Dexterity-BEV: Aligning 3D World and Actions for Generalizable Robot Policies Learning

Huayi Zhou[*1,2] **Wei Gao**[*1] **Dekun Lu**[1] **Ruiji Liu**[1] **Zhanqi Zhang**[1]
Ziyang Zhang[1] **Jian Chen**[1] **Wenlve Zhou**[1] **Sheng Xu**[2] **Shumin Li**[1]
Kangyi Guo[1] **Shichen Xu**[1] **Zixin Huang**[1] **Yongyi Su**[1] **Kui Jia**[‡1,2]
[1]DexForce Technology [2]The Chinese University of Hong Kong, Shenzhen
[*]Equal Contribution [‡]Corresponding Author

Abstract: End-to-end manipulation policies, combined with web-scale pretrained Vision-Language Models (VLMs), show the promise for generalizable and dexterous robotic manipulation. However, they inherit two key limitations from 2D foundation models: 1) the reliance on 2D RGB inputs that ignores the intrinsically 3D nature of manipulation; and 2) the lack of spatial 3D alignment between input-output spaces as well as across diverse robot embodiments, camera setups, and trajectory datasets. In this paper, we present a series of contributions to address these issues. First, we introduce *aligned vertex map* and *vertex spectrum* — a pixel-wise 3D representation that elevates 2D visual inputs to 3D, using camera calibration and optional depth. This novel input representation marries 3D awareness with the generalization of 2D large VLMs. Then, we propose to align the inputs and outputs of manipulation policies by expressing per-pixel 3D information of each camera view and robot actions to a shared coordinate. Based on this, we designate a canonical *Bird's-Eye-View (BEV) alignment frame* and innovatively propose to construct BEV images, producing a view-invariant representation robust to camera pose variations. To enable training and evaluation at scale, we develop a comprehensive data processing pipeline to perform such alignments; we also introduce a novel temporal alignment scheme for trajectories across diverse robots, human operators, and datasets. These contributions collectively mitigate input and output spatial-temporal misalignments, improving the consistency and generalization for real-world manipulation. Pretrained checkpoint, source code and data processing pipeline are available in https://hnuzhy.github.io/projects/Dex-BEV.

Keywords: End-to-end Manipulation, VLAs, Spatial-Temporal Alignment, BEV

1 Introduction

End-to-end manipulation policies [1, 2, 3] offer significant potential for enabling embodied agents to understand and interact with the world. The success of Large Language Models (LLMs) [4, 5, 6], Vision-Language Models (VLMs) [7, 8, 9] and (video) World-Models [10, 11, 12] has injected new inspiration into manipulation research. Benefits from web-scale pretraining, these foundation models demonstrate promising zero-shot generalization. Consequently, researchers aim to imbue robots with similar generalization capability to build robotics foundation models.

With this motivation, researchers are increasingly exploring Vision-Language-Action (VLAs) [13, 14, 15, 16, 17, 18]. Some contributions augment VLAs with future video stream prediction, thus leading to World-Action Models (WAMs) [19, 20, 21, 22]. These models are usually derived from pretrained 2D VLMs and further trained on manipulation datasets, typically consisting of corresponded RGB frames, robot/human action trajectories and task instructions. Many dexterous manipulation behaviors challenging for traditional modular perception-planning-action pipelines, automatically emerge from these models [1, 13, 14].

These VLAs/WAMs inherit strong capability from VLMs in terms of visual perception and textual understanding. However, VLMs are typically trained on two-dimensional (2D) RGB image and

