $\epsilon - A_t$, by minimizing the following flow-matching loss:

$$\mathcal{L}_{fm}(\theta) = \mathbb{E}_\tau \left[\|V_\theta(\phi_t, A_t^\tau, q_t) - (\epsilon - A_t)\|^2 \right]. \tag{1}$$

We sample the timestep τ from the distribution $p(\tau) = \text{Beta}\left(\frac{s-\tau}{s}; 1.5, 1\right)$ with $s = 0.999$ as in Black et al. [7]. At inference time, we generate action chunks via K-step denoising. We first sample an initial chunk $A_t^0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and then apply forward Euler integration to iteratively refine it:

$$A_t^{\tau+1/K} = A_t^\tau + \frac{1}{K} V_\theta(\phi_t, A_t^\tau, q_t).$$

Following GR00T N1 [8], we set $K = 4$ throughout all of our experiments, and we use the same Diffusion Transformer (DiT) architecture [10] for V_θ with alternating cross-attention and self-attention layers to condition on the robot's vision language embedding ϕ_t.

3 Method

3.1 Latent World Modeling through Future Latent Representation Alignment

To enable the latent representation within the DiT blocks to predict future latent states, we add M learnable future token embeddings to the input sequence, such that the sequence contains three components: (1) the current proprioceptive state q_t encoded via a state encoder, (2) noised action chunk $A_t^\tau = \{\tau a_t + (1 - \tau)\epsilon\}_t^{t+H}$ encoded by an action encoder, and (3) a set of M learnable future tokens. Next, we slice out the intermediate DiT representations corresponding to the M future tokens at an internal layer L, project those features using an MLP, and finally align these with the frozen vision-language embeddings of the future observation ϕ_{t+H} (see Figure 2).

Our approach is similar to how Representation Alignment (REPA) [11] is applied to improve text-to-image diffusion models, but with several important differences arising from the setting of latent world modeling. First, we align a DiT policy with *future* embeddings, rather than embeddings of the current observation. Second, our architecture adds learnable future tokens, so that the flow matching and alignment proceed along separate streams within the DiT, which interact via self-attention.

In this way, we encourage the DiT modules to internally reason about the future latent state while maintaining their action prediction capability through action flow-matching. Letting B indicate batch dimension and D indicate embedding dimension, we can write the latent alignment objective as

$$\mathcal{L}_{align}(\theta) = -\mathbb{E}_\tau \left[cos(f_\theta(\phi_t, A_t^\tau, q_t), g(\phi_{t+H})) \right] \tag{2}$$

3

where $f_\theta \to \mathbb{R}^{B \times M \times D}$ outputs the DiT activations for the M future tokens at layer L, and $g \to \mathbb{R}^{B \times M \times D}$ is the encoder of the future observation ϕ_{t+H}. The overall loss function is

$$\mathcal{L} = \mathcal{L}_{fm} + \lambda \mathcal{L}_{align} \tag{3}$$

Empirically, we found $\lambda = 0.2$ worked the best in our experiments. We refer the readers to Section 4.4 for a detailed analysis of this choice.

3.2 Action-aware Future Embedding Model

While our future latent alignment framework is broadly compatible with various embedding models, we find that incorporating an *action-aware* future embedding yields further improvements in both performance and efficiency. To this end, we propose a compact vision-language embedding of the robot's current observation, explicitly optimized for policy learning. The design objective is twofold: achieving **compactness** while ensuring **action-awareness**.

Specifically, we leverage both the vision and text encoders from SigLIP-2 [12] to encode the robot's image observations and text instructions. The encoded tokens are then fused using four layers of self-attention transformer blocks to capture cross-modal dependencies. Subsequently, we apply a Q-former [13] module to compress the fused sequence into $M = 32$ learnable query tokens, producing a compact, fixed-size representation that naturally generalizes to multi-camera inputs. To ensure action-awareness, we train the vision language embedding end-to-end with the regular action flow-matching objective to predict the robot's actions by attaching 8 DiT blocks. In this way, all task-relevant information is guaranteed to be captured within the latent token embeddings.

To pretrain the embedding model, we leverage a diverse mixture of cross-embodiment robot datasets, comprising both simulated and real-world humanoid tabletop manipulation data from GR00T N1 [8] and seven additional datasets from Open X-Embodiment [14], totaling approximately 2,000 hours of robotic data. Following pretraining, we posttrain the downstream policy jointly with the latent world model and the action prediction objective across downstream domains and tasks. Specifically, for posttraining, we initialize the downstream policy's encoder with the pretrained embedding model, while also using the pretrained

Figure 3: Data mixture of pretrained action-aware vision language embedding model

embedding model to define the prediction targets for future latent representations. To mitigate distribution shifts between pretraining and downstream visual observations, rather than keeping the embedding model entirely frozen, we adopt an exponential moving average (EMA) update with respect to the policy's encoder. This strategy allows the embedding model to gradually adapt in tandem with the evolving vision and language encoders during policy fine-tuning. Empirically, we find that an EMA update rate of 0.995 performs the best. We refer the readers to Section 4.4 for a detailed analysis of this choice.
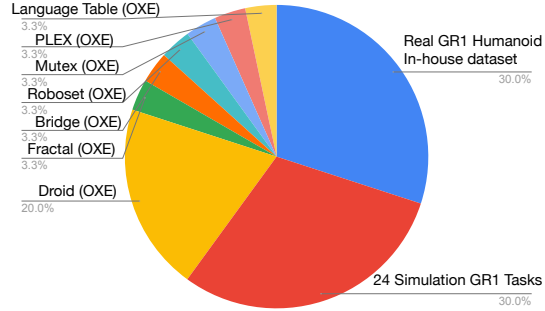
4 Experiments

4.1 Multitask Benchmark Performance

In this section, we evaluate our latent world model on two multitask benchmarks that cover both single-arm manipulation and bimanual humanoid tabletop manipulation tasks. For the single-arm manipulation benchmark, we adopt RoboCasa [15], consisting of 24 atomic tasks in a simulated kitchen environment, including pick-and-place, door manipulation, faucet operation, and more. Robot's observations include three RGB images captured from cameras mounted on the left, right,

Methods	FLARE	Policy Only	UWM	GR00T N1 (Scratch)	Diffusion Policy
Pick and Place	**53.2%**	43.8%	35.6%	44.1%	29.2%
Open & Close Doors / Drawers	**88.8%**	78.7%	82.0%	80.0%	78.7%
Others	**80.0%**	75.2%	74.2%	69.6%	61.3%
24 RoboCasa Tasks Average	**70.1%**	61.9%	60.8%	60.6%	51.7%
Pick and Place Tasks	**56.2%**	42.9%	26.5%	45.9%	37.8%
Articulated Tasks	**51.3%**	47.4%	38.4%	42.8%	50.1%
24 GR1 Tasks Average	**55.0%**	44.0%	29.5%	45.1%	40.9%

Table 1: Task Success Rate Breakdown for Multitask Policy on RoboCasa and GR1 Tabletop Manipulation

RoboCasa (24 tasks w/ Panda Arm)

