X-DISTILL: CROSS-ARCHITECTURE VISION DISTILLATION ENABLES DATA-EFFICIENT VISUOMOTOR LEARNING

Anonymous authors
Paper under double-blind review

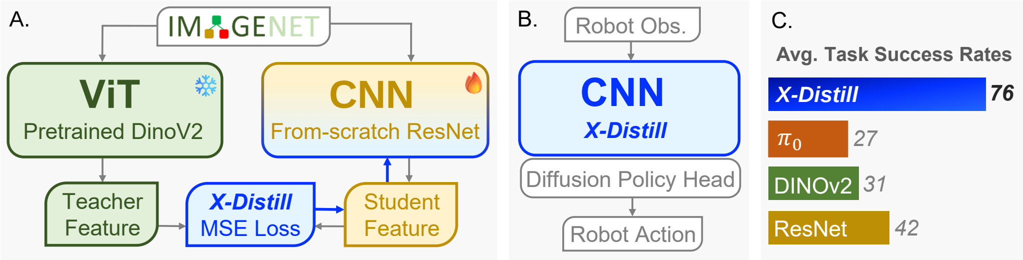

Figure 1: **X-Distill** is a simple yet effective visual encoder enabling data-efficient visuomotor learning. **A.** X-Distill is obtained by cross-architecture knowledge distillation from a large ViT teacher into a compact CNN student on general-purpose image datasets. **B.** Designed for visuomotor policy learning, X-Distill can be jointly fine-tuned end-to-end with a diffusion policy head on robotics-specific datasets. **C.** Given a few ($20 \sim 25$) demonstrations per task, X-Distill significantly outperforms representative counterparts on real-world manipulation tasks, exhibiting its surprising effectiveness.

ABSTRACT

Visuomotor policies often leverage large pre-trained Vision Transformers (ViTs) for their powerful generalization capabilities. However, their significant data requirements present a major challenge in the data-scarce context of most robotic learning settings, where compact CNNs with strong inductive biases can be more easily optimized. To address this trade-off, we introduce X-Distill, a simple yet highly effective method that synergizes the strengths of both architectures. Our approach involves an offline, cross-architecture knowledge distillation, transferring the rich visual representations of a large, frozen DINOv2 teacher to a compact ResNet-18 student on the general-purpose ImageNet dataset. This distilled encoder, now endowed with powerful visual priors, is then jointly fine-tuned with a diffusion policy head on the target manipulation tasks. Extensive experiments on 34 simulated benchmarks and 5 challenging real-world tasks demonstrate that our method consistently outperforms policies equipped with from-scratch ResNet or fine-tuned DINOv2 encoders. Notably, X-Distill also surpasses 3D encoders that utilize privileged point cloud observations or much larger Vision-Language Models. Our work highlights the efficacy of a simple, well-founded distillation strategy for achieving state-of-the-art performance in data-efficient robotic manipulation.

1 INTRODUCTION

Visuomotor policies, exemplified by Diffusion Policy Chi et al. (2023), are promising solutions for generalizable robotic manipulation. As end-to-end approaches, they typically rely on a visual encoder to extract manipulation-centric features from the raw pixels of a scene, followed by a policy head that generates concrete robot actions conditioning on the extracted visual features.

Benefiting from the success of large-scale vision pre-training Caron et al. (2021); Radford et al. (2021), it has become a common practice in recent advances Chi et al. (2024); Lin et al. (2024); Xue et al. (2025); Team et al. (2025) to initialize the visual encoder in a visuomotor policy with off-the-shelf, pre-trained Vision Transformers (ViTs) Dosovitskiy et al. (2021), e.g., CLIP Radford et al. (2021) or DINOv2 Oquab et al. (2023). These ViT-backend pre-trained models are found to exhibit enhanced generalization capabilities compared to Convolutional Neural Network (CNN) counterparts lacking open-world semantic knowledge, e.g., a ResNet He et al. (2016) trained from scratch.

However, lacking the strong intrinsic inductive biases inherent to CNNs, such as locality and translation equivariance, ViTs are known to struggle when faced with limited amounts of training data Dosovitskiy et al. (2021); Touvron et al. (2021). This issue becomes prominent and inevitable in the context of robot learning, where the dataset size is significantly smaller than in computer vision. Despite the recent trend among embodied AI startups to train policies with hundreds of *hours* of high-quality data produced by a data collection factory Black et al. (2024); Bu et al. (2025); Jiang et al. (2025), most researchers in academia typically collect data by hand, thus favoring data-efficient policies that perform well under a dataset size constraint of tens to a few hundred manipulation *trajectories*.

In this work, we find that simple advances to the visual encoder can yield higher-performing and more data-efficient visuomotor policies. More specifically, we design a cross-architecture vision distillation mechanism, or **X-Distill** in short, which attempts to combine the merits of both the open-world semantic generalization capabilities of pre-trained ViT models, and the inductive bias of CNN architectures that facilitate policy optimization under the low-data regime.

On an implementation level, we instantiate X-Distill by selecting DINOv2 (ViT-L/14) as the teacher encoder, a lightweight from-scratch ResNet-18 as the student encoder, and the mean squared error (MSE) between the teacher and student features as the knowledge distillation Hinton et al. (2015) loss. To make the X-Distilled encoder generally effective for diverse tasks, environments, and robot platforms, we choose the general-purpose ImageNet dataset as the distillation corpus, avoiding potential overfitting to any specific robotic scenarios. After X-Distillation, the CNN-backend encoder with ViT pre-training knowledge can be seamlessly integrated into the policy learning pipeline, jointly fine-tuned with the policy head in an end-to-end manner on any robotics-specific datasets.

We validate the effectiveness of X-Distill by conducting experiments on 34 simulated tasks across MetaWorld (Yu et al., 2020), Adroit (Kumar, 2016; Rajeswaran et al., 2017b), and DexArt (Bao et al., 2023) benchmarks, with 10 demonstrations per task. We also design 5 real-world tasks, carefully defining their In-Distribution (ID) and Out-of-Distribution (OOD) conditions and preparing $20 \sim 25$ demonstrations per task. Empirically, we find that Diffusion Policy with X-Distill consistently outperforms counterparts equipped with ResNet from scratch or DINOv2 as the encoder. Additionally, our policy also outperforms 3D Diffusion Policy Ze et al. (2024), which utilizes privileged 3D observation, as well as π_0 Black et al. (2024), a Vision-Language-Action (VLA) model that adopts a much larger VLM Beyer et al. (2024) as the visual perception encoder. Finally, we present a detailed qualitative analysis of the learned representations, providing insights into how X-Distill achieves superior performance over the baseline methods. ***Please refer to the project website for robot videos.***

2 Related Work

2.1 Visual Representation Learning

After the dominance of Convolutional Neural Networks (CNNs) He et al. (2016); Simonyan & Zisserman (2014); Tan & Le (2019) in the 2010s, Vision Transformers (ViTs) Dosovitskiy et al. (2021); Touvron et al. (2021); Liu et al. (2021) have gained increasing popularity in the 2020s because of their superior scaling capabilities and impressive representational power when pre-trained on large-scale datasets Caron et al. (2021); He et al. (2020; 2022); Oquab et al. (2023). Despite this trend, CNNs maintain a crucial edge in low-data regimes and continue to see widespread practical deployment. The key reason for this is the strong **inductive bias** inherent in CNNs. The convolutional operator imposes assumptions of locality and spatial weight sharing, which make them remarkably data-efficient. On the other hand, ViTs lack such biases and therefore require exposure to massive datasets to learn fundamental visual concepts. This discrepancy in data requirements keeps CNNs popular in many specialized domains, such as medical diagnostics Shamshad et al. (2023) or manufacturing quality control Liu et al. (2024), where large labeled datasets are often unavailable.

Recent works also explore adapting frozen models via task-specific adapters Sharma et al. (2023); Liu et al. (2023). Due to the lack of public codebases, we evaluate this paradigm using LoRA Hu et al. (2022). Our results (Sec. 4) show that in data-scarce regimes, our cross-architecture distillation significantly outperforms such parameter-efficient fine-tuning (PEFT) methods.

While parameter-efficient fine-tuning (PEFT) using adapters has been proposed Sharma et al. (2023); Liu et al. (2023), our experiments with LoRA Hu et al. (2022) reveal that in extremely data-scarce regimes, cross-architecture distillation yields superior sample efficiency compared to adapting large ViTs.

2.2 Cross-Architecture Knowledge Distillation

Knowledge distillation (KD) has become a cornerstone technique for model compression and knowledge transfer. Most work has focused on distillation between **homologous architectures**, including traditional CNN-to-CNN approaches Hinton et al. (2015) and more modern ViT-to-ViT frameworks designed for efficiency such as TinyViT Wu et al. (2022). For robotics, homologous knowledge distillation was recently explored by Theia (Shang et al., 2024), which fuses knowledge from multiple pre-trained ViTs into a single unified ViT encoder. In comparison, **cross-architecture distillation** is comparatively underexplored. Liu et al. (2022a). A representative work is DeiT Touvron et al. (2021), a CNN-to-ViT distillation where a CNN teacher can stabilize a data-hungry ViT student. By contrast, our work adopts the converse approach: a ViT-to-CNN distillation aiming to combine the inductive bias of a CNN with the powerful semantic understanding of a large-scale pre-trained ViT.

2.3 Visuomotor Policy Learning

Visuomotor policy learning is a promising paradigm for robotic manipulation. Representative works in this vein include Diffusion Policy Chi et al. (2023) and related approaches (Pearce et al., 2023; Reuss et al., 2023; Xian et al., 2023; Hu et al., 2024; Sridhar et al., 2024; Prasad et al., 2024), which typically consists of a visual encoder followed by a policy head network. Recently, Vision-Language-Action (VLA) models have been proposed to replace the visual encoder with more capable vision-language models (VLMs) Brohan et al. (2023); Reed et al. (2023); Black et al. (2024); Team et al. (2024); Liu et al. (2025), enabling impressive generalization abilities such as zero-shot skill deployment in unseen homes Intelligence et al. (2025). However, finetuning the VLM requires a substantial amount of training data. State-of-the-art VLAs such as π_0 Black et al. (2024), AgiBot GO-1 Bu et al. (2025), and Galaxea G0 Jiang et al. (2025) all rely on their embodiment-specific large-scale datasets, measured either in millions by the number of trajectories or in hundreds of hours by the physical on-robot execution time. In this work, we focus on training capable visuomotor policies when a limited amount of training data is available, i.e., using only ~ 25 demonstration trajectories per task.