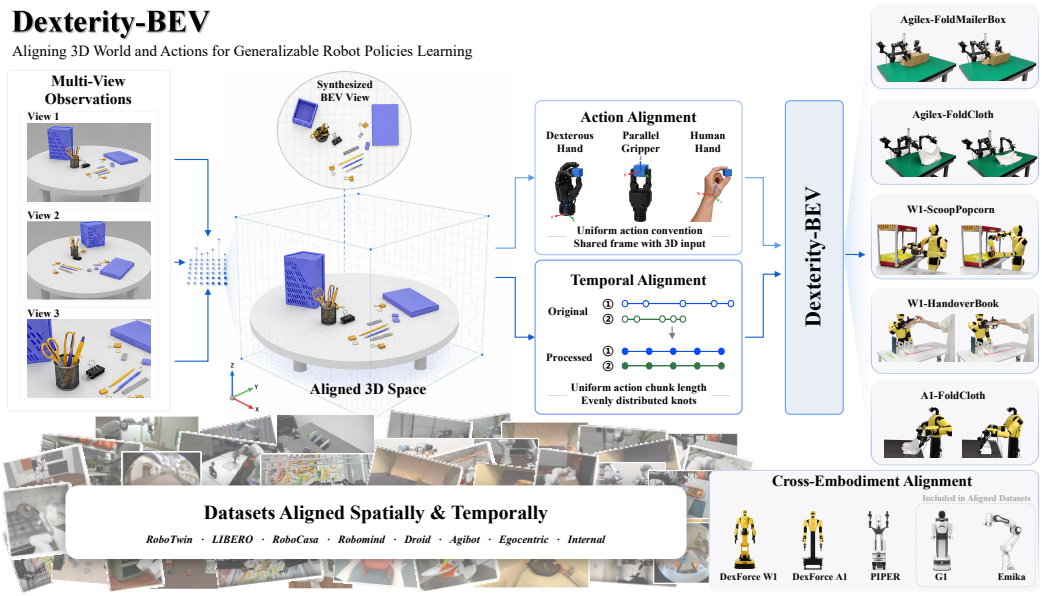

Figure 1: We introduce Dexterity-BEV (Dex-BEV), a series of technical and systematic contributions for manipulation policy learning that generalizes among different embodiments, camera views and datasets. In particular, we introduce 3D input representations that easily integrated with pretrained 2D VLMs; spatial alignment between multi-view cameras & robot actions; and temporal alignment between trajectories from different robots and/or tele-operators. These concepts lead to a comprehensive data processing pipeline and trajectory datasets aligned spatially and temporally.

video inputs, depite robotic manipulation is intrinsically three-dimensional (3D). As a result, most existing endeavors [13, 14, 15, 16, 17, 18] lack explicit 3D information from camera input, such as camera calibration results (intrinsic and extrinsic matrices) and depth images. Consequently, several other contributions [23, 24, 25, 26, 27] have explored alternative 3D inputs, such as point clouds and voxels. However, datasets with these 3D representations are not yet comparable to the 2D counterparts in terms of scale and diversity, which limits the generalization of pretrained 3D VLMs/encoders, and consequently the capability of derived manipulation polices.

Moreover, the output space of existing VLAs/WAMs is typically *not aligned* in terms of: 1) misalignment with (2D) input observations; and 2) misalignment across different embodiments (robots), datasets and manipulation scenarios. In particular, existing VLAs usually produce joint angles or end-effector (EE) poses as output. Joint angles depend on robot types, thus joint trajectories to accomplish the same manipulation task can be vastly different for two types of robots. On the other hand, the EE pose values depend on robot types, frame conventions and designations of the "world" frame (in which EE poses are expressed). For instance, the "world" frame depends on the table setup in the LIBERO [28] dataset; many bi-arm manipulators (e.g., CobotMagic) express left/right EE poses in the base frames of left/right sub-arms, thus these EE pose values cannot reflect the base offset between two sub-arms. These spatial misalignment in joint or EE space causes additional (sometimes unnecessary) variations on action trajectory distribution that end-to-end models must overcome. In addition to 3D spatial misalignment, robot trajectory instances for a task might take different amounts of time, due to the variations in robot hardware setup and human tele-operation. This *temporal misalignment* imply that the policy must address different geometric "length" of action chunks (explained in Subsec. 3.1 and Subsec. 3.4). All these misalignments challenge the expressiveness and generalization of policy learning.