Close Doors   Turn Off Stove   Pick and Place   Setup Mug   Turn On Microwave

Humanoid Tabletop Manipulation (24 tasks w/ GR1)

Bottle To Cabinet   Bread To Pan   Cup To Drawer   Cupcake To Box   Potato To Shelf
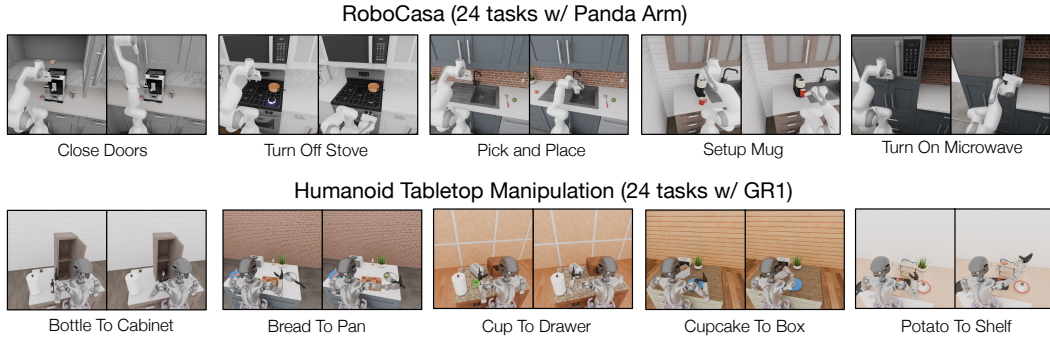
Figure 4: Multitask Simulation Benchmarks: We use 24 RoboCasa [15] and 24 GR-1 tabletop manipulation tasks as a multitask simulation benchmark suite in this paper.

and wrist of the robot. Next, we incorporate 24 GR-1 tabletop simulation tasks from GR00T N1 [8], which emphasize dexterous hand control with the GR-1 humanoid robot. This suite includes 18 object rearrangement tasks—picking up and placing objects between source and target containers—and 6 tasks involving interaction with articulated objects such as cabinets, drawers, and microwaves. Observation consists of a single RGB image from an egocentric camera positioned on the robot's head.

To ensure a fair comparison between our method and the baseline, for experiments in this section, we do not use the pretrained embedding model mentioned in Section 3.2. Instead, we pretrain the embedding model exclusively on the same in-domain multitask dataset for 80,000 gradient steps, ensuring that any performance gains cannot be attributed to pretraining data with the embedding model. In particular, we include the following baselines for the experimental results:

1. Diffusion Policy [16]: Diffusion Policy models action distributions via a diffusion-based generative process, rather than using flow matching. It uses a U-Net architecture that progressively denoises random noise to generate the final action.

2. UWM [4]: We select UWM as the main baseline for methods that jointly learn video and action prediction objectives. UWM predicts image VAE latents and actions jointly with a diffusion objective.

3. GR00T N1 (Scratch) [8]: Since GR00T N1 is pretrained on a much broader data mixture, we ensure a fair comparison by using the same architecture but initializing the DiT layers from scratch, while only loading the pretrained Eagle VLM [17] model weights.

4. **FLARE** with Policy Only: We use the exact same model architecture as **FLARE**, as mentioned in Section 3.2, but train it solely with the policy learning objective.

All methods are trained for 80,000 gradient steps on the multitask robot dataset, except for UWM. We noticed that UWM performance is still improving at the end of 80k gradient steps, and thus we extend its training to 400k steps—five times the training budget allocated to the other methods. Following GR00T N1 [8], we evaluate each model checkpoint for 50 episodes per task every 1000 gradient steps, and report the maximum success rate over the final five checkpoints for each method.
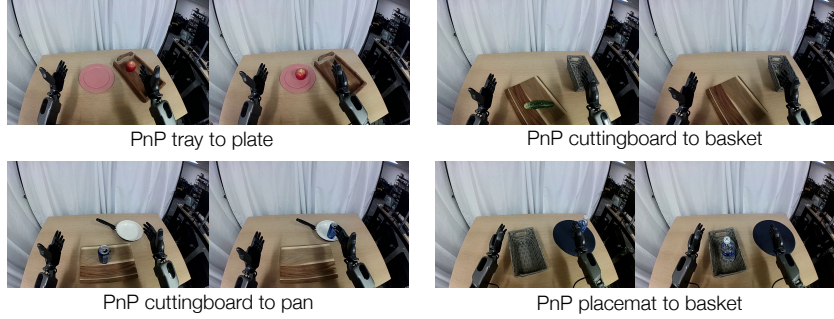
PnP tray to plate   PnP cuttingboard to basket

PnP cuttingboard to pan   PnP placemat to basket

Figure 5: Real GR1 Tasks Setup: We evaluate four tabletop manipulation tasks on a real GR1 humanoid robot.

Figure 6: (**Left**): Post-training results on 24 RoboCasa tasks. (**Right**): Post-training results on 4 Real GR1 humanoid tasks.

As shown in Table 1, we draw two key observations. First, **FLARE** consistently outperforms all baseline methods, including both the policy-only baselines and UWM. This highlights the strength of our compact, action-aware latent world modeling objective in enabling more effective policy learning. Additionally, in our experiments, we also observe that **FLARE** with the policy-only objective, trained for 160k gradient steps, achieves only 44.1% success rate, resulting in no performance difference compared with 80k gradient steps. Thus, the improved results cannot simply be attributed to more training steps with **FLARE**. Second, even when trained with only the policy objective, FLARE still achieves performance on par with GR00T N1 initialized from scratch, despite GR00T N1 using a larger VLM backbone. This result underscores the quality of our Q-former-based vision-language embedding model in capturing action-relevant information.

4.2 Data-efficient Post-training with Cross-embodiment Pretrained Embedding Model

While the latent world model demonstrates substantial performance gains, as shown in the previous section, it requires training a separate embedding model for each domain. In this section, we evaluate **FLARE** with the pretrained embedding model mentioned in Section 3.2 as the future prediction target, focusing on unseen embodiments and tasks with data-limited posttraining settings. Specifically, we select 24 RoboCasa arm tasks and 4 real-world GR1 humanoid tabletop manipulation tasks as the evaluation benchmarks, and post-train the policy jointly with the latent world model and policy objectives, comparing it against a baseline that is post-trained using only the policy objective. In particular, for the policy-only baseline, we initialize both the Q-former-based vision language embedding and the policy's DiT model weights from the cross-embodiment pretrained model. For **FLARE**, we only warm start the vision language embedding model.