3 Method

3.1 X-Distill: A Cross-Architecture Distillation Method

As detailed in Algorithm 1a in Appendix A, we employ cross-architecture knowledge distillation to transfer the representational capabilities of a large Vision Transformer (ViT) into a compact CNN with beneficial inductive biases. Crucially, this entire process is conducted exclusively on the general-purpose ImageNet-1K Deng et al. (2009) dataset ($\mathcal{X}$), which contains approximately 1.3 million images depicting a wide variety of real-world objects and scenes. This decoupling of visual feature distillation from the downstream domain-specific datasets makes X-Distill entirely **domain-agnostic**. In other words, the resulting X-Distill encoder is universally suitable for all kinds of robotic manipulation tasks, thus avoiding potential overfitting to any specific environments, camera setups, or robotic embodiments.

Selection of teacher and student networks. We select the pre-trained DINOv2 (ViT-L/14) model as our teacher $\mathcal{T}$. With approximately 304M parameters, this large-scale model is used off-the-shelf as a frozen feature extractor, serving as a robust source of semantic and structural visual knowledge. For the student model $\mathcal{S}$, we choose a highly compact ResNet-18 architecture with only

11M parameters. The choice of student network prioritizes not only its computational efficiency with a network parameter size nearly 28× smaller than the teacher, but also its strong inductive biases such as spatial locality that are beneficial for manipulation tasks.

Domain-agnostic distillation. The student is trained to replicate the feature outputs of the teacher on ImageNet-1K. For a given input image x, we extract the global `[CLS]` token from the DINOv2 teacher, which serves as the target feature vector. The ResNet-18 student architecture is modified with a final linear layer to match the feature dimension of the teacher. The core objective is then to minimize the direct Mean Squared Error (MSE) between these two feature vectors:

$$\mathcal{L}_{\text{KD}} = \mathbb{E}_{x \sim \mathcal{X}} \left[\| f_{\mathcal{T}}(x) - f_{\mathcal{S}}(x) \|_2^2 \right] \tag{1}$$

where $f_{\mathcal{T}}$ and $f_{\mathcal{S}}$ represent the complete feature extraction processes of the teacher and the dimension-aligned student, respectively. This process results in a ResNet-18 with parameter weights $\mathcal{S}^*$, which encodes the open-world generalization knowledge of the teacher network.

3.2 Finetuning X-Distill for Visuomotor Policy Learning

Given the powerful initialization provided by X-Distill, we deploy the encoder $\mathcal{S}^*$ for downstream policy learning on a target robotics dataset, as outlined in Algorithm 1b. We use a Diffusion Policy Chi et al. (2023) head, which generates action chunks conditioned on robot observations.

At each timestep, the distilled encoder $\mathcal{S}^*$ processes a history of camera images $x_{t-T_o+1:t}$ into a visual feature vector z_{img}. This vector is concatenated with the robot's proprioceptive state s_t to form a comprehensive conditioning vector, $c = \text{concat}(z_{\text{img}}, s_t)$. This conditioning vector c guides the entire action generation process. During inference, actions are generated by iteratively denoising a random Gaussian tensor, conditioned on this vector c.

Crucially, both the distilled encoder $\mathcal{S}^*$ and the diffusion policy head π_θ are jointly trained on robotics-specific datasets. This end-to-end optimization allows the powerful, general-purpose features from the distillation phase to be fine-tuned and specialized for the specific demands of the manipulation task. The entire system is optimized by minimizing the diffusion loss objective:

$$\mathcal{L}_{\text{diff}} = \mathbb{E}_{\mathbf{A}^0, \epsilon, k} \left[\| \epsilon - \epsilon_\theta(\mathbf{A}^0 + \sigma_k \epsilon | c, k) \|^2 \right], \tag{2}$$

where $\mathbf{A}^0$ denotes the ground-truth actions, $\epsilon \sim \mathcal{N}(0, \mathbf{I})$, and k is sampled from the diffusion steps.

4 Simulation Experiments

4.1 Setup

Simulation benchmarks. To thoroughly evaluate the effectiveness of our method, we conduct experiments across a total of 34 tasks from 3 distinct MuJoCo-based robotic manipulation benchmarks. Our evaluation encompasses tasks requiring parallel gripper manipulation from MetaWorld (Yu et al., 2020), dexterous motor skills from Adroit (Kumar, 2016; Rajeswaran et al., 2017b), and articulated object manipulation from DexArt (Bao et al., 2023). Tasks in MetaWorld are categorized into various difficulty levels—*easy*, *medium*, *hard*, and *very hard*—based on Seo et al. (2023). A brief overview of the tasks is provided in Appendix B.

Expert demonstrations. 10 trajectories are collected for each simulation task. For MetaWorld, scripted policies are employed. Trajectories in the remaining domains are gathered using agents trained via reinforcement learning (RL): specifically, VRL3 (Wang et al., 2022) is applied for Adroit, while PPO (Schulman et al., 2017) is utilized for the remaining benchmarks. Other training-related hyperparameters can be found in the Appendix C.

Evaluation Metric. We report all results averaged over 3 random seeds (0, 1, and 2). For each individual training run, we evaluate the policy on 20 episodes every 200 epochs, and the highest success rate achieved throughout the run is reported for that seed. The final values presented in our tables are the mean of these scores across the 3 seeds.

Table 1: **Averaged success rates on MetaWorld, Adroit and Dexart benchmarks.** PointNet-DP3 is marked in gray because it processes privileged background-cropped 3D point clouds.

Method	MetaWorld				Adroit (3)	Dexart (2)	Average
	(easy 20)	(medium 7)	(hard 1)	(very hard 1)			
ResNet-scratch	76.6	48.0	38.0	50.0	37.7	54.5	64.1
DINOv2	78.5	46.0	**48.0**	38.0	51.7	58.0	66.2
Depth-Anything	68.2	29.3	42.0	43.0	40.3	**66.0**	56.1
Theia	50.9	13.7	0.0	38.3	8.7	24.0	36.0
X-Distill (Ours)	**93.9**	**88.3**	**48.0**	**88.0**	**68.3**	63.5	**87.2**
PointNet-DP3	90.4	70.6	14.0	72.0	40.7	85.0	84.0

4.2 Performance

Compared methods. We compare X-Distilled ResNet-18 (11M) against several visual encoder counterparts with a similar number of parameters, including:

- **ResNet-scratch** (He et al., 2016), ResNet-18 (11M) trained from scratch;
- **DINOv2** (Oquab et al., 2023), ViT-small (21M) pre-trained using large-scale self-supervision;
- **Depth-Anything** (Yang et al., 2024), ViT-small (24.8M) trained for monocular depth estimation;
- **Theia** (Shang et al., 2024), ViT-small (22M) that distills multiple vision foundation models.

Additionally, we also benchmark against **PointNet-DP3** (Ze et al., 2024), a PointNet-based architecture (0.06M) processing privileged background-cropped 3D point cloud observations.

Main results. As summarized in Table 1, X-Distill achieves the best overall average performance across all 34 tasks. It consistently outperforms all 2D vision baselines by a significant margin, securing state-of-the-art success rates in most simulation benchmarks. This validates the effectiveness of our distillation strategy for data-scarce visuomotor learning.

Notably, our 2D approach remains highly competitive even in geometrically demanding settings where methods leveraging privileged 3D inputs have a natural advantage. For instance, the DexArt-Toilet task requires the robot to lift the toilet lid from a frontal viewpoint, which is inherently challenging to estimate the depth relationship between the gripper and the object to be manipulated from a single RGB image. Nevertheless, X-Distill still demonstrates decent performance in many of these challenging tasks, showcasing a strong prior in spatial reasoning. More detailed settings and results are available in Appendix B and Figure 6.

Table 2: **Ablation study on MetaWorld benchmarks.** We evaluate the impact of teacher model scale (DINOv2-L vs. S), student architectural bias (CNN vs. ViT), and student model scale.

Teacher	Student	MW-20 (easy)	MW-7 (medium)	MW-1 (hard)	MW-1 (v. hard)	Average
DINOv2-L	ResNet-18 (11M)	93.9	**88.3**	48.0	88.0	**90.7**
	ViT-S-Half (11M)	72.0	25.3	2.0	40.0	57.2
	ConvNeXt (89M)	91.8	77.4	**50.0**	83.0	86.6
DINOv2-S	ResNet-18 (11M)	**94.3**	87.3	43.0	**90.0**	90.6

4.3 Ablation Studies

We conduct ablation studies to investigate the impact of the teacher network parameter size, as well as the student network architectural bias and parameter size within our X-Distill framework. The ablation results are summarized in Table 2.

Teacher network parameter size. We distill **DINOv2-S** (21M) and **DINOv2-L** (304M) teachers into the same ResNet-18 student. No significant difference can be observed between DINOv2-S and DINOv2-L, indicating our X-Distill framework is insensitive to the specific network configurations of a well-pre-trained teacher network. Nevertheless, we use the DINOv2-L teacher for all subsequent experiments to ensure the maximized knowledge quality that the teacher could provide.

Student network architectural bias. We distill the same DINOv2-L teacher into a **ResNet-18** (11M) and a customized **ViT-S-Half** (11M) of the same size. The ResNet-18 student substantially

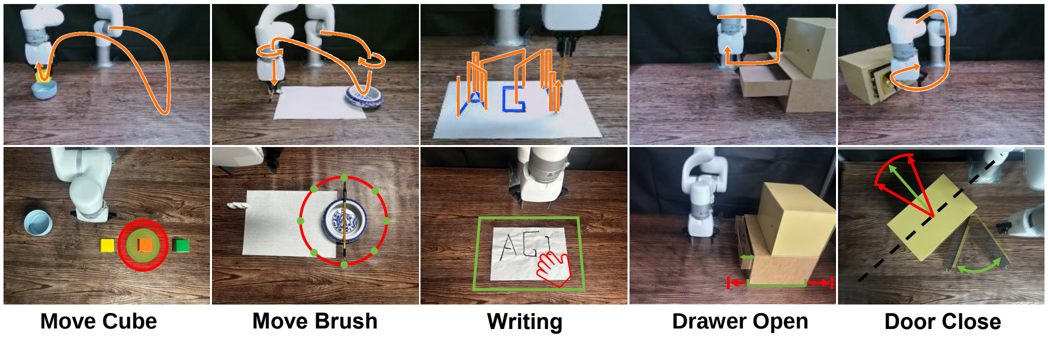

Figure 2: **Visualization of configurations for our real-world tasks.** The orange arrow provides a schematic representation of the gripper trajectory as derived from the data. The green regions represent the distribution of object/robot configurations seen during training demonstrations, while the red regions illustrate the novel configurations used for generalization testing.

outperforms its ViT counterpart by 33.5%. This highlights the crucial role of convolutional inductive biases for visuomotor learning in a low-data regime, supporting our primary hypothesis.