In this paper, we make a series of technical and system contributions to mitigate these limitations. In particular, **1)** we introduce *aligned vertex map* and *vertex spectrum* formation, previously used in other fields such as 3D reconstruction [29, 30] and autonomous driving [31, 32], into VLAs as input space. This formulation elevates 2D-centric model inputs to 3D by providing per-pixel 3D information, exploiting camera calibrations and optional depth images. Thus, we aim to combine

the benefit of 3D input space with the generalization of vision and language foundation models, pretrained on web-scale 2D datasets. Moreover, **2)** we propose to align multi-view observations and output actions, by expressing per-pixel 3D information of each camera view, robot proprioceptive measurements and actions to a shared coordinate, thanks to camera extrinsic parameters. Base on these formulations, **3)** we propose to designate *BEV frames* (refer to Sub. 3.3 for more details) as the alignment frame, and innovatively construct BEV images that are less-variant to different camera setups and change in camera view points, inspired by contributions in autonomous driving [31, 32]. To facilitate the training, evaluation and deployment of our models, we devise a comprehensive data processing pipeline with the following distinctions. Systematically, **4)** we implement 3D spatial alignment for both internal and public datasets, by combining manual operations (assisted by a customized GUI application), rule-based algorithms and vision foundation models. In addition to spatial alignment, **5)** we propose to align different trajectories temporally among different robots, tele-operators and datasets. These contributions constitute the unified Dexterity-BEV (Dex-BEV) architecture and training receipt.

While these contributions above are generally applicable to both VLAs and WAMs, in this paper we focus on VLAs as the instantiated ones and defer WAMs, or the prediction of explicit future (3D) state, to a later study. Simulated and real-world experiments show that Dexterity-BEV achieves significant performance improvements given variations of camera views, robot base poses, and/or manipulation scenarios. We will make the code and data pipeline publicly available.

2 Related Works

VLAs and WAMs. The scaling of diverse robotic demonstrations pre-training has rapidly advanced VLA models. Pioneering works such as [16, 17, 13] validated the efficacy of VLA models derived from 2D VLMs. Efficient teleoperation systems like ALOHA [33, 34] enabled large-scale dataset collection and spurred various VLA datasets [35, 36, 37]. Many contributions [38, 39, 14, 40] explore VLA models with different architectures, learning algorithms and auxiliary tasks. One prominent example is future video generation in WAMs [19, 41, 21, 20, 22, 10]. However, these models rely predominantly on 2D image backbones. The lack of 3D input might lead to performance degradation in terms of precision and robustness to unseen camera view points.

3D Representations in VLAs/WAMs. Consequently, many contributions [42, 43, 24, 44] attempt to incorporate various form of 3D input into VLAs/WAMs. It is straightforward to use point cloud, voxel grid, and 3D Gaussian Splatting [45, 25, 46, 24] as input. However, these pure 3D representations cannot benefit from VLM backbones pretrained on web-scale 2D image and video datasets. Another branch of contributions [44, 43, 42] fuse 3D information into 2D VLM backbones, and our method falls into this category. Existing methods, based on depth image [43], stereo [44] or camera-frame vertex map [47], typically process each camera view independently; therefore, correlation information between multiple camera views (e.g., head and wrist cameras) is not provided directly to the models. Instead, we propose to provide this information by expressing all vertex maps/spectrums in a shared BEV frame. This idea is extended to achieve alignment between multi-view observations, robot proprioception, and action trajectories. The proposed BEV image is inspired by BridgeVLA [48] and autonomous driving contributions [49, 50, 31, 51]. Compared with [48], we further augment the RGB BEV image with a pixel-aligned vertex map. Then, an alternative network architecture and training receipt are used with emphasis on reactive manipulation tasks (e.g, cloth folding), which might be challenging for the classical motion planner in [48].

3 Methodology

Dex-BEV elevates 2D-centric models into a spatially aligned 3D-aware representation for both observations and actions. This section is organized as follows: Subsec. 3.1 provides preliminaries about VLM and VLA. Subsec. 3.2 details our *Aligned Vertex Map Formulation* for projecting pixel features into a shared 3D frame. Subsec. 3.3 extends our formulation with *BEV Frame, BEV Image Construction and Vertex Spectrum*. Finally, Subsec. 3.4 presents the *Data Processing Pipeline* for 3D spatial standardization and temporal trajectory alignment.