For the evaluation protocol, we follow the same procedure described in Section 4.1 for the 24 RoboCasa tasks. For the 4 real-world GR-1 tasks shown in Figure 5, we define 8 reference initial frames per task, each involving 4 distinct objects (apple, can, bottled water, cucumber) to manipulate, and report the success rate of the final policy checkpoint for each method.
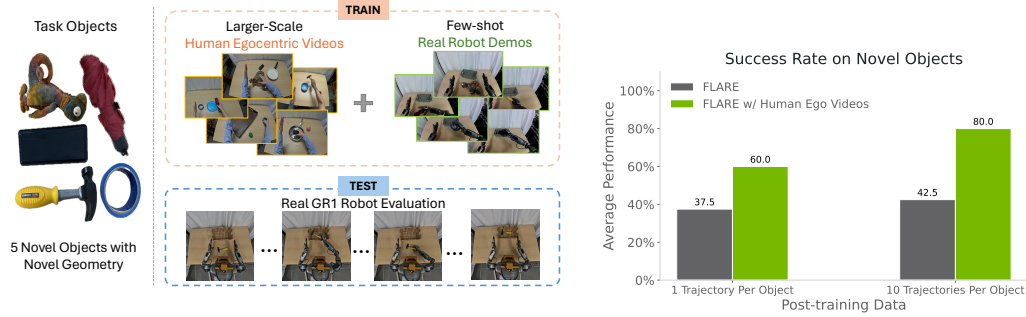
Figure 7: Generalizing to unseen objects with human egocentric videos and few-shot real robot demos

As shown in Figure 6, across both the 24 RoboCasa simulation tasks and the real-world GR-1 humanoid tasks, **FLARE** consistently outperforms the policy-only baseline. The improvement is especially pronounced under limited data conditions, achieving a 10% gain on RoboCasa with 100 trajectories per task for posttraining. Notably, although the pretrained embedding model has never seen RoboCasa tasks during pretraining, using it as the future embedding achieves comparable performance with 1000 trajectories to an embedding model trained exclusively on the 24 RoboCasa arm tasks (71.3% vs. 70.2% as reported in Section 4.1).

On the real GR-1 humanoid robot, we achieve a success rate of up to **95.1%**, averaging 14% higher than the baseline method. Qualitatively, we observe that in scenarios where a can or water bottle is placed close to the robot's hand, the baseline method trained with only the policy objective often knocks over the object. In contrast, **FLARE** policy learns to maneuver around or over the object and successfully grasp, highlighting the benefits of future latent reasoning enabled by **FLARE**.

4.3 Leveraging Human Egocentric Trajectories without Action Labels

While our previous experiments demonstrate that the proposed future latent alignment objective significantly enhances policy performance when trained on action-labeled data, we further show that it can be naturally extended to trajectories without action annotations, such as human egocentric demonstrations. This setting is particularly attractive, as collecting human demonstrations is substantially more cost-effective and efficient than teleoperating a robot to execute the same tasks.

To evaluate this, we select five novel objects with distinctive geometries that are absent from the training dataset, each requiring novel grasping strategies. For instance, the blue tape object is large and thus requires a top-down grasp by the robot hand. For each object, we collect 150 human egocentric demonstrations per object by mounting a GoPro on the demonstrator's head while they perform similar tasks as the humanoid robot. On the robot side, we collect only 10 teleoperated demonstrations per object and train the policy using a mixture of these limited demonstrations, our GR-1 pretraining dataset, and the egocentric human videos.

For real-robot demonstrations with actions, we apply both the action flow-matching loss and the future alignment objective. In contrast, for the human egocentric videos without action labels, we rely solely on the future alignment loss to learn the latent dynamics. At evaluation time, we select five initial poses as reference images for each object and measure the robot's success rate. Partial credit (0.5) is given when the robot successfully grasps the object but fails to place it into the basket.

As shown in Figure 7, with only **1** teleoperated trajectory per object, **FLARE** already achieves up to a 60% success rate on novel objects. When provided with 10 trajectories per object, and jointly trained with human videos, **FLARE** further improves to an 80% success rate—roughly doubling the performance of a baseline trained solely on action-labeled data. These results highlight that **FLARE** not only enhances learning from action-labeled demonstrations, but also effectively leverages unlabeled human demonstrations to improve generalization by capturing latent task dynamics.

4.4 Ablation Study

Using the Pretrained Siglip2 as Future Embedding model: While leveraging a policy-oriented future embedding model results in strong policy performance and enhanced training efficiency, we also explore an alternative setting that employs pretrained SigLIP2-Large vision tokens at timestep $t + 16$ as prediction targets. Specifically, we experiment using both raw SigLIP2 vision tokens (256 tokens per image) and 2×2 average-pooled tokens (64 tokens per image). As illustrated in Table 2, our **FLARE** framework maintains compatibility with diverse teacher encoder models beyond the policy-oriented embedding model. Although we get the optimal performance with the embedding model pretrained specifically on the target domain, using a more general-purpose vision encoder such as SigLIP2 still yields a significant 7% improvement over baseline methods.

Method	Success Rate (%)
No FLARE loss	43.9
SigLIP2	49.6
SigLIP2 (Average Pooled)	50.9
Action-aware Embedding	**55.0**

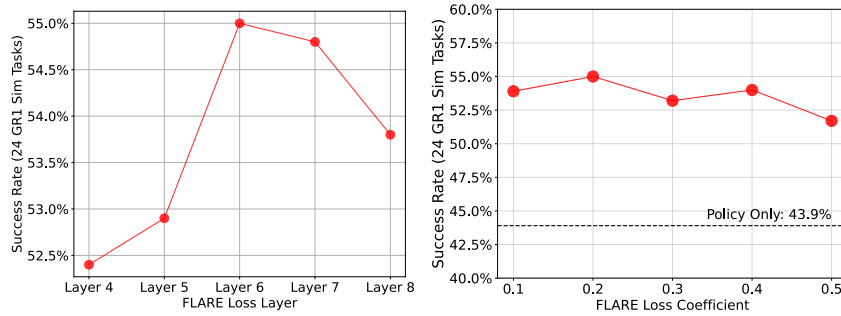**Table 2:** Ablation of target embedding models.

Figure 8: (**Left**): Ablation of the DiT Layer used in FLARE loss (**Left**): Ablation of FLARE loss coefficient.

Index of FLARE Loss Layer and Coefficient of FLARE Loss: A key design decision in **FLARE** is selecting the DiT layer at which to apply the future latent alignment loss, and the coefficient λ of **FLARE** loss. In our main experiments, we apply this objective at layer 6 out of 8 total layers in the DiT architecture. Applying it at deeper layers allows a larger portion of the model weights to benefit from the supervision of future latent prediction, but may also lead to conflicts between the action prediction and future alignment objectives. To evaluate the effect of these two hyperparameters, we evaluate **FLARE** on the GR1 simulation benchmark with different layer indexes and coefficients used for alignment. As shown in Figure 8, the model maintains strong performance across a range of hyperparameter setups. However, we do notice that applying the alignment objective too early—*e.g.*, at layer 4—leads to a notable drop in performance, highlighting the importance of aligning the future prediction objective with the action denoising process.
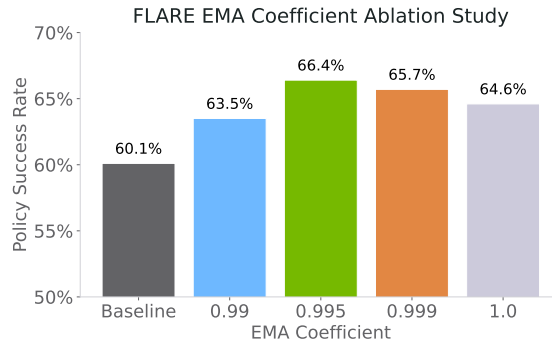

Figure 9: Effect of EMA Coefficient ρ: We report the policy success rate using 24×300 training trajectories across 24 RoboCasa tasks. Baseline is trained without **FLARE** future alignment loss, *i.e.*, a policy-only objective.