Student network parameter size. We compare our compact **ResNet-18** (11M) student to a much larger **ConvNeXt** (89M) Liu et al. (2022b) CNN counterpart. Despite its greater capacity, the larger model achieves a slightly degraded success rate by 4.1% on robotics tasks. This confirms our intuition that smaller visual encoders with stronger inductive biases are easier to optimize, thus beneficial for data-efficient policy learning.

5 Real-World Experiments

5.1 Experiment Setup

We conduct all real-world experiments with an X-Arm 6 robotic arm, capture image observations through a web camera at 15Hz, and prepare a small collection of demonstrations ($20 \sim 25$) per task via Meta-Quest VR teleoperation (in Appendix E). We design 5 tabletop manipulation tasks, and carefully define their In-Distribution (ID) and Out-of-Distribution (OOD) object randomization ranges for rigorous and repeatable evaluation. Task execution trajectories as well as ID and OOD ranges are illustrated in Figure 2. Detailed numbers of demonstrations and evaluation trials can be found in Table 3.

More specifically, **Move Cube** requires the robot to pick up an orange cube and place it into a bowl. In addition to testing on OOD cube positions, we also conduct a color generalization (C-Gen) test with unseen yellow and green cubes. **Move Brush** requires the robot to pick up a brush pen with various initial orientational and translational offsets and place it onto a stand. **Writing "AGI"** requires the robot to sequentially write letters "AGI" on a randomly placed piece of paper. We conduct OOD dynamic perturbation trials, where human perturbators randomly drag the paper elsewhere while the robot is writing letters. **Drawer Open** requires the robot to insert its finger into varying initial gaps of the randomly placed drawer, and then open it by sliding outward. **Door Close** requires the robot to close the door from various initial open angles by pushing it inward. More task descriptions are provided in the Appendix F.

5.2 Experiment Results and Analysis

Baselines. We compare our X-Distill encoder against three representative counterparts. The first two are Diffusion Policies equipped with either a ResNet encoder trained from scratch or an off-the-shelf DINOv2 encoder. Both of the two baseline policies, as well as our approach, are trained for 1500 epochs on our task-specific data. Our third baseline is the state-of-the-art Vision-Language-Action (VLA) model, π_0 Black et al. (2024). Considering its significant computational requirements, we performed supervised fine-tuning (SFT) for $30,000$ steps, following the official recommendations,

Table 3: **A comparison of task execution success rates (%) for real-world tasks**, along with the numbers of demonstrations and evaluation trials.

Method	Move Cube			Move Brush		Writing "AGI"		Drawer Open		Door Close		**Average**
	ID	OOD	C-Gen	ID	OOD	ID	OOD	ID	OOD	ID	OOD	
# Demos	20	0	0	24	0	25	0	20	0	20	0	–
# Eval Trials	15	5	10	4	8	5	4	5	15	5	10	–
ResNet-scratch	66.7	0.0	50.0	0.0	0.0	40.0	25.0	**100.0**	13.3	60.0	80.0	41.9
DINOv2	26.7	20.0	20.0	0.0	0.0	0.0	0.0	80.0	13.3	**100.0**	90.0	31.4
π_0 (SFT)	0.0	0.0	0.0	25.0	0.0	0.0	0.0	80.0	33.3	80.0	90.0	26.7
X-Distill (Ours)	**93.3**	**40.0**	**70.0**	**75.0**	**25.0**	**100.0**	**100.0**	**100.0**	**53.3**	**100.0**	**100.0**	**75.6**

which took approximately 20 hours on a single A100 GPU. All methods were trained using the same dataset consisting of the same limited number of demonstrations to ensure a fair comparison.

Main results. The quantitative results for real-world experiments are summarized in Table 3. X-Distill demonstrates clear superiority, consistently outperforming all baseline approaches by a large margin and achieving the highest success rates across both ID and OOD evaluation settings. Simply finetuning a large ViT encoder like DINOv2 yields poor performance, confirming the challenge of effectively optimizing large Transformer networks in data-scarce scenarios and underscoring the effectiveness of our cross-architecture distillation.

The performance gap is particularly insightful when comparing against the VLA model, π_0. While π_0 shows reasonable success on simpler tasks like **Drawer Open**, it struggles significantly on more complex, high-precision tasks such as **Writing "AGI"**, where its performance drops to zero. This suggests that directly finetuning a large, generalist VLA on small, task-specific datasets is a significant challenge. In contrast, our X-Distill framework effectively bridges this gap by transferring knowledge into a compact, data-efficient architecture, highlighting the importance of matching the model and pre-training strategy to the available data resources.

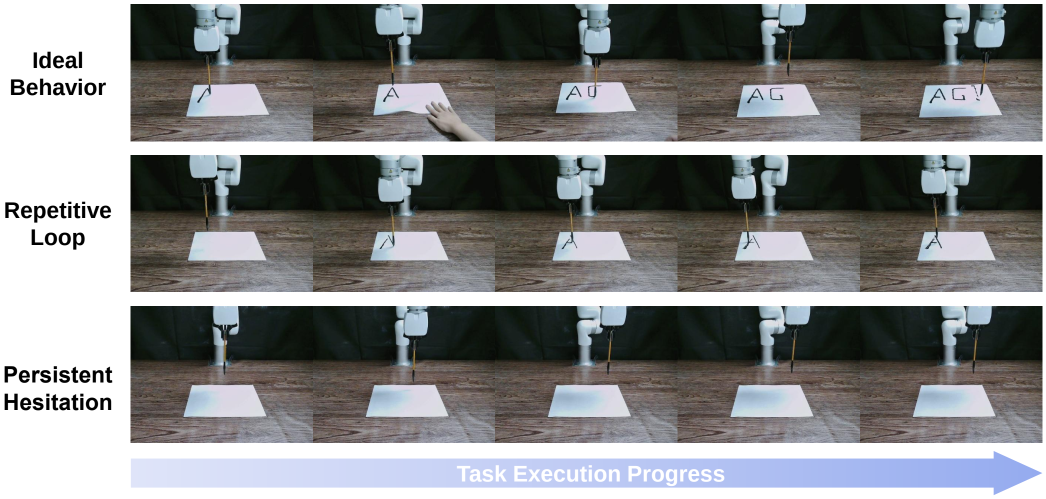

Figure 3: **Representative trajectory types observed in the "Writing AGI" task.** We identify three distinct behaviors: **(1) Ideal Behavior:** Successful and robust execution of all three letters, even under perturbation. **(2) Repetitive Loop:** Perseverative behavior where the policy gets stuck repeatedly writing the first letter 'A'. **(3) Persistent Hesitation:** Dithering motion above the paper without initiating the writing task.

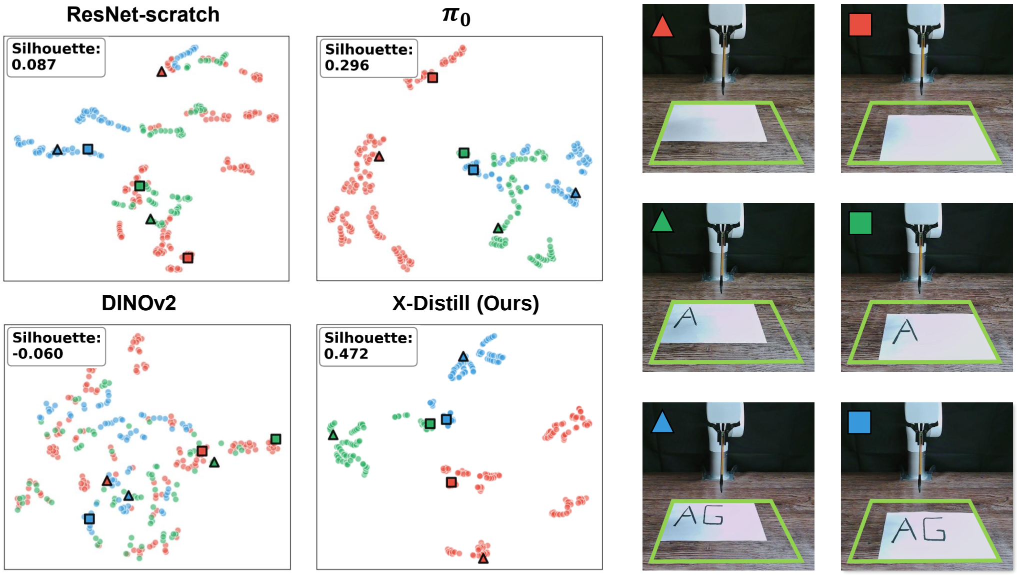

Figure 4: **t-SNE visualization of learned feature spaces on the "Writing AGI" task.** Our X-Distill encoder learns to form three distinct clusters corresponding to the task's semantic stages, quantitatively confirming a well-separated feature space with a high Silhouette Score Rousseeuw (1987) of 0.472, which indicates a high degree of cluster cohesion and separation compared with the baselines. This semantic separability is crucial for the policy to accurately identify the current task stage, enabling precise long-horizon planning for the sequential writing task.

5.3 Qualitative Analysis

We focus on our most challenging long-horizon task, **Writing "AGI"**, where success critically depends on the encoder's ability to discern subtle but crucial visual state changes. For instance, before starting to write 'G', the robot's physical state is nearly identical to when it starts writing 'A'; the only distinguishing information is the visual context of the letter 'A' already present on the paper.