3.1 Preliminary

Most VLAs are derived from pretrained VLMs, which extract visual-textual representations from 2D images and instructions. Given an RGB image $\mathbf{I}_{t,i} \in \mathbb{R}^{H\times W\times 3}$ from the i-th camera at step t and an instruction $\mathcal{L}$, the VLM extracts visual tokens $\mathbf{F}_{t,i} = \mathsf{Enc}_{vis}(\mathbf{I}_{t,i})$ and language tokens $\mathbf{E}_{lang} = \mathsf{Enc}_{lang}(\mathcal{L})$. Multi-view visual tokens are aggregated into $\tilde{\mathbf{F}}_t$, which is further fused into contextual embedding $\mathbf{c}_t = \mathcal{F}_\theta(\tilde{\mathbf{F}}_t, \mathbf{E}_{lang})$.

VLAs predict robot actions from multimodal state $\mathcal{X}_t = \{\{(\mathbf{O}_{t,i}, \mathbf{K}_i, \mathbf{T}_{t,i})\}_{i=1}^N, \mathcal{L}, \mathbf{s}_t\}$ at each step t. $\mathbf{O}_{t,i}$ contains an RGB image $\mathbf{I}_{t,i}$ and an optional, pixel-aligned depth map $\mathbf{D}_{t,i} \in \mathbb{R}^{H\times W}$, where N is the number of cameras. The matrices $\mathbf{K}_i \in \mathbb{R}^{3\times 3}$ and $\mathbf{T}_{t,i} \in SE(3)$ denote camera intrinsics and extrinsics, respectively. Given the input $\mathcal{X}_t$, the VLA policy predicts a chunk of M future actions $\{\mathbf{A}_{t+m}\}_{m=1}^M$. Recent VLA models condition an action decoder on the VLM embedding $\mathbf{c}_t$ using Flow Matching (FM) [52, 14, 40] to model precise action distributions. FM trains a vector field $\mathbf{v}_\theta(\mathbf{a}_\sigma, \sigma, \mathbf{c}_t)$ along a probability path $\psi_\sigma(\mathbf{a}) = \sigma\mathbf{a}_1 + (1-\sigma)\mathbf{a}_0$ between Gaussian noise $\mathbf{a}_0 \sim \mathcal{N}(0, \mathbf{I})$ and ground-truth actions $\mathbf{a}_1$ by minimizing:

$$\mathcal{L}_{FM} = \mathbb{E}_{\sigma\sim\mathcal{U}[0,1],\mathbf{a}_1\sim p_{data},\mathbf{a}_0\sim p_0}\left[\|\mathbf{v}_\theta(\sigma\mathbf{a}_1 + (1-\sigma)\mathbf{a}_0, \sigma, \mathbf{c}_t) - (\mathbf{a}_1 - \mathbf{a}_0)\|^2\right]. \tag{1}$$

During inference, the action sequence is sampled via an ODE solver: $\mathbf{a}_1 = \mathbf{a}_0 + \int_0^1 \mathbf{v}_\theta(\mathbf{a}_\sigma, \sigma, \mathbf{c}_t)d\sigma$.

3.2 Aligned Vertex Map Formulation

Following Subsec. 3.1, the observation at step t is defined as $\mathcal{X}_t = \{\{(\mathbf{O}_{t,i}, \mathbf{K}_i, \mathbf{T}_{t,i})\}_{i=1}^N, \mathcal{L}, \mathbf{s}_t\}$. In this subsection, we assume all cameras are calibrated and depth images are available, thus the observation becomes $\mathbf{O}_{t,i} = (\mathbf{I}_{t,i}, \mathbf{D}_{t,i})$. *This assumption is relaxed in Sec. 3.3 to address setups without depth images on one or more camera views.* Given depth map $\mathbf{D}_{t,i}$ and intrinsics $\mathbf{K}_i$, the pixel (u, v) is back-projected to obtain a 3D vertex in the i-th camera frame:

$$\mathbf{P}_{camera\_i}(u, v) = \mathbf{K}_i^{-1}[u, v, 1]^T\mathbf{D}_{t,i}(u, v), \tag{2}$$

where $\mathbf{P}_{camera\_i}$ is a vertex map, and the time subscript t is omitted for clarity. The 2D pixel structure of $\mathbf{P}_{camera\_i}$ enables easy integration into 2D VLMs. Prior methods like SpatialVLA [42] directly leverage this local map to formulate 3D positional embeddings for visual features:

$$\mathbf{F}_{\mathbf{combined\_i}} = \mathbf{F}_{\mathbf{img\_i}} + \mathbf{F}_{\mathbf{3d\_i}} = \mathsf{Enc}_{vis}(\mathbf{I}_{t,i}) + \mathsf{Enc}_{3d}(\mathbf{P}_{camera\_i}). \tag{3}$$

However, local vertex maps $\mathbf{P}_{camera\_i}$ lack geometric correlation across distinct viewpoints. A single physical 3D point observed across multiple views will yield highly divergent values due to differing camera extrinsics $\mathbf{T}_{t,i}$ and $\mathbf{T}_{t,j}$. Inspired by contributions [29, 30] in 3D reconstruction, we propose to transform all camera-frame vertex maps into a shared reference frame $\mathbf{T}_{align\_t}$:

$$\mathbf{F}_{\mathbf{3d\_i}} = \mathsf{Enc}_{3d}(\mathbf{P}_{\mathbf{aligned\_i}}) = \mathsf{Enc}_{3d}(\mathbf{T}_{align\_t}^{-1}\mathbf{T}_{\mathbf{t,i}}\mathbf{P}_{camera\_i}). \tag{4}$$

This step ensures that $\mathbf{P}_{\mathbf{aligned\_i}}$ maintains global spatial consistency in 3D while remaining pixel-aligned with RGB image $\mathbf{I}_{t,i}$. Crucially, the robot proprioception $\mathbf{s}_{t,i}$ and target actions $\mathbf{A}_t$ are also represented as $SE(3)$ poses expressed in this shared $\mathbf{T}_{align\_t}$ frame. *Combined with unified 3D frame conventions (detailed in Subsec. 3.4), the entire perception-action loop is tightly integrated within an embodiment-agnostic 3D workspace.*

The $\mathbf{T}_{align\_t}$ is typically the first camera view in 3D reconstruction. In this paper, we instantiate $\mathbf{T}_{align\_t}$ as a canonical Bird's-Eye View (**BEV**) frame and construct additional BEV images, as detailed in Subsec. 3.3.

3.3 Several Extensions and Network Architecture

BEV Frame and BEV Image Construction. To minimize input variations caused by heterogeneous robotic embodiments and diverse camera setups, we formalize the shared alignment frame $\mathbf{T}_{align\_t}$ as a canonical Bird's-Eye View (BEV) reference frame. Following the conventions in autonomous

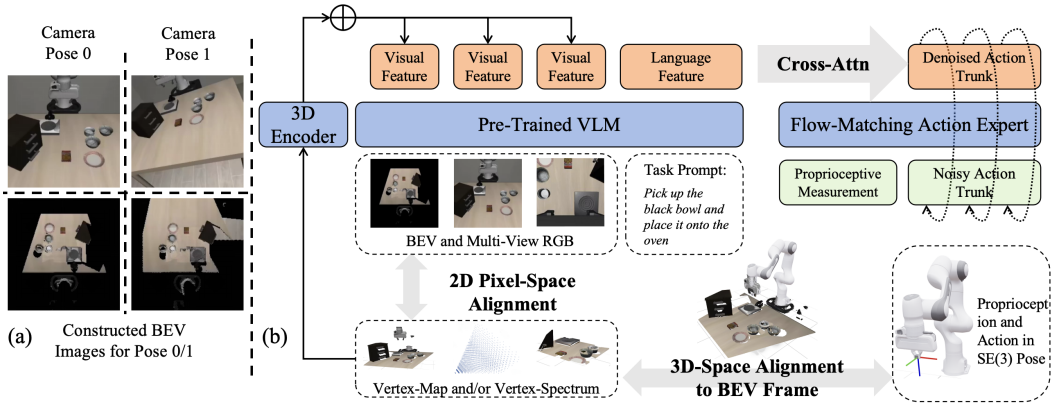

Figure 2: (a) We propose to construct BEV images and associated vertex maps towards invariance to different camera view points. Note that the synthesized BEV images for two vastly different camera poses are very similar to each other, and objects are located at almost identical pixel locations in BEV images. (b) An overview of Dex-BEV architecture. Please refer Sec. 3.3 for a detailed explanation.