8

Exponential Moving Average (EMA) of Pretrained Action-aware Embedding Model: As discussed in Section 3.2, to address the distribution shift between pretraining and downstream tasks for our action-aware vision-language target embedding model, we incorporate an exponential moving average (EMA) update. Specifically, at each gradient step, the target embedding model parameters are updated as follows:

$$\theta_{\text{target\_vl\_embedding}} \leftarrow \rho\theta_{\text{target\_embedding}} + (1 - \rho)\theta_{\text{policy\_vl\_embedding}}$$

The EMA update enables the prediction target to adapt slowly in tandem with the evolving policy encoder, providing stability across training. Here, We evaluate several choices of the EMA coefficient $\rho \in \{0.99, 0.995, 0.999, 1.0\}$, each using 24×300 trajectories to train the **FLARE** policy. The final average success rates are reported in Figure 9. We find that while all EMA variants outperform the baseline method without **FLARE** future latent alignment objective, $\rho = 0.995$ yields the best performance and is used in all experiments. Notably, even with $\rho = 1.0$ (*i.e.*, no EMA), **FLARE** still surpasses the baseline, whereas $\rho = 0.99$ performs the worst, likely due to the instability caused by frequent target updates.

5 Related Work

Generative World Models for Robotics: There has been a rich body of research on world models for robotics, ranging from model-based control to model-based reinforcement learning [18, 19, 20, 21, 22, 23]. More recently, with advances in image and video generation, several works have explored the integration of generative modeling into policy learning [1, 2, 6, 4, 3, 24]. One line of work [6, 25] uses image diffusion models with inverse dynamics models to close the perception-to-action loop. The GR1 and GR2 families introduce end-to-end models that jointly predict discrete image tokens and actions using a unified next-token prediction objective. Other approaches [4, 3, 26, 27, 28, 29, 30] instead aim to jointly predict continuous image latents and actions. For instance, UWM [4] and UVA [3] jointly denoise VAE latents of future frames along with robot actions. DINO-WM [26] utilizes DINO features [26] to train a latent dynamics model for model-based planning.

Our work builds upon recent advances in representation learning, particularly Representation Alignment [11], which has shown remarkable success in accelerating the convergence of diffusion transformers for image generation and is key to state-of-the-art flow-matching models like Seedream-3.0 [31]. However, our approach differs in two crucial ways: we train a flow-matching *policy* rather than an image model, and we align the DiT representation with features from *future* observations rather than current ones. In contrast to existing works, **FLARE** introduces an implicit latent world model objective that bypasses explicit reconstruction of future frames or latents. This simple design enables reasoning over a compact, action-aware latent space and avoids the computational burden of high-fidelity generation, while maintaining compatibility with standard VLA architectures, without requiring major architectural redesign. While DINO-WM focuses on zero-shot planning, **FLARE** is designed for policy and world model co-training, though planning could be a valuable future extension.

Vision Language Action Models. A growing body of recent work [32, 33, 7, 34, 35, 36, 37, 38, 39, 40, 41, 42] has focused on developing general-purpose foundational vision-language-action (VLA) models by fine-tuning vision-language models for downstream robotics tasks. Among these works, models such as [34, 42, 43, 44] autoregressively predict sequences of discrete action tokens using the next-token prediction objective. In contrast, methods like [45, 7, 8] leverage diffusion-based or flow-matching policy heads to bridge pretrained VLMs with continuous action generation. In this work, inspired by the architecture of GR00T-N1 [8], we adopt a flow-matching policy head built with diffusion transformer blocks, using interleaved self-attention and cross-attention layers to condition on the fused vision-language embeddings.

Learning from Egocentric Videos. Several approaches have sought to enhance robot learning by leveraging human egocentric videos. These efforts extract diverse forms of information, such as human-object interactions [46], object affordances [47, 48, 49, 50], and visual trace trajectories [51, 52]. Other lines of work aim to translate human motions into robotic behaviors using hand

pose estimators [53, 54, 50, 55, 56, 57] or motion capture systems [58]. In this work, we show that future latent alignment provides a lightweight and effective alternative that does not require explicit pose estimators or point tracking tools, maximally reducing the engineering efforts. A complementary direction focuses on learning latent actions from visual deltas between current and future frames to guide downstream policy learning [41, 59, 60, 61, 62, 63]. Unlike latent actions as intermediate representations, whose correlation with ground-truth actions is unclear, our action-aware vision-language embedding directly aligns with future observations, resulting in a simple yet effective framework that naturally captures all the temporal dynamics information essential for effective policy learning.

6 Limitations

In this work, we focus mainly on imitation learning with pick-and-place tasks on a real humanoid robot. Extending to more complex humanoid tasks that require more fine-grained dexterous manipulation, and incorporating reinforcement learning into the training paradigm, remains an important direction for future work. Moreover, although our method enables generalization to novel objects, it still relies on a small number of expert demonstrations, which may limit scalability in settings where such data is hard to acquire. Additionally, in this paper, we focus on egocentric human video datasets collected in controlled settings using head-mounted GoPro cameras. Extending to more diverse and larger-scale egocentric motion datasets captured in natural environments becomes a promising future direction of our work.

7 Conclusion

We present Future Latent Representation Alignment (**FLARE**), a simple yet effective framework for jointly learning robot policy and latent world dynamics. By aligning the future representations of the robot's observations with the hidden states of the action denoising network, **FLARE** enables the policy to implicitly reason about future states while predicting actions. This approach leads to state-of-the-art performance on challenging robotic manipulation benchmarks. Furthermore, **FLARE** unlocks co-training with human egocentric video demonstrations that lack action labels, significantly improving generalization to novel objects with minimal real-robot teleoperation data.

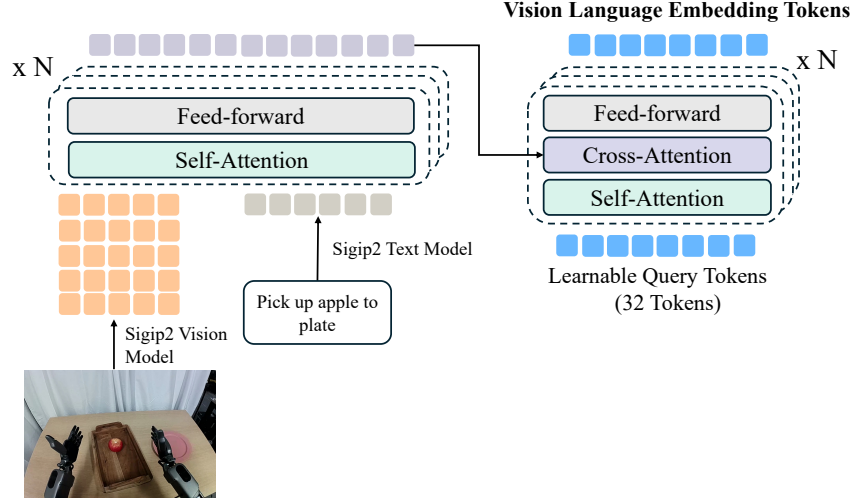# A Details of Q-former based Vision Language Embedding Module

Figure 10: Our Q-former-based Vision Language Embedding Module

We present the architectural details of our compact Q-former-based vision-language embedding module. Specifically, we adopt `siglip2-large-patch16-256` as the backbone for both vision and language encoders. The SigLIP2 vision encoder processes 256×256 resolution robot images into 256 patch tokens, while the language encoder encodes padded robot instructions into 32 language tokens. These 256 vision tokens and 32 language tokens are concatenated and passed through four layers of self-attention transformers to yield 288 fused vision-language tokens. To obtain a compact representation, we apply a Q-former architecture [13], where 32 learnable query tokens—randomly initialized—interact with the 288 fused tokens through interleaved self-attention and cross-attention layers, producing 32 compressed vision-language tokens.