Empirically, we observe that Diffusion Policy with ResNet-scratch often fails by repeatedly executing the trajectory for 'A', indicating an inability to visually differentiate these critical semantic states. Meanwhile, Diffusion Policy with DINOv2 and π_0 (SFT) often get stuck and start trembling before writing any letter, which is a potential sign of underfitting. In comparison, Diffusion Policy with X-Distill is the only one among the compared methods that manages to differentiate all critical stages and completes writing all the three letters sequentially one by one. Even under severe external disturbances that drag the paper away during the writing process, the X-Distill-empowered policy is still robust, responsively following the movement of the paper and rapidly adapting to the correct position for writing the next letter. Example trajectories are shown in Figure 3. To provide deeper insights into how X-Distill achieves superior quantitative performance compared to its counterparts, we conduct further t-SNE analysis and saliency map visualization of the learned visual representations.

t-SNE visualization of feature space separability. The global structure of the learned feature space is visualized via t-SNE Maaten & Hinton (2008) in Figure 4. Each data point corresponds to the feature of a frame sampled from three crucial stages for policy decision making marked in three respective colors: (1) before writing 'A', (2) before writing 'G', and (3) before writing 'I'. The features produced from an ideal visual encoder should form three distinct clusters, corresponding to the three colors. It can be observed that X-Distill gives a more separable feature space than the Paligemma Beyer et al. (2024) encoder extracted from π_0, while the features from both ResNet-scratch and DINOv2 are nearly indistinguishable. These results indicate that X-Distill learns a feature space that is semantically coherent and robust to visual distractors.

Inspecting task-relevant feature attribution via saliency maps. To further investigate how our X-Distill achieves emergent semantic feature separation, we inspect pixel-level feature attribution by visualizing the saliency maps. For saliency visualization, we adopt Grad-CAM for CNN-based

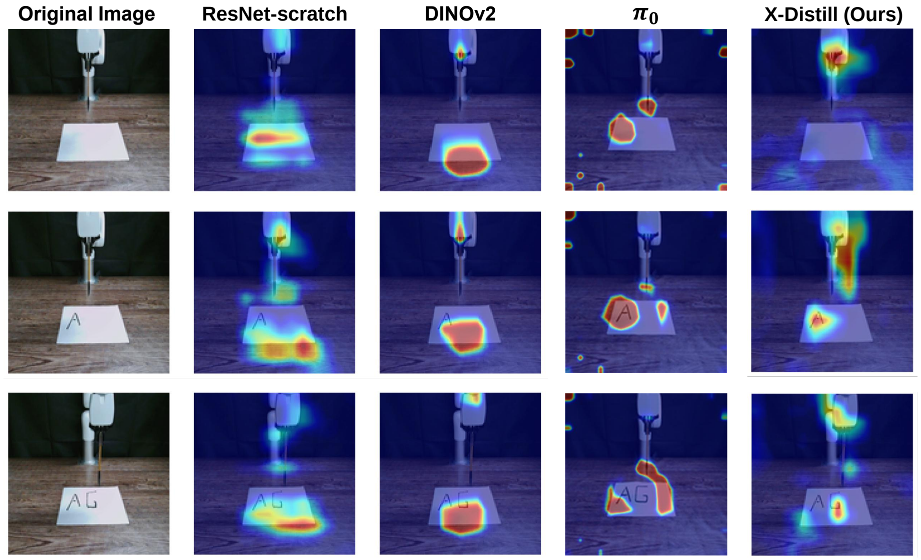

Figure 5: **Saliency map comparison on the "Writing AGI" task.** We visualize the model's visual focus at the beginning of each writing stage. Our X-Distill encoder correctly shifts its attention from the gripper (before 'A'), to the letter 'A' (before 'G'), and finally to the letter 'G' (before 'I'). Baseline models exhibit diffuse or irrelevant attention.

models Selvaraju et al. (2019), and the cross-attention strengths between the `[CLS]` token and all local patch features for ViT-based models Dosovitskiy et al. (2020); Chefer et al. (2021), providing a cross-architecture comparison of the visual focus. As shown in Figure 5, both DINOv2 and π_0 are unable to effectively shift the high-attention regions throughout the task progress, which cross-verifies our earlier judgment of underfitting. Meanwhile, the saliency maps of ResNet-scratch and X-Distill exhibits more reasonable shifting patterns, but the latter is significantly more precise. More specifically, before writing 'A' on the blank page, the full attention of X-Distill is focused on the **robot gripper**, the primary actor. Then, before writing 'G', its focus dynamically shifts to the **letter 'A'** already on the paper. Finally, before writing 'I', X-Distill shifts attention again, attending to the **letter 'G'**, whose appearance serves as the cue to write the final letter 'I'.

The t-SNE and saliency map visualizations combined reveal that X-Distill successfully learns a semantically meaningful and robust visual representation. Such representation can well differentiate critical states and dynamically focus on task-relevant visual cues, which ultimately contributes to the policy's success in complex long-horizon manipulation tasks.

6 Conclusion

We introduced **X-Distill**, a framework addressing the trade-off between ViT generalization and CNN sample efficiency. By distilling DINOv2 features into a ResNet-18 on ImageNet, we obtain a robust encoder for data-scarce robotics. Extensive experiments on 34 simulated and 5 real-world tasks show X-Distill outperforms standard baselines and even privileged 3D or VLA policies. Our analysis attributes this to the learned semantically separable feature space. Ultimately, X-Distill demonstrates that a simple, well-founded distillation strategy is a key enabler for data-efficient visuomotor learning.

Limitations and Future Work. While effective, our direct feature distillation leaves room for exploration. Future directions include adopting sophisticated techniques to align intermediate features Liu et al. (2022a) and distilling from multimodal VLA teachers to incorporate language priors. Additionally, while we focus on the data-scarce regime, investigating X-Distill's scalability in data-rich scenarios and its application to dynamic tasks like mobile manipulation remain important open questions.

LLM usage disclosure. We used the Gemini 2.5 pro model to refine grammar and phrasing.

References

Chen Bao, Helin Xu, Yuzhe Qin, and Xiaolong Wang. Dexart: Benchmarking generalizable dexterous manipulation with articulated objects. In *CVPR*, 2023.

Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, et al. Paligemma: A versatile 3b vlm for transfer. *arXiv preprint arXiv:2407.07726*, 2024.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, Pete Florence, Chuyuan Fu, Montse Gonzalez Arenas, Keerthana Gopalakrishnan, Kehang Han, Karol Hausman, Alexander Herzog, Jasmine Hsu, Brian Ichter, Alex Irpan, Nikhil Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Lisa Lee, Tsang-Wei Edward Lee, Sergey Levine, Yao Lu, Henryk Michalewski, Igor Mordatch, Karl Pertsch, Kanishka Rao, Krista Reymann, Michael Ryoo, Grecia Salazar, Pannag Sanketi, Pierre Sermanet, Jaspiar Singh, Anikait Singh, Radu Soricut, Huong Tran, Vincent Vanhoucke, Quan Vuong, Ayzaan Wahid, Stefan Welker, Paul Wohlhart, Jialin Wu, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. Rt-2: Vision-language-action models transfer web knowledge to robotic control, 2023. URL https://arxiv.org/abs/2307.15818.

Qingwen Bu, Jisong Cai, Li Chen, Xiuqi Cui, Yan Ding, Siyuan Feng, Shenyuan Gao, Xindong He, Xuan Hu, Xu Huang, et al. Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. *arXiv preprint arXiv:2503.06669*, 2025.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9650–9660, 2021.

Hila Chefer, Shir Gur, and Lior Wolf. Transformer interpretability beyond attention visualization, 2021. URL https://arxiv.org/abs/2012.09838.

Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *RSS*, 2023.

Cheng Chi, Zhenjia Xu, Chuer Pan, Eric Cousineau, Benjamin Burchfiel, Siyuan Feng, Russ Tedrake, and Shuran Song. Universal manipulation interface: In-the-wild robot teaching without in-the-wild robots. In *Proceedings of Robotics: Science and Systems (RSS)*, 2024.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021.

Alexey Dosovitskiy et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CVPR*, 2016.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. *CVPR*, 2020.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. *CVPR*, 2022.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *NeurIPS*, 2015.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

Kaizhe Hu, Zihang Rui, Yao He, Yuyao Liu, Pu Hua, and Huazhe Xu. Stem-ob: Generalizable visual imitation learning with stem-like convergent observation through diffusion inversion, 2024. URL https://arxiv.org/abs/2411.04919.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: a vision-language-action model with open-world generalization, 2025. URL https://arxiv.org/abs/2504.16054.

Tao Jiang, Tianyuan Yuan, Yicheng Liu, Chenhao Lu, Jianning Cui, Xiao Liu, Shuiqi Cheng, Jiyang Gao, Huazhe Xu, and Hang Zhao. Galaxea open-world dataset and g0 dual-system vla model. *arXiv preprint arXiv:2509.00576*, 2025.

Vikash Kumar. *Manipulators and manipulation in high dimensional spaces*. PhD thesis, University of Washington, Seattle, 2016.

Fanqi Lin, Yingdong Hu, Pingyue Sheng, Chuan Wen, Jiacheng You, and Yang Gao. Data scaling laws in imitation learning for robotic manipulation. *arXiv preprint arXiv:2410.18647*, 2024.

Jiaqi Liu, Guoyang Xie, Jinbao Wang, Shangnian Li, Chengjie Wang, Feng Zheng, and Yaochu Jin. Deep industrial image anomaly detection: A survey. *Machine Intelligence Research*, 21(1): 104–135, 2024.

Songming Liu, Lingxuan Wu, Bangguo Li, Hengkai Tan, Huayu Chen, Zhengyi Wang, Ke Xu, Hang Su, and Jun Zhu. Rdt-1b: a diffusion foundation model for bimanual manipulation, 2025. URL https://arxiv.org/abs/2410.07864.

Yufan Liu, Jiajiong Cao, Bing Li, Weiming Hu, Jingting Ding, and Liang Li. Cross-architecture knowledge distillation, 2022a. URL https://arxiv.org/abs/2207.05273.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *ICCV*, 2021.

Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s, 2022b. URL https://arxiv.org/abs/2201.03545.

Zuxin Liu, Jesse Zhang, Kavosh Asadi, Yao Liu, Ding Zhao, Shoham Sabach, and Rasool Fakoor. Tail: Task-specific adapters for imitation learning with large pretrained models. *arXiv preprint arXiv:2310.05905*, 2023.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

Suraj Nair, Aravind Rajeswaran, Vikash Kumar, Chelsea Finn, and Abhinav Gupta. R3m: A universal visual representation for robot manipulation. *arXiv preprint arXiv:2203.12601*, 2022.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023. URL https://arxiv.org/abs/2304.07193.

Tim Pearce, Tabish Rashid, Anssi Kanervisto, Dave Bignell, Mingfei Sun, Raluca Georgescu, Sergio Valcarcel Macua, Shan Zheng Tan, Ida Momennejad, Katja Hofmann, et al. Imitating human behaviour with diffusion models. *ICLR*, 2023.