driving [31, 32] ("lidar frame") and BridgeVLA [48], we instantiate $\mathbf{T}_{align\_t}$ as either: 1) the robot base frame; or 2) the bottom-center of a 3D cubic region-of-interest (RoI) surrounding the table-top workspace, if the scenario is a table-top manipulation.

For the designated BEV frame, we construct a synthetic BEV image inspired by contributions [49, 50, 31, 51] in autonomous driving. This BEV image is constructed by a top-down orthographic projection of the aggregated colored point clouds from all cameras. Alongside this projection, we compute a corresponding pixel-wise 3D vertex map for the BEV image. An illustration is shown in Fig. 2 (a), the BEV images provide a viewpoint-invariant geometric input space for policy learning.

Vertex Spectrum to Address Optional Depth Observation. Inspired by PETR in autonomous driving [31, 32], we propose generating a *vertex spectrum* for RGB-only views, in order to accommodate platforms without depth sensors. For a pixel $\mathbf{p} = [u, v, 1]^T$ in the i-th camera view, we sample M discrete depth hypotheses d_j using a linear-increasing discretization (LID) [50]:

$$d_j = d_{min} + (d_{max} - d_{min}) \cdot \frac{j(j+1)}{M(M+1)}, \tag{5}$$

where $[d_{min}, d_{max}]$ represents the operational depth range. Each pixel-depth pair is back-projected and transformed via the extrinsic matrix $\mathbf{T}_{t,i}$ into the aligned BEV frame, yielding a volumetric coordinate grid $\mathcal{G}_{u,v} \in \mathbb{R}^{M\times 3}$. This grid is then processed by a lightweight encoder to formulate a 2D positional embedding that is element-wise added to the corresponding RGB features.

Overall Architecture. As illustrated in Fig. 2 (b), the overall architecture ingests these fused multi-view tokens, the synthetic BEV features, 3D vertex maps & vertex spectrum, and the language instruction into a VLM backbone. The extracted multi-modal representations are then processed by a flow-matching action expert [14, 40] to model the target action distribution. Crucially, both proprioceptive measurement and output action for robots are parameterized as $SE(3)$ poses expressed within the unified BEV frame. Thus, the multi-view input and action output are aligned in 3D space, and unified convention can be applied across different embodiments and datasets.

3.4 Data Alignment Processing Pipeline

To facilitate robust training, evaluation, and cross-platform deployment, we implement a comprehensive pipeline for 3D spatial and temporal alignments across heterogeneous datasets.

3D Spatial Alignment. As shown in Fig. 3, for each dataset, camera intrinsics and extrinsics are unified into standard OpenCV formats by combining manual 3D GUI matching, iterative closest point (ICP) registration, and data-driven estimators like DepthAnything V3 [30]. For trajectories lacking active depth measurements, missing channels are re-generated by replaying actions in simulation;