B Pretraining Data Mixture

Details of pretraining data mixture are presented in Table 3.

Table 3: Action-Aware Vision Language Embedding Pre-training Dataset Statistics

Dataset	Length (Frames)	Duration (hr)	FPS	Camera View	Category
GR-1 In-house Dataset	6.4M	88.4	20	Egocentric	Real robot
DROID (OXE) [64]	23.1M	428.3	15	Left, Right, Wrist	Real robot
RT-1 (OXE) [32]	3.7M	338.4	3	Egocentric	Real robot
Language Table (OXE) [65]	7.0M	195.7	10	Front-facing	Real robot
Bridge-v2 (OXE) [66]	2.0M	111.1	5	Shoulder, left, right, wrist	Real robot
MUTEX (OXE) [67]	362K	5.0	20	Wrist	Real robot
Plex (OXE) [68]	77K	1.1	20	Wrist	Real robot
RoboSet (OXE) [69]	1.4M	78.9	5	Left, Right, Wrist	Real robot
GR-1 Simulation	125.5M	1,742.6	20	Egocentric	Simulation
Total	169.5M	2,989.5	–	–	–

C Training Details

For the pretraining of the action-aware vision language embedding module, we use 256 NVIDIA H100 GPUs with a batch size of 8192 for 150,000 gradient steps. We use AdamW [70] optimizer with $\beta_1 = 0.95$, $\beta_2 = 0.999$, and $\epsilon = 1\text{e-}8$. A weight decay of 1e-5 is applied, and the learning rate

follows a cosine scheduling strategy with a warmup ratio of 0.05. Following [7, 8], we sample the flowmatching denoising timestep from $p(\tau) = \text{Beta}(\frac{s-\tau}{s}; 1.5, 1)$, $s = 0.999$.

For the multitask experiments of **FLARE** conducted in Sections 4.1 and 4.2, we use 32 NVIDIA H100 GPUS with batch size 1024 for 80,000 gradient steps, while keeping the rest of the hyperparameter setups exactly the same.

D Pseudocode of FLARE

Here we present a Python-style pseudocode of **FLARE** loss calculation as well as the entire training loop.