Aaditya Prasad, Kevin Lin, Jimmy Wu, Linqi Zhou, and Jeannette Bohg. Consistency policy: Accelerated visuomotor policies via consistency distillation. In *Robotics: Science and Systems*, 2024.

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. *International Conference on Machine Learning (ICML)*, pp. 8748–8763, 2021.

Aravind Rajeswaran, Vikash Kumar, Abhishek Gupta, Giulia Vezzani, John Schulman, Emanuel Todorov, and Sergey Levine. Learning complex dexterous manipulation with deep reinforcement learning and demonstrations. *arXiv*, 2017a.

Aravind Rajeswaran, Vikash Kumar, Abhishek Gupta, Giulia Vezzani, John Schulman, Emanuel Todorov, and Sergey Levine. Learning complex dexterous manipulation with deep reinforcement learning and demonstrations. *arXiv preprint arXiv:1709.10087*, 2017b.

Scott Reed, Konrad Zolna, Emilio Parisotto, Serkan Colmenarejo, Alexander Novikov, Gabriel Barth-Maron, Mai Gimenez, Yvette Sulsky, Jackie Kay, Jost Springenberg, et al. Openvla: An open-source vision-language-action model. *arXiv:2306.00091*, 2023.

Moritz Reuss, Maximilian Li, Xiaogang Jia, and Rudolf Lioutikov. Goal-conditioned imitation learning using score-based diffusion policies. *arXiv preprint arXiv:2304.02532*, 2023.

Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128(2):336–359, October 2019. ISSN 1573-1405. doi: 10.1007/s11263-019-01228-7. URL http://dx.doi.org/10.1007/s11263-019-01228-7.

Younggyo Seo, Danijar Hafner, Hao Liu, Fangchen Liu, Stephen James, Kimin Lee, and Pieter Abbeel. Masked world models for visual control. In *CoRL*, 2023.

Fahad Shamshad, Salman Khan, Syed Waqas Zamir, Muhammad Haris Khan, Munawar Hayat, Fahad Shahbaz Khan, and Huazhu Fu. Transformers in medical imaging: A survey. *Medical image analysis*, 88:102802, 2023.

Jinghuan Shang, Karl Schmeckpeper, Brandon B May, Maria Vittoria Minniti, Tarik Kelestemur, David Watkins, and Laura Herlant. Theia: Distilling diverse vision foundation models for robot learning. *arXiv preprint arXiv:2407.20179*, 2024.

Mohit Sharma, Claudio Fantacci, Yuxiang Zhou, Skanda Koppula, Nicolas Heess, Jon Scholz, and Yusuf Aytar. Lossless adaptation of pretrained vision models for robotic manipulation. *arXiv preprint arXiv:2304.06600*, 2023.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *ICLR*, 2014.

Kaustubh Sridhar, Souradeep Dutta, Dinesh Jayaraman, James Weimer, and Insup Lee. Memory-consistent neural networks for imitation learning. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=R3Tf7LDdX4.

Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *ICML*, 2019.

Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, Jianlan Luo, You Liang Tan, Lawrence Yunliang Chen, Pannag Sanketi, Quan Vuong, Ted Xiao, Dorsa Sadigh, Chelsea Finn, and Sergey Levine. Octo: An open-source generalist robot policy, 2024. URL https://arxiv.org/abs/2405.12213.

TRI LBM Team, Jose Barreiros, Andrew Beaulieu, Aditya Bhat, Rick Cory, Eric Cousineau, Hongkai Dai, Ching-Hsin Fang, Kunimatsu Hashimoto, Muhammad Zubair Irshad, Masha Itkina, Naveen Kuppuswamy, Kuan-Hui Lee, Katherine Liu, Dale McConachie, Ian McMahon, Haruki Nishimura, Calder Phillips-Grafflin, Charles Richter, Paarth Shah, Krishnan Srinivasan, Blake Wulfe, Chen Xu, Mengchao Zhang, Alex Alspach, Maya Angeles, Kushal Arora, Vitor Campagnolo Guizilini, Alejandro Castro, Dian Chen, Ting-Sheng Chu, Sam Creasey, Sean Curtis, Richard Denitto, Emma Dixon, Eric Dusel, Matthew Ferreira, Aimee Goncalves, Grant Gould, Damrong Guoy, Swati Gupta, Xuchen Han, Kyle Hatch, Brendan Hathaway, Allison Henry, Hillel Hochsztein, Phoebe Horgan, Shun Iwase, Donovon Jackson, Siddharth Karamcheti, Sedrick Keh, Joseph Masterjohn, Jean Mercat, Patrick Miller, Paul Mitiguy, Tony Nguyen, Jeremy Nimmer, Yuki Noguchi, Reko Ong, Aykut Onol, Owen Pfannenstiehl, Richard Poyner, Leticia Priebe Mendes Rocha, Gordon Richardson, Christopher Rodriguez, Derick Seale, Michael Sherman, Mariah Smith-Jones, David Tago, Pavel Tokmakov, Matthew Tran, Basile Van Hoorick, Igor Vasiljevic, Sergey Zakharov, Mark Zolotas, Rares Ambrus, Kerri Fetzer-Borelli, Benjamin Burchfiel, Hadas Kress-Gazit, Siyuan Feng, Stacie Ford, and Russ Tedrake. A careful examination of large behavior models for multitask dexterous manipulation. 2025. URL https://arxiv.org/abs/2507.05331.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. *ICML*, 2021.

Che Wang, Xufang Luo, Keith Ross, and Dongsheng Li. Vrl3: A data-driven framework for visual deep reinforcement learning. *Advances in Neural Information Processing Systems*, 2022.

Kan Wu, Jinnian Zhang, Houwen Peng, Mengchen Liu, Bin Xiao, Jianlong Fu, and Lu Yuan. Tinyvit: Fast pretraining distillation for small vision transformers, 2022. URL https://arxiv.org/abs/2207.10666.

Zhou Xian, Nikolaos Gkanatsios, Theophile Gervet, Tsung-Wei Ke, and Katerina Fragkiadaki. Chaineddiffuser: Unifying trajectory diffusion and keypose prediction for robotic manipulation. In *CoRL*, 2023.

Zhengrong Xue, Shuying Deng, Zhenyang Chen, Yixuan Wang, Zhecheng Yuan, and Huazhe Xu. Demogen: Synthetic demonstration generation for data-efficient visuomotor policy learning. *arXiv preprint arXiv:2502.16932*, 2025.

Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10371–10381, 2024.

Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *CoRL*, 2020.

Yanjie Ze, Gu Zhang, Kangning Zhang, Chenyuan Hu, Muhan Wang, and Huazhe Xu. 3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations. *arXiv preprint arXiv:2403.03954*, 2024.

A Algorithm pseudo-code

Algorithm 1 outlines our two-stage framework. Sub-algorithm (a) describes the knowledge distillation process where a student ResNet is trained to mimic the representations of a frozen DINOv2 teacher on ImageNet, yielding our X-Distill encoder. Sub-algorithm (b) shows how this pre-trained encoder is fine-tuned with a diffusion policy head on target robotic datasets for effective policy learning.

Algorithm 1 How to acquire and leverage X-Distill.

(a) Acquiring X-Distill via Knowledge Distillation

```
Require: Teacher encoder $\mathcal{T}$ (frozen DINOv2), Student encoder $\mathcal{S}$ (from-scratch ResNet), Domain-agnostic dataset $\mathcal{D}_{\text{large}}$ (ImageNet).
for each training epoch do
    for each batch $x \in \mathcal{D}_{\text{large}}$ do
        $z_T \leftarrow f_{\mathcal{T}}(x)$.
        $z_S \leftarrow f_{\mathcal{S}}(x)$.
        $L \leftarrow \mathcal{L}_{\text{KD}}(z_S, \text{sg}(z_T))$        ▷ Eq. (1)
        Update student encoder $\mathcal{S}$ via $\nabla_{\mathcal{S}} L$.
    end for
end for
Save the weights of S as S*
return X-Distilled encoder weights $\mathcal{S}^*$.
```

(b) Leveraging X-Distill via Policy Finetuning

```
Require: X-Distilled encoder weights $\mathcal{S}^*$, Diffusion policy head $\pi_\theta$, Domain-specific dataset $\mathcal{D}_{\text{robotics}}$.
Initialize encoder $\mathcal{S}$ with weights from $\mathcal{S}^*$.
for each training epoch do
    for each batch $(o, a) \in \mathcal{D}_{\text{robotics}}$ do
        $z_{img} \leftarrow f_{\mathcal{S}}(x)$.
        $c \leftarrow \text{concat}(z_{img}, s)$.
        Compute $L_{\text{diff}}$.        ▷ Eq. (2)
        Update $\mathcal{S}$ and $\pi_\theta$ via $\nabla_{\mathcal{S},\theta} L_{\text{diff}}$.
    end for
end for
return Trained encoder and policy $(\mathcal{S}^{**}, \pi_\theta^*)$.
```

B Simulation Environments

Simulated tasks. We collect a diverse set of simulated tasks to systematically evaluate imitation learning algorithms, with a particular focus on robotic manipulation in 2D visual settings. Our benchmark draws on three key environments: MetaWorld (Yu et al., 2020), Adroit (Rajeswaran et al., 2017a), and DexArt (Bao et al., 2023), each offering distinct challenges in vision-based motor control.

MetaWorld provides a suite of robotic manipulation tasks designed for multi-task and meta-reinforcement learning, featuring a variety of object interactions in simulated tabletop scenarios, all observable via RGB images. Adroit focuses on dexterous hand manipulation using a high-DoF simulated human hand, with tasks such as object relocation and in-hand rotation, posing significant challenges in policy learning from pixel inputs. DexArt introduces tasks related to articulated object manipulation and artistic activities, such as painting or tool use, requiring precise and fine-grained visuomotor control.

The full list of included tasks is available in Table 5, 6.

C Training Details

Our X-Distill model is trained with the hyperparameter configurations summarized in Table 4.