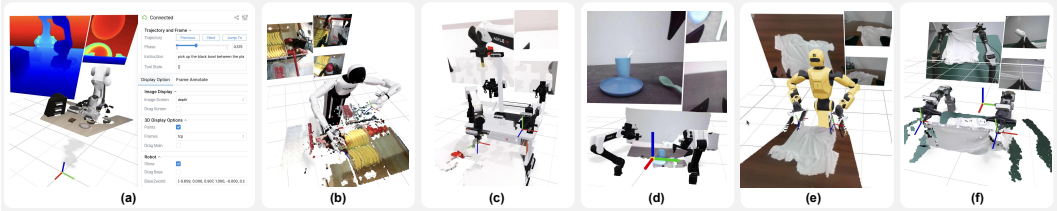

Figure 3: **3D spatial alignment in our data processing pipeline.** (a) We develop a customized GUI application for 3D alignment and visualization, as explained in Subsec. 3.4. In (a-f), we show the 3D alignment of representative public and internal datasets, including (a) LIBERO [28], (b) Agibot-Alpha/Beta [54], (c) RoboTwin 2.0 [55], (d) RoboMind 2.0 [37] and our internal datasets (e-f). We also apply an unified TCP frame convention, as shown in these figures.

for some real-world dataset (e.g., Droid), depth images can be synthesized using vision foundation models such as FoundationStereo [53]. Finally, high-quality robot URDF models are registered to the shared 3D observation space. We enforce a unified tool center point (TCP) convention across disparate embodiments, consistently anchoring parallel-jaw gripper frames at the tip of the jaws and multi-finger configurations at the wrist. These standardized kinematic chains allow us to compute unified absolute $SE(3)$ poses across all platforms using forward kinematics.

Cross-Trajectory Temporal Alignment. The speed of a trajectory can depend on the robot platform and human teleoperation, which creates additional variations on robot trajectories that VLAs must overcome. On the other hand, most manipulation tasks can be regarded as *quasi-static*: a slowed or accelerated (within an extent) trajectory can still accomplish the given manipulation task. Although this is not true for all manipulations (e.g., throwing a ball), nearly all tasks in current VLA datasets are quasi-static. With this observation, we propose normalizing the end-effector speed to a standard value across multiple robots and VLA datasets. In other words, we re-compute the physical time for knots of robot trajectories for temporal alignment. The detailed procedure is in Appendix.

4 Experiments

Our evaluation aims to demonstrate that Dex-BEV provides a superior and more interpretable framework for dexterous robotic manipulation compared to existing 2D and 3D VLA paradigms. We systematically test its efficacy across diverse simulated benchmarks [28, 55, 56] and real-world platforms, focusing on its spatial reasoning capabilities and cross-embodiment generalization.

4.1 Evaluation on Simulation Benchmarks

We perform quantitative comparisons on the LIBERO [28] and RoboTwin-2.0 [55, 56] benchmarks. Our method is compared with two competitive VLA baselines: the π_0 [14] and X-VLA [39] [1]. Moreover, we conduct a 2D ablation study of the proposed method that 1) removes all 3D inputs; and 2) disables 3D alignment by expressing all SE(3) poses following the conventions of X-VLA [39]. As detailed below, we use the official and modified setups to evaluate the generalization of our method with respect to different embodiments, camera viewpoints and robot/scene base poses.

We first evaluate our method on the official setup of LIBERO [28] and RoboTwin-2.0 [56]. These benchmarks are based on different robot platforms, the single-arm 7-DoF franka for LIBERO [28] and dual-arm 12-DoF agile-x for RoboTwin-[56]. The results are shown in Tab. 1. To highlight the generalization to different embodiments, we use one checkpoint (network weight) for both evaluation. The results for baselines are the higher one from our rollout of released checkpoints and the reported results in [39]. Compared with these SOTA baselines, our method achieves roughly the same results on LIBERO [28] and higher success rate on RoboTwin [55] in Tab. 1, despite deploying on vastly different robot platforms. Moreover, the 2D ablation, which use the same input/output as X-VLA [39], suffers major performance drop. This highlights the effectiveness of proposed 3D inputs and alignments.

[1] We emphasize that given the complementary nature of our proposed Dex-BEV, similar quality results would be obtained if comparing with other representative VLAs [15, 40, 13].

Table 1: **Simulation benchmark results and generalization to different embodiments.** We present the success rate for each compared method across task suites in LIBERO [28] and RoboTwin 2.0 [55]. Our method achieves roughly the same results on LIBERO and higher success rate on RoboTwin compared to strong baselines, despite deploying on vastly different robot platforms. In comparison with 2D ablation, the proposed 3D inputs and alignments lead to major improvement.