Algorithm 1 Python-style pseudocode for FLARE training

```
# target_vl_embedding: pretrained action-aware vision language embedding
# vl_embedding:  vision language embedding of the current policy
# dit: diffusion transformer of the current policy
# action_embedding: 2-layer MLP to embed noisy actions
# state_embedding:  2-layer MLP to embed priorioceptive state
# action_decode:    2-layer MLP to decode robot's actions
# embedding_decode: 2-layer MLP to decode predicted embeddings
# I: Index of the layer to apply FLARE loss
# N: Number of gradient steps
# M: Number of tokens in VL
# lambda: coefficient of FLARE loss (default is 0.2)

### Initialization
future_tokens = nn.Embedding(M, hiddem_dim)
vl_embedding.load_state_dict(vl_embedding.state_dict())
target_vl_embedding.requires_grad = False

for n in range(N):
    obs, proprio, actions, future_obs = dataset.next()

    ### Prepare noisy action inputs
    noise = gaussian.sample()
    timestep = beta.sample() # sample flowmatching timestep
    noisy_action = timestep * actions + (1-timestep) * noise
    velocity = actions - noise

    ### Get state, action, and observation embedding tokens
    action_tokens = action_embed(noisy_action, timestep)
    state_token   = state_embed(state)
    vl_tokens     = vl_embedding(obs)

    ### Pass through DiT layers
    sa_tokens  = torch.concat([state_token, action_tokens, future_tokens], dim=1)
    policy_outputs = dit(sa_tokens, vl_tokens, output_hidden_states=True)

    ### Calculate action flowmatching loss based on final layer output
    action_outputs = action_decoder(policy_outputs[-1][:, 1:1 + action_tokens.shape[1]])
    action_loss = MSE(action_outputs, velocity)

    ### Calculate FLARE loss based on layer I
    with torch.no_grad():
        embedding_to_align = target_vl_embedding(future_obs)
    predict_embedding = decode_embedding(policy_outputs[I][:, -M:])
    flare_loss = 1-COSINE_SIMILARITY(predict_embedding, embedding_to_align)

    ### Optimize the combined loss
    loss = action_loss + lambda * flare_loss
    optimizer.zero_grad()
    loss.backward()
    optimizer.step()
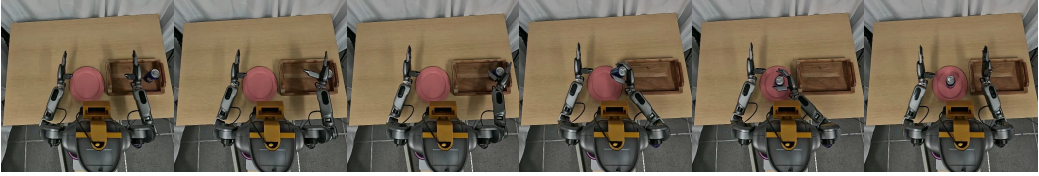```

---

# E Real GR1 Humanoid Rollouts

## E.1 4 Pick-and-place Tasks

Below, we present policy rollouts from the **FLARE** trained policy on 4 real-world GR1 humanoid pick-and-place tasks, together with the task's language instructions. Qualitatively, we observe that when manipulating objects such as a bottled water or a Coke can, the **FLARE** policy learns to maneuver the hand around the object, hovering over the water bottle, rather than striking and knocking it over.

pick up bottled water to basket



pick up can to plate



pick up cucumber to basket



pick up apple to pan



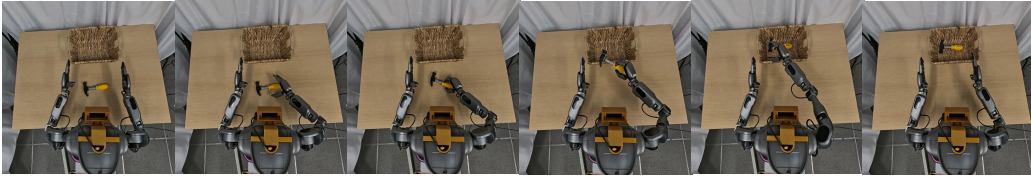Figure 11: **FLARE** policy rollout on real GR1 humanoid robot with 4 pick-and-place tasks

## E.2 Manipulating Novel Objects

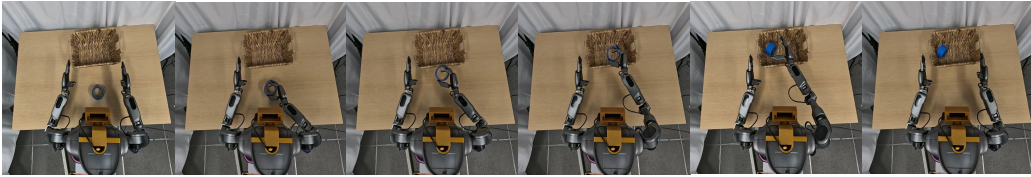Below, we present policy rollouts from the **FLARE** trained policy manipulating 5 novel objects.

pick up stuffed toy to basket

pick up hammer to plate

pick up blue tape to basket

pick up blackboard eraser to pan
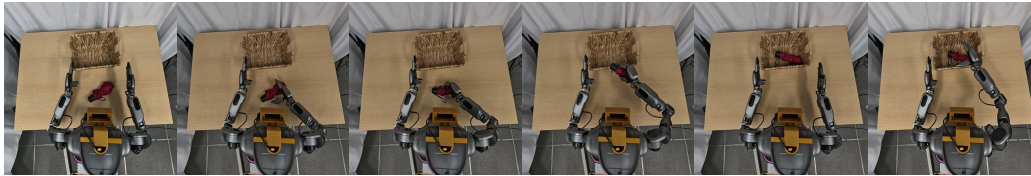
pick up umbrella to pan

Figure 12: **FLARE** policy rollout manipulating 5 novel objects

# References

[1] 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 *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=NxoFmGgWC9.

[2] C.-L. Cheang, G. Chen, Y. Jing, T. Kong, H. Li, Y. Li, Y. Liu, H. Wu, J. Xu, Y. Yang, H. Zhang, and M. Zhu. Gr-2: A generative video-language-action model with web-scale knowledge for robot manipulation. *arXiv preprint arXiv:2410.06158*, 2024.

[3] S. Li, Y. Gao, D. Sadigh, and S. Song. Unified video action model. *arXiv preprint arXiv:2503.00200*, 2025.

[4] C. Zhu, R. Yu, S. Feng, B. Burchfiel, P. Shah, and A. Gupta. Unified world models: Coupling video and action diffusion for pretraining on large robotic datasets. 2025.

[5] Q. Zhao, Y. Lu, M. J. Kim, Z. Fu, Z. Zhang, Y. Wu, Z. Li, Q. Ma, S. Han, C. Finn, A. Handa, M.-Y. Liu, D. Xiang, G. Wetzstein, and T.-Y. Lin. Cot-vla: Visual chain-of-thought reasoning for vision-language-action models, 2025. URL https://arxiv.org/abs/2503.22020.

[6] Y. Du, S. Yang, B. Dai, H. Dai, O. Nachum, J. B. Tenenbaum, D. Schuurmans, and P. Abbeel. Learning universal policies via text-guided video generation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=bo8q5MRcwy.

[7] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter, et al. $\pi_0$: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

[8] NVIDIA, :, J. Bjorck, F. Castañeda, N. Cherniadev, X. Da, R. Ding, L. J. Fan, Y. Fang, D. Fox, F. Hu, S. Huang, J. Jang, Z. Jiang, J. Kautz, K. Kundalia, L. Lao, Z. Li, Z. Lin, K. Lin, G. Liu, E. Llontop, L. Magne, A. Mandlekar, A. Narayan, S. Nasiriany, S. Reed, Y. L. Tan, G. Wang, Z. Wang, J. Wang, Q. Wang, J. Xiang, Y. Xie, Y. Xu, Z. Xu, S. Ye, Z. Yu, A. Zhang, H. Zhang, Y. Zhao, R. Zheng, and Y. Zhu. Gr00t n1: An open foundation model for generalist humanoid robots, 2025. URL https://arxiv.org/abs/2503.14734.

[9] Y. Lipman, R. T. Chen, H. Ben-Hamu, M. Nickel, and M. Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*.

[10] W. Peebles and S. Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.

[11] S. Yu, S. Kwak, H. Jang, J. Jeong, J. Huang, J. Shin, and S. Xie. Representation alignment for generation: Training diffusion transformers is easier than you think. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=DJSZGGZYVi.

[12] M. Tschannen, A. Gritsenko, X. Wang, M. F. Naeem, I. Alabdulmohsin, N. Parthasarathy, T. Evans, L. Beyer, Y. Xia, B. Mustafa, et al. SigLIP 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025.

[13] J. Li, D. Li, S. Savarese, and S. Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 19730–19742. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/li23q.html.

[14] Open X-Embodiment Collaboration et al. Open X-Embodiment: Robotic learning datasets and RT-X models. International Conference on Robotics and Automation, 2024.

[15] S. Nasiriany, A. Maddukuri, L. Zhang, A. Parikh, A. Lo, A. Joshi, A. Mandlekar, and Y. Zhu. Robocasa: Large-scale simulation of everyday tasks for generalist robots. In *Robotics: Science and Systems (RSS)*, 2024.

[16] 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*, 2024.