Table 4: Summary of key hyperparameter configurations

Parameter Description	Parameter Name	Value
Diffusion Process		
Number of diffusion timesteps	num_train_timesteps	50
Noise schedule	beta_schedule	squaredcos_cap_v2
Prediction target	prediction_type	epsilon
Network Architecture		
Feature dimension	feature_dim	64
U-Net decoder channels	down_dims	[256, 512, 1024]
Convolution kernel size	kernel_size	5
Group normalization groups	n_groups	8
Condition modulation type	condition_type	film
Training Configuration		
Batch size	batch_size	64
Number of epochs	num_epochs	3000
Base learning rate	lr	0.0001
Optimizer	optimizer	AdamW
Weight decay	weight_decay	0.000001
Gradient accumulation steps	gradient_accumulate_every	1
EMA decay	use_ema	true
Data Configuration		
Observation history steps	n_obs_steps	2
Prediction horizon	horizon	4
Action steps	n_action_steps	4
Data loading workers	num_workers	8
Inference		
Number of denoising steps	num_inference_steps	16

D All Simulation Results

Figure 6 presents the training curves across representative tasks from MetaWorld, Adroit, and DexArt, demonstrating the consistent performance advantages of our X-Distill approach throughout the learning process.

Tables 5 and 6 provide comprehensive quantitative comparisons against strong baselines on all benchmark tasks. Our method achieves superior or competitive performance across the majority of tasks, particularly excelling in challenging manipulation scenarios like handle pulling, peg insertion, and complex multi-step operations. The results highlight X-Distill's robustness across varying task difficulties and embodiment domains.

Tables 7 and 8 present an extensive ablation study examining the impact of different teacher-student architecture combinations. Notably, the DINOv2-L to ResNet-18 configuration emerges as the most effective balance between performance and efficiency, while the consistent superiority of distilled representations over from-scratch training underscores the value of our knowledge distillation approach.

Table 5: **Main results on MetaWorld tasks.** 29 tasks are evenly split into 6 rows (5 tasks per row).

MetaWorld (Easy)

Alg \Task	Lever Pull	Door Close	Drawer Open	Door Lock	Door Unlock
ResNet-scratch	30 ± 18	**100 ± 0**	77 ± 6	70 ± 19	82 ± 5
Theia	48 ± 14	**100 ± 0**	**100 ± 0**	47 ± 3	83 ± 8
Depth-Anything	17 ± 2	**100 ± 0**	63 ± 5	48 ± 2	78 ± 13
DINOv2	47 ± 5	**100 ± 0**	**100 ± 0**	63 ± 2	90 ± 4
X-Distill (Ours)	**75 ± 8**	**100 ± 0**	**100 ± 0**	**100 ± 0**	**100 ± 0**
PointNet-DP3	79 ± 8	100 ± 0	100 ± 0	100 ± 0	100 ± 0

MetaWorld (Easy)

Alg \Task	Drawer Close	Faucet Close	Faucet Open	Handle Press	Handle Pull
ResNet-scratch	**100 ± 0**	**100 ± 0**	**100 ± 0**	83 ± 13	25 ± 22
Theia	**100 ± 0**	3 ± 3	7 ± 3	**100 ± 0**	13 ± 10
Depth-Anything	**100 ± 0**	88 ± 6	**100 ± 0**	85 ± 7	18 ± 2
DINOv2	**100 ± 0**	93 ± 2	**100 ± 0**	**100 ± 0**	28 ± 5
X-Distill (Ours)	**100 ± 0**	**100 ± 0**	**100 ± 0**	**100 ± 0**	**95 ± 4**
PointNet-DP3	100 ± 0	100 ± 0	100 ± 0	100 ± 0	45 ± 8

MetaWorld (Easy)

Alg \Task	Handle Pull Side	Plate Slide	Plate Slide Back	Plate Slide Back Side	Plate Slide Side
ResNet-scratch	3 ± 5	90 ± 14	**100 ± 0**	**100 ± 0**	**100 ± 0**
Theia	12 ± 16	40 ± 30	38 ± 8	62 ± 47	2 ± 3
Depth-Anything	12 ± 2	80 ± 12	**100 ± 0**	**100 ± 0**	**100 ± 0**
DINOv2	48 ± 5	80 ± 4	**100 ± 0**	**100 ± 0**	**100 ± 0**
X-Distill (Ours)	**95 ± 7**	**100 ± 0**	**100 ± 0**	**100 ± 0**	**100 ± 0**
PointNet-DP3	100 ± 0	100 ± 0	100 ± 0	100 ± 0	100 ± 0

MetaWorld (Easy)

Alg \Task	Reach Wall	Window Close	Window Open	Reach	Peg unplug side
ResNet-scratch	**77 ± 2**	**100 ± 0**	93 ± 10	47 ± 13	55 ± 8
Theia	67 ± 3	42 ± 6	95 ± 5	48 ± 6	10 ± 0
Depth-Anything	48 ± 10	90 ± 14	70 ± 4	45 ± 4	22 ± 6
DINOv2	53 ± 6	**100 ± 0**	78 ± 13	**52 ± 2**	38 ± 2
X-Distill (Ours)	73 ± 6	**100 ± 0**	**100 ± 0**	**52 ± 6**	**87 ± 2**
PointNet-DP3	68 ± 3	100 ± 0	100 ± 0	24 ± 1	92 ± 2

MetaWorld (Medium)

Alg \Task	Coffee Push	Bin picking	Coffee Pull	Push Wall	Peg Insert Side
ResNet-scratch	82 ± 10	68 ± 9	55 ± 4	48 ± 10	28 ± 13
Theia	32 ± 3	10 ± 0	2 ± 3	3 ± 6	0 ± 0
Depth-Anything	38 ± 2	32 ± 2	47 ± 6	33 ± 2	13 ± 2
DINOv2	35 ± 0	52 ± 13	52 ± 6	48 ± 6	35 ± 4
X-Distill (Ours)	**97 ± 5**	**95 ± 4**	**95 ± 4**	**80 ± 0**	**88 ± 2**
PointNet-DP3	95 ± 4	65 ± 19	85 ± 11	49 ± 8	72 ± 5

MetaWorld (Medium / Hard / Very Hard)

Alg \Task	Sweep	Sweep into	Pick out of hole	Disassemble
ResNet-scratch	22 ± 2	33 ± 9	38 ± 9	50 ± 29
Theia	12 ± 3	37 ± 3	0 ± 0	38 ± 6
Depth-Anything	20 ± 4	22 ± 6	42 ± 10	43 ± 6
DINOv2	48 ± 5	52 ± 5	**48 ± 9**	38 ± 5
X-Distill (Ours)	**85 ± 4**	**78 ± 5**	**48 ± 6**	**88 ± 6**
PointNet-DP3	83 ± 5	45 ± 16	14 ± 9	72 ± 6

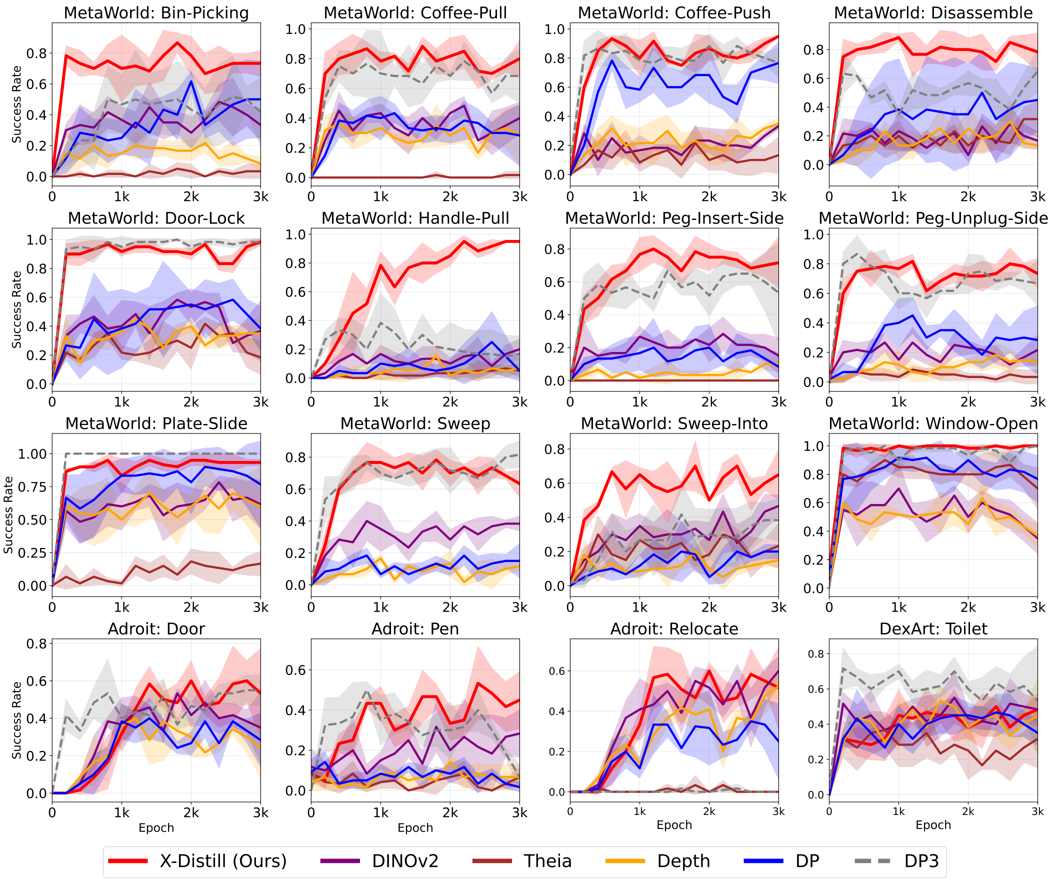

Figure 6: **Training curves on representative simulation tasks.** Success rates are shown for selected tasks from MetaWorld, Adroit, and DexArt.

Table 6: **Main results on Adroit and Dexart tasks.** Tasks are grouped by difficulty and arranged across five visually aligned rows.

	Adroit			**Dexart**	
Alg \ Task	Door	Pen	Relocate	Laptop	Toilet
ResNet-scratch	47 ± 7	18 ± 2	48 ± 8	52 ± 5	57 ± 2
Theia	7 ± 2	14 ± 1	5 ± 4	0 ± 0	48 ± 2
Depth-Anything	52 ± 6	16 ± 1	53 ± 5	70 ± 4	62 ± 2
DINOv2	57 ± 6	38 ± 11	60 ± 7	53 ± 13	**63 ± 5**
X-Distill (Ours)	**73 ± 9**	**60 ± 11**	**72 ± 5**	**65 ± 4**	62 ± 2
PointNet-DP3	67 ± 6	52 ± 6	3 ± 2	90 ± 4	80 ± 0

Table 7: **Main results on MetaWorld tasks.** Tasks are grouped by difficulty and arranged across visually aligned sections.

Metaworld - Easy Tasks

Teacher	Student	Lever Pull	Door Close	Drawer Open	Door Lock
DINOv2-L	ResNet-18 (11M)	75 ± 8	**100 ± 0**	**100 ± 0**	**100 ± 0**
	ViT-S-Half (11M)	40 ± 4	**100 ± 0**	**100 ± 0**	68 ± 10
	ConvNeXt (89M)	73 ± 8	**100 ± 0**	**100 ± 0**	**100 ± 0**
DINOv2-S	ResNet-18 (11M)	**82 ± 12**	**100 ± 0**	**100 ± 0**	**100 ± 0**

Metaworld - Easy Tasks

Teacher	Student	Door Unlock	Drawer Close	Faucet Close	Faucet Open
DINOv2-L	ResNet-18 (11M)	**100 ± 0**	**100 ± 0**	**100 ± 0**	**100 ± 0**
	ViT-S-Half (11M)	68 ± 2	**100 ± 0**	22 ± 17	**100 ± 0**
	ConvNeXt (89M)	**100 ± 0**	**100 ± 0**	**100 ± 0**	**100 ± 0**
DINOv2-S	ResNet-18 (11M)	**100 ± 0**	**100 ± 0**	**100 ± 0**	**100 ± 0**

Metaworld - Easy Tasks

Teacher	Student	Handle Press	Handle Pull	Handle Pull Side	Plate Slide
DINOv2-L	ResNet-18 (11M)	**100 ± 0**	95 ± 4	**95 ± 7**	**100 ± 0**
	ViT-S-Half (11M)	98 ± 2	13 ± 5	10 ± 0	93 ± 2
	ConvNeXt (89M)	**100 ± 0**	80 ± 15	78 ± 6	98 ± 2
DINOv2-S	ResNet-18 (11M)	**100 ± 0**	**98 ± 3**	90 ± 13	**100 ± 0**

Metaworld - Easy Tasks

Teacher	Student	Plate Slide Back	Plate Slide Back Side	Plate Slide Side	Reach Wall
DINOv2-L	ResNet-18 (11M)	**100 ± 0**	**100 ± 0**	**100 ± 0**	73 ± 6
	ViT-S-Half (11M)	85 ± 7	**100 ± 0**	**100 ± 0**	72 ± 5
	ConvNeXt (89M)	**100 ± 0**	**100 ± 0**	**100 ± 0**	67 ± 3
DINOv2-S	ResNet-18 (11M)	**100 ± 0**	**100 ± 0**	**100 ± 0**	**75 ± 5**

Metaworld - Easy Tasks

Teacher	Student	Window Close	Window Open	Coffee Push	Bin Picking
DINOv2-L	ResNet-18 (11M)	**100 ± 0**	**100 ± 0**	**97 ± 5**	**95 ± 4**
	ViT-S-Half (11M)	92 ± 8	95 ± 4	35 ± 4	10 ± 4
	ConvNeXt (89M)	**100 ± 0**	**100 ± 0**	88 ± 8	88 ± 6
DINOv2-S	ResNet-18 (11M)	**100 ± 0**	**100 ± 0**	95 ± 5	85 ± 9

Metaworld - Medium Tasks

Teacher	Student	Reach	Peg Unplug Side	Coffee Pull	Push Wall
DINOv2-L	ResNet-18 (11M)	**52 ± 6**	87 ± 2	95 ± 4	80 ± 0
	ViT-S-Half (11M)	50 ± 4	33 ± 2	23 ± 18	37 ± 6
	ConvNeXt (89M)	52 ± 3	**88 ± 8**	83 ± 3	73 ± 8
DINOv2-S	ResNet-18 (11M)	**52 ± 6**	**88 ± 6**	**95 ± 0**	**83 ± 3**

Table 8: **Main results on MetaWorld tasks (continued).**

Metaworld - Medium/Hard Tasks					
Teacher	Student	Peg Insert Side	Sweep	Sweep Into	Pick Out of Hole
DINOv2-L	ResNet-18 (11M)	88 ± 2	85 ± 4	78 ± 5	48 ± 6
	ViT-S-Half (11M)	7 ± 6	45 ± 8	20 ± 4	2 ± 2
	ConvNeXt (89M)	57 ± 16	75 ± 5	$\mathbf{78 \pm 6}$	$\mathbf{50 \pm 5}$
DINOv2-S	ResNet-18 (11M)	$\mathbf{90 \pm 5}$	$\mathbf{85 \pm 5}$	78 ± 3	43 ± 16

Very Hard Tasks		
Teacher	Student	Disassemble
DINOv2-L	ResNet-18 (11M)	88 ± 6
	ViT-S-Half (11M)	40 ± 7
	ConvNeXt (89M)	83 ± 8
DINOv2-S	ResNet-18 (11M)	$\mathbf{90 \pm 5}$

E Real-world Setup

Our real-world evaluation setup, depicted in Figure 7, features a 6-DoF X-Arm robot manipulator controlled via policies trained with our method. The system utilizes a UGreen camera for visual perception and a Meta-Quest headset for human demonstration data collection, enabling comprehensive evaluation of manipulation capabilities in physical environments.

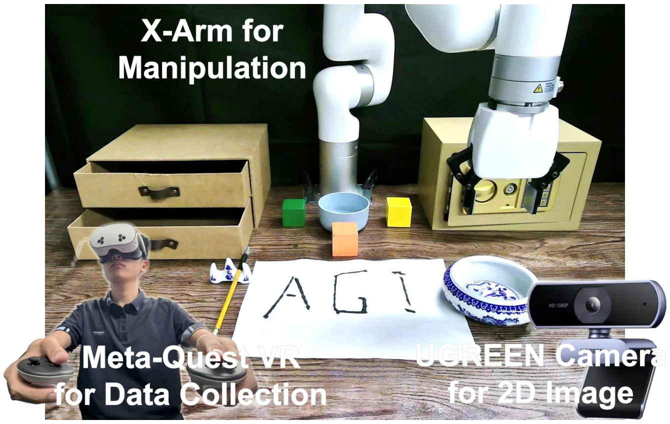

Figure 7: **Hardware setup** comprising the 6-DoF X-Arm robot for control, Meta-Quest headset for data collection, and UGreen camera for visual perception.

F Keyframe sequence from real-world task

The following figures present keyframe sequences from our real-world robotic manipulation experiments, illustrating both successful executions and representative failure cases for each task.

Move Cube Task: Successful completion requires the robot to reliably pick up the cube and place it into the bowl. Common failure modes include the gripper colliding with the cube, triggering emergency stops due to excessive force, or failing to establish a secure grasp on the object (see Figure 8).

Writing “AGI” Task: Success is defined by the accurate writing of the letters "AGI". Typical failures include writing incorrect characters, repeatedly writing the same letter without progression, or complete failure to produce any legible writing (see Figure 9).

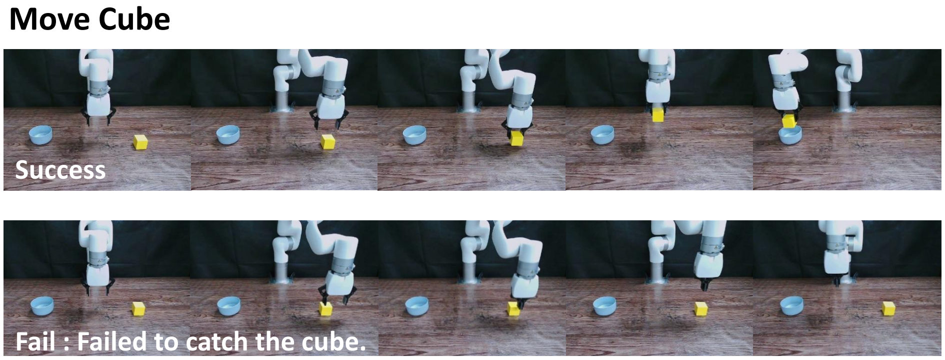

Figure 8: **Move Cube task keyframe sequence.** Illustrates successful executions and typical failure cases of the Move Cube task.

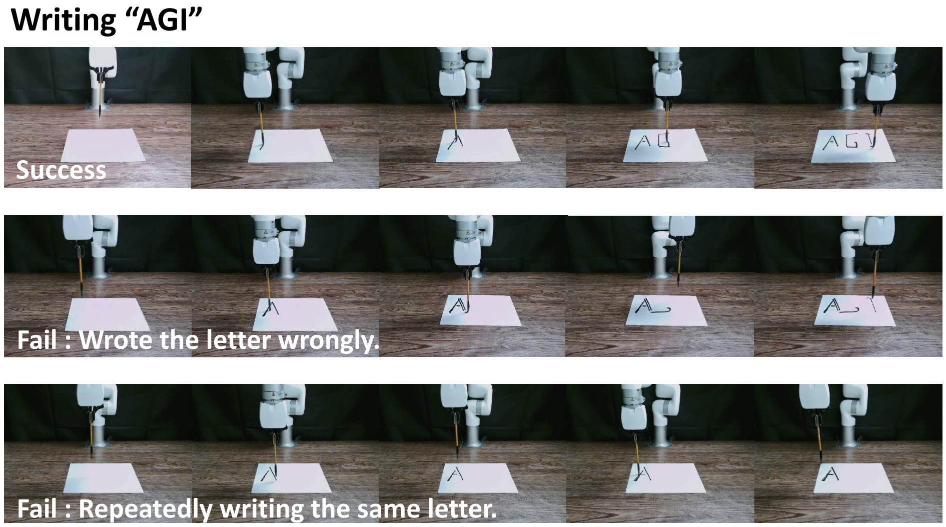
Writing "AGI"
Success
Fail : Wrote the letter wrongly.
Fail : Repeatedly writing the same letter.

Figure 9: **Writing "AGI" task keyframe sequence.** Illustrates successful executions and typical failure cases of the writing "AGI" task.

Move Brush Task: Successful execution involves correctly orienting and placing the brush into the holder. Common failures include incorrect brush orientation, collisions with the brush handle triggering emergency stops, or complete misses when attempting to grasp the brush (see Figure 10).