[17] Z. Li, G. Chen, S. Liu, S. Wang, V. VS, Y. Ji, S. Lan, H. Zhang, Y. Zhao, S. Radhakrishnan, et al. Eagle 2: Building post-training data strategies from scratch for frontier vision-language models. *arXiv preprint arXiv:2501.14818*, 2025.

[18] X. Jiang, Q. Chen, S. Han, M. Li, J. Dong, and R. Zhang. When to trust your model: Model-based policy optimization, 2020. URL https://openreview.net/forum?id=SkgPIpcGar. Submitted to NeurIPS 2019 Reproducibility Challenge.

[19] D. Hafner, T. Lillicrap, M. Norouzi, and J. Ba. Mastering atari with discrete world models. *arXiv preprint arXiv:2010.02193*, 2020.

[20] N. Hansen, X. Wang, and H. Su. Temporal difference learning for model predictive control. 2022.

[21] J. Cheng, D. Kang, G. Fadini, G. Shi, and S. Coros. Rambo: Rl-augmented model-based optimal control for whole-body loco-manipulation, 2025. URL https://arxiv.org/abs/2504.06662.

[22] X. Wang, R. Zheng, Y. Sun, R. Jia, W. Wongkamjan, H. Xu, and F. Huang. COPlanner: Plan to roll out conservatively but to explore optimistically for model-based RL. In *NeurIPS 2023 Workshop on Generalization in Planning*, 2023. URL https://openreview.net/forum?id=9lkkqGagDF.

[23] R. Zheng, X. Wang, H. Xu, and F. Huang. Is model ensemble necessary? model-based RL via a single model with lipschitz regularized value function. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=hNyJBk3CwR.

[24] Y. Du, S. Yang, P. Florence, F. Xia, A. Wahid, brian ichter, P. Sermanet, T. Yu, P. Abbeel, J. B. Tenenbaum, L. P. Kaelbling, A. Zeng, and J. Tompson. Video language planning. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=9pKtcJcMP3.

[25] S. Huang, M. Levy, Z. Jiang, A. Anandkumar, Y. Zhu, L. Fan, D.-A. Huang, and A. Shrivastava. Ardup: Active region video diffusion for universal policies, 2025. URL https://arxiv.org/abs/2406.13301.

[26] G. Zhou, H. Pan, Y. LeCun, and L. Pinto. Dino-wm: World models on pre-trained visual features enable zero-shot planning. *arXiv preprint arXiv:2411.04983*, 2024.

[27] M. Schwarzer, N. Rajkumar, M. Noukhovitch, A. Anand, L. Charlin, R. D. Hjelm, P. Bachman, and A. C. Courville. Pretraining representations for data-efficient reinforcement learning. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 12686–12699. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/69eba34671b3ef1ef38ee85caae6b2a1-Paper.pdf.

[28] M. Schwarzer, A. Anand, R. Goel, R. D. Hjelm, A. Courville, and P. Bachman. Data-efficient reinforcement learning with self-predictive representations. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=uCQfPZwRaUu.

[29] R. Zheng, X. Wang, Y. Sun, S. Ma, J. Zhao, H. Xu, H. Daumé III, and F. Huang. Taco: Temporal latent action-driven contrastive loss for visual reinforcement learning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 48203–48225. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/96d00450ed65531ffe2996daed487536-Paper-Conference.pdf.

[30] R. Zheng, Y. Liang, X. Wang, S. Ma, H. Daumé III, H. Xu, J. Langford, P. Palanisamy, K. S. Basu, and F. Huang. Premier-taco is a few-shot policy learner: pretraining multitask representation via temporal action-driven contrastive loss. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

[31] Y. Gao, L. Gong, Q. Guo, X. Hou, Z. Lai, F. Li, L. Li, X. Lian, C. Liao, L. Liu, et al. Seedream 3.0 technical report. *arXiv preprint arXiv:2504.11346*, 2025.

[32] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu, J. Ibarz, B. Ichter, A. Irpan, T. Jackson, S. Jesmonth, N. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, I. Leal, K.-H. Lee, S. Levine, Y. Lu, U. Malla, D. Manjunath, I. Mordatch, O. Nachum, C. Parada, J. Peralta, E. Perez, K. Pertsch, J. Quiambao, K. Rao, M. Ryoo, G. Salazar, P. Sanketi, K. Sayed, J. Singh, S. Sontakke, A. Stone, C. Tan, H. Tran, V. Vanhoucke, S. Vega, Q. Vuong, F. Xia, T. Xiao, P. Xu, S. Xu, T. Yu, and B. Zitkovich. Rt-1: Robotics transformer for real-world control at scale. In *arXiv preprint arXiv:2212.06817*, 2022.

[33] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn, P. Florence, C. Fu, M. G. Arenas, K. Gopalakrishnan, K. Han, K. Hausman, A. Herzog, J. Hsu, B. Ichter, A. Irpan, N. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, I. Leal, L. Lee, T.-W. E. Lee, S. Levine, Y. Lu, H. Michalewski, I. Mordatch, K. Pertsch, K. Rao, K. Reymann, M. Ryoo, G. Salazar, P. Sanketi, P. Sermanet, J. Singh, A. Singh, R. Soricut, H. Tran, V. Vanhoucke, Q. Vuong, A. Wahid, S. Welker, P. Wohlhart, J. Wu, F. Xia, T. Xiao, P. Xu, S. Xu, T. Yu, and B. Zitkovich. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *arXiv preprint arXiv:2307.15818*, 2023.

[34] M. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi, Q. Vuong, T. Kollar, B. Burchfiel, R. Tedrake, D. Sadigh, S. Levine, P. Liang, and C. Finn. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

[35] R. Zheng, Y. Liang, S. Huang, J. Gao, H. D. III, A. Kolobov, F. Huang, and J. Yang. TraceVLA: Visual trace prompting enhances spatial-temporal awareness for generalist robotic policies. In *The Thirteenth International Conference on Learning Representations*, 2025.

[36] J. Wen, Y. Zhu, J. Li, M. Zhu, K. Wu, Z. Xu, R. Cheng, C. Shen, Y. Peng, F. Feng, et al. Tinyvla: Towards fast, data-efficient vision-language-action models for robotic manipulation. *arXiv preprint arXiv:2409.12514*, 2024.

[37] C.-L. Cheang, G. Chen, Y. Jing, T. Kong, H. Li, Y. Li, Y. Liu, H. Wu, J. Xu, Y. Yang, H. Zhang, and M. Zhu. Gr-2: A generative video-language-action model with web-scale knowledge for robot manipulation. *arXiv preprint arXiv:2410.06158*, 2024.

[38] X. Li, M. Liu, H. Zhang, C. Yu, J. Xu, H. Wu, C. Cheang, Y. Jing, W. Zhang, H. Liu, et al. Vision-language foundation models as effective robot imitators. *arXiv preprint arXiv:2311.01378*, 2023.

[39] H. Zhen, X. Qiu, P. Chen, J. Yang, X. Yan, Y. Du, Y. Hong, and C. Gan. 3d-vla: 3d vision-language-action generative world model. *arXiv preprint arXiv:2403.09631*, 2024.

[40] J. Huang, S. Yong, X. Ma, X. Linghu, P. Li, Y. Wang, Q. Li, S.-C. Zhu, B. Jia, and S. Huang. An embodied generalist agent in 3d world. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2024.

[41] S. Ye, J. Jang, B. Jeon, S. J. Joo, J. Yang, B. Peng, A. Mandlekar, R. Tan, Y.-W. Chao, B. Y. Lin, L. Liden, K. Lee, J. Gao, L. Zettlemoyer, D. Fox, and M. Seo. Latent action pretraining from videos. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=VYOe2eBQeh.