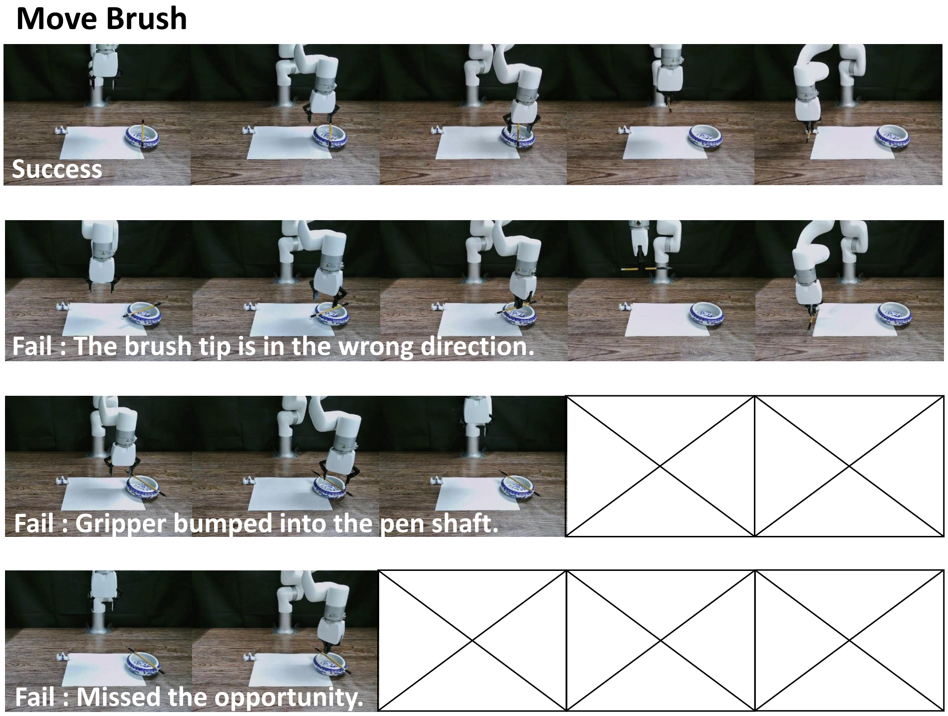
Move Brush
Success
Fail : The brush tip is in the wrong direction.
Fail : Gripper bumped into the pen shaft.
Fail : Missed the opportunity.

Figure 10: **Move Brush task keyframe sequence.** Illustrates successful executions and typical failure cases of the move brush task.

Drawer Open Task: Success requires smoothly opening the drawer without collisions. The primary failure mode involves the gripper colliding with the drawer edges, preventing proper engagement and manipulation (see Figure 11).

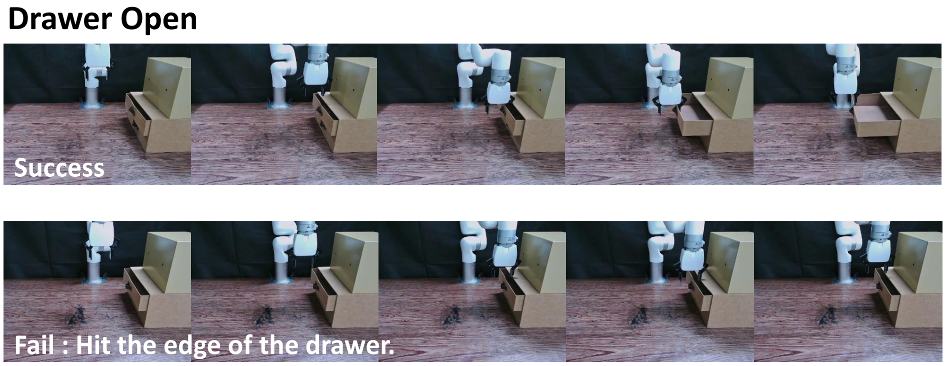

Figure 11: **Drawer Open task keyframe sequence.** Illustrates successful executions and typical failure cases of the drawer open task.

Door Close Task: Successful completion entails securely closing the door. The main failure occurs when the gripper fails to make proper contact with the door surface, resulting in inability to initiate or complete the closing motion (see Figure 12).

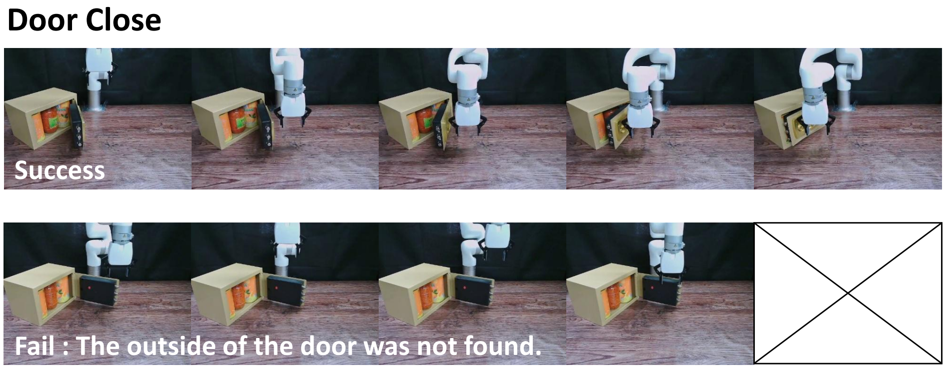

Figure 12: **Door Close task keyframe sequence.** Illustrates successful executions and typical failure cases of the door close task.

G ADDITIONAL EXPERIMENTAL RESULTS

In this section, we provide comprehensive results from additional experiments conducted to investigate the impact of different pre-training objectives, parameter-efficient fine-tuning methods, and data scaling laws.

G.1 COMPARISON WITH R3M AND PEFT (LORA)

To further validate the effectiveness of our X-Distill framework, we compared it against two important baselines on the 29 MetaWorld tasks (10 demonstrations per task):

- **R3M** (Nair et al., 2022): A ResNet-18 encoder pre-trained on Ego4D human videos. This comparison isolates the effect of the pre-training objective (video-language vs. our distillation) since the architecture is identical to ours.
- **DINOv2 + LoRA**: We applied Low-Rank Adaptation (LoRA) (Hu et al., 2022) to the DINOv2-Small encoder (rank=16, applied to q, k, v matrices) during policy learning. This tests whether parameter-efficient fine-tuning can bridge the gap between ViTs and CNNs in the low-data regime.

The results are summarized in Table 9.

Table 9: Comparison against R3M (alternative pre-training) and LoRA (PEFT) on MetaWorld tasks (10 demos). X-Distill significantly outperforms both, highlighting the superiority of our distillation objective and architectural choice.

Method	Configuration	Success Rate (%)
X-Distill (Ours)	ResNet-18 (Distilled)	**90.8**
DINOv2 (Full Finetune)	ViT-S (Pre-trained)	68.2
DINOv2 + LoRA	ViT-S (PEFT, rank=16)	74.5
ResNet-scratch	ResNet-18 (Random Init)	67.4
R3M	ResNet-18 (Frozen, Ego4D)	14.2

Analysis. First, **R3M** achieves a low success rate of 14.2%, significantly underperforming even the scratch-trained ResNet. This suggests that while R3M's representations are temporally smooth for video analysis, they lack the fine-grained spatial precision required for manipulation control, validating that our specific distillation from DINOv2 is critical. Second, while **LoRA** (74.5%) indeed improves over full fine-tuning (68.2%) by mitigating catastrophic forgetting, it still significantly underperforms X-Distill (90.8%). This refutes the hypothesis that PEFT is the optimal solution for this specific low-data regime and confirms that the architectural inductive bias of the CNN student is the dominant factor for success.

G.2 Data Scaling Analysis

To understand the scalability of our method and identify the potential "crossover point" where ViTs might outperform CNNs, we conducted a data scaling experiment on the real-world “Move Brush” task. We trained policies using $N = \{20, 40, 60, 80, 100\}$ demonstrations. The results are visualized in Figure 13 and detailed in Table 10.

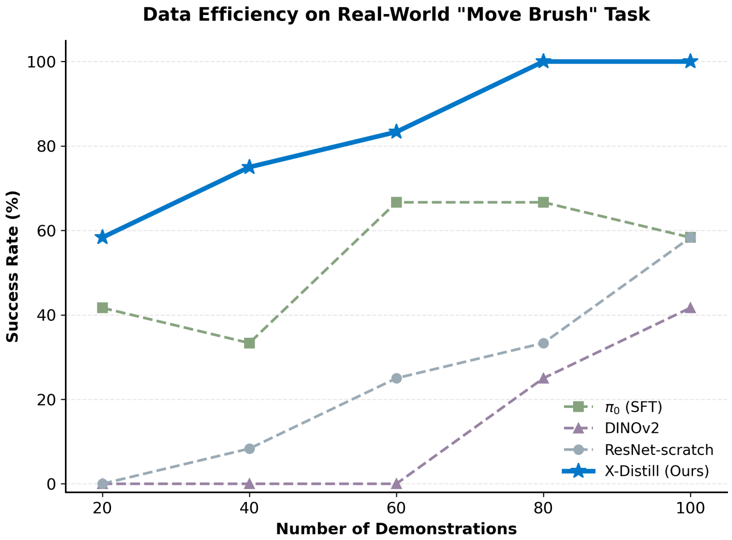

Figure 13: **Data scaling efficiency analysis.** X-Distill (Blue) demonstrates superior data efficiency, saturating at 100% success with only 80 demos. ViT-based baselines start slowly, and generalist VLAs (π_0) show instability.

Table 10: Detailed success rates (out of 12 trials) and percentages for the Data Scaling experiment on the “Move Brush” task.

Method	Number of Demonstrations				
	20	40	60	80	100
ResNet-scratch	0.0%	8.3%	25.0%	33.3%	58.3%
DINOv2	0.0%	0.0%	0.0%	25.0%	41.7%
π_0 (SFT)	41.7%	33.3%	66.7%	66.7%	58.3%
X-Distill (Ours)	**58.3%**	**75.0%**	**83.3%**	**100.0%**	**100.0%**

Analysis. X-Distill demonstrates extreme data efficiency, yielding workable policies ($> 50\%$) with as few as 20 demos and reaching perfect performance at 80 demos. In contrast, the ViT baseline (DINOv2) requires significantly more data to begin learning, only reaching 41.7% at 100 demos. This trend suggests that the crossover point where a ViT might outperform our distilled CNN lies at thousands of trajectories, far beyond the typical data regime for accessible robotic learning.