[42] J. Yang, R. Tan, Q. Wu, R. Zheng, B. Peng, Y. Liang, Y. Gu, M. Cai, S. Ye, J. Jang, Y. Deng, L. Liden, and J. Gao. Magma: A foundation model for multimodal ai agents, 2025. URL https://arxiv.org/abs/2502.13130.

[43] 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, 2025. URL https://arxiv.org/abs/2501.09747.

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

[45] Octo Model Team, D. Ghosh, H. Walke, K. Pertsch, K. Black, O. Mees, S. Dasari, J. Hejna, C. Xu, J. Luo, T. Kreiman, Y. Tan, L. Y. Chen, P. Sanketi, Q. Vuong, T. Xiao, D. Sadigh, C. Finn, and S. Levine. Octo: An open-source generalist robot policy. In *Proceedings of Robotics: Science and Systems*, Delft, Netherlands, 2024.

[46] J. Zeng, Q. Bu, B. Wang, W. Xia, L. Chen, H. Dong, H. Song, D. Wang, D. Hu, P. Luo, et al. Learning manipulation by predicting interaction. *arXiv preprint arXiv:2406.00439*, 2024.

[47] S. Bahl, R. Mendonca, L. Chen, U. Jain, and D. Pathak. Affordances from human videos as a versatile representation for robotics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[48] A. Kannan, K. Shaw, S. Bahl, P. Mannam, and D. Pathak. Deft: Dexterous fine-tuning for real-world hand policies. *arXiv preprint arXiv:2310.19797*, 2023.

[49] M. K. Srirama, S. Dasari, S. Bahl, and A. Gupta. Hrp: Human affordances for robotic pre-training. *arXiv preprint arXiv:2407.18911*, 2024.

[50] K. Shaw, S. Bahl, and D. Pathak. Videodex: Learning dexterity from internet videos. In *Conference on Robot Learning*, 2023.

[51] C. Wen, X. Lin, J. So, K. Chen, Q. Dou, Y. Gao, and P. Abbeel. Any-point trajectory modeling for policy learning. *arXiv preprint arXiv:2401.00025*, 2023.

[52] H. Bharadhwaj, R. Mottaghi, A. Gupta, and S. Tulsiani. Track2act: Predicting point tracks from internet videos enables diverse zero-shot robot manipulation. *arXiv e-prints*, pages arXiv–2405, 2024.

[53] C. Wang, L. Fan, J. Sun, R. Zhang, L. Fei-Fei, D. Xu, Y. Zhu, and A. Anandkumar. Mimicplay: Long-horizon imitation learning by watching human play. *arXiv preprint arXiv:2302.12422*, 2023.

[54] Y. Zhu, A. Lim, P. Stone, and Y. Zhu. Vision-based manipulation from single human video with open-world object graphs. *arXiv preprint arXiv:2405.20321*, 2024.

[55] H. Bharadhwaj, A. Gupta, S. Tulsiani, and V. Kumar. Zero-shot robot manipulation from passive human videos. *arXiv preprint arXiv:2302.02011*, 2023.

[56] J. Ye, J. Wang, B. Huang, Y. Qin, and X. Wang. Learning continuous grasping function with a dexterous hand from human demonstrations. *IEEE Robotics and Automation Letters*, 8(5): 2882–2889, 2023.

[57] Y. Qin, Y.-H. Wu, S. Liu, H. Jiang, R. Yang, Y. Fu, and X. Wang. Dexmv: Imitation learning for dexterous manipulation from human videos. In *European Conference on Computer Vision*, 2022.

[58] J. Yang, Z.-a. Cao, C. Deng, R. Antonova, S. Song, and J. Bohg. Equibot: Sim (3)-equivariant diffusion policy for generalizable and data efficient learning. *arXiv preprint arXiv:2407.01479*, 2024.

[59] J. Bruce, M. Dennis, A. Edwards, J. Parker-Holder, Y. Shi, E. Hughes, M. Lai, A. Mavalankar, R. Steigerwald, C. Apps, Y. Aytar, S. Bechtle, F. Behbahani, S. Chan, N. Heess, L. Gonzalez, S. Osindero, S. Ozair, S. Reed, J. Zhang, K. Zolna, J. Clune, N. de Freitas, S. Singh, and T. Rocktäschel. Genie: Generative interactive environments, 2024. URL https://arxiv.org/abs/2402.15391.

[60] Y. Chen, Y. Ge, W. Tang, Y. Li, Y. Ge, M. Ding, Y. Shan, and X. Liu. Moto: Latent motion token as the bridging language for learning robot manipulation from videos, 2025. URL https://arxiv.org/abs/2412.04445.

[61] D. Schmidt and M. Jiang. Learning to act without actions. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=rvUq3cxpDF.

[62] Z. Ren, Y. Wei, X. Guo, Y. Zhao, B. Kang, J. Feng, and X. Jin. Videoworld: Exploring knowledge learning from unlabeled videos, 2025. URL https://arxiv.org/abs/2501.09781.

[63] Q. Bu, J. Cai, L. Chen, X. Cui, Y. Ding, S. Feng, S. Gao, X. He, X. Huang, S. Jiang, et al. Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. *arXiv preprint arXiv:2503.06669*, 2025.

[64] A. Khazatsky, K. Pertsch, S. Nair, A. Balakrishna, S. Dasari, S. Karamcheti, S. Nasiriany, M. K. Srirama, L. Y. Chen, K. Ellis, P. D. Fagan, J. Hejna, M. Itkina, M. Lepert, Y. J. Ma, P. T. Miller, J. Wu, S. Belkhale, S. Dass, H. Ha, A. Jain, A. Lee, Y. Lee, M. Memmel, S. Park, I. Radosavovic, K. Wang, A. Zhan, K. Black, C. Chi, K. B. Hatch, S. Lin, J. Lu, J. Mercat, A. Rehman, P. R. Sanketi, A. Sharma, C. Simpson, Q. Vuong, H. R. Walke, B. Wulfe, T. Xiao, J. H. Yang, A. Yavary, T. Z. Zhao, C. Agia, R. Baijal, M. G. Castro, D. Chen, Q. Chen, T. Chung, J. Drake, E. P. Foster, J. Gao, D. A. Herrera, M. Heo, K. Hsu, J. Hu, D. Jackson, C. Le, Y. Li, K. Lin, R. Lin, Z. Ma, A. Maddukuri, S. Mirchandani, D. Morton, T. Nguyen, A. O'Neill, R. Scalise, D. Seale, V. Son, S. Tian, E. Tran, A. E. Wang, Y. Wu, A. Xie, J. Yang, P. Yin, Y. Zhang, O. Bastani, G. Berseth, J. Bohg, K. Goldberg, A. Gupta, A. Gupta, D. Jayaraman, J. J. Lim, J. Malik, R. Martín-Martín, S. Ramamoorthy, D. Sadigh, S. Song, J. Wu, M. C. Yip, Y. Zhu, T. Kollar, S. Levine, and C. Finn. Droid: A large-scale in-the-wild robot manipulation dataset. 2024.

[65] C. Lynch, A. Wahid, J. Tompson, T. Ding, J. Betker, R. Baruch, T. Armstrong, and P. Florence. Interactive language: Talking to robots in real time, 2022.

[66] H. Walke, K. Black, A. Lee, M. J. Kim, M. Du, C. Zheng, T. Zhao, P. Hansen-Estruch, Q. Vuong, A. He, V. Myers, K. Fang, C. Finn, and S. Levine. Bridgedata v2: A dataset for robot learning at scale. In *Conference on Robot Learning (CoRL)*, 2023.

[67] R. Shah, R. Martín-Martín, and Y. Zhu. Mutex: Learning unified policies from multimodal task specifications. In *7th Annual Conference on Robot Learning*, 2023.

[68] G. Thomas, C.-A. Cheng, R. Loynd, F. V. Frujeri, V. Vineet, M. Jalobeanu, and A. Kolobov. Plex: Making the most of the available data for robotic manipulation pretraining. In *CoRL*, 2023.

[69] H. Bharadhwaj, J. Vakil, M. Sharma, A. Gupta, S. Tulsiani, and V. Kumar. Roboagent: Generalization and efficiency in robot manipulation via semantic augmentations and action chunking. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[70] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.