Method	Cross Embodiments	LIBERO (Official)					RoboTwin 2.0	
		Spatial	Object	Goal	Long	Average	Clean	Randomized
π_0 [14]	False	96.8	98.8	95.8	85.2	94.2	46.4	16.4
X-VLA [39]	False	98.2	98.6	97.8	97.6	98.1	70.0	39.0
2D Ablation	True	93.2	95.0	92.8	90.2	92.8	64.8	35.2
Dex-BEV	True	98.2	98.0	97.8	97.0	97.8	76.0	42.0

Table 2: **Modified LIBERO benchmark to evaluate generalization to camera view points and robot/scene base poses.** The proposed method achieves reasonable success rate despite significant variations on camera viewpoints and base poses of robot & scene (everything except the robot, such as the table and objects).

Method	Modified LIBERO (Mutated Camera & Scene Layout)				
	Spatial	Object	Goal	Long	Average
X-VLA (official ckpt)	<10	<10	<10	<10	<10
2D Ablation	<10	<10	<10	<10	<10
Dex-BEV	92.8	89.4	91.0	86.2	89.9

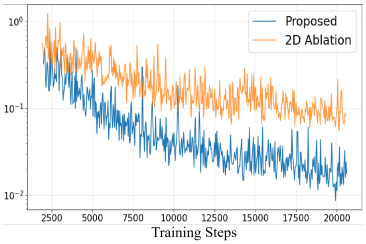

Figure 4: **Training loss comparison.** This corresponds to Tab. 2.

We further conduct simulated experiments to exanimate the generalization to different camera view points and environment layouts. To achieve this, we modify the setup on the LIBERO [28] datasets and platforms. In particular, for each trajectory we randomly modify the third-view camera pose by placing it at different distance and rotating it relative to the world-z axis, the optical axis and the tilting angle. Moreover, we apply local 6-DoF random perturbation to the base pose of the robot and scene (everything except the robot, such as the table and objects) for each trajectory. During the regeneration of LIBERO demonstration trajectory, we first move the robot end-effector to compensate the movement of robot and scene base pose.

The simulation results are presented in Tab. 2. The official X-VLA [39] checkpoint and 2D ablation cannot address strong perturbation of camera poses and scene layouts above. On the other hand, our method achieves a reasonable success rate in this evaluation, benefiting from the representation and alignment of the 3D input. Fig. 4 compares the training dynamics of our method and 2D ablation. The 2D baseline cannot adequately adsorb the pose variations in the training data.

4.2 Evaluation on Real-World Platforms

To validate the practical utility, robustness, and physical precision of Dex-BEV, we deploy our framework across four distinct dual-arm hardware setups: an Agilex bimanual platform, two DexForce wheeled-humanoid robots equipped with two dexterous hands (W1*) or parallel grippers (W1), and a DexForce A1 semi-humanoid robot. Our real-world evaluation comprises five long-horizon tasks that involve intricate bimanual coordination and interactions with deformable, articulated, or granular objects: (1) `Fold Mailer Box` and (2) `Fold Cloth` on the Agilex platform; (3) `Scoop Popcorn` and (4) `Handover Book` on the W1 humanoid; and (5) `Fold Cloth` on the A1 semi-humanoid with . For these different robotic embodiments and corresponding task rollout examples, please refer Fig. 1 right and Fig. 5 for more details. These scenarios present high-dimensional joint synchronization, and multi-contact dynamics, making them inherently challenging for 2D-aware policies. We baseline our framework against strong competitors, including π_0 [14] and X-VLA [39]. As quantitatively shown in Tab. 3, Dex-BEV demonstrates a stable execution profile and commands a significant success rate advantage over all baselines, establishing a new state-of-the-art for physical dual-arm dexterity.

Table 3: Quantitative comparison results of real-robot experiments (reporting average success rates across 30 trails).

Task 1 (Agilex)	`Fold Mailer Box`
π_0 [14]	13/30 (43.3%)
X-VLA [39]	17/30 (56.7%)
Dex-BEV	**23/30 (76.7%)**
Task 2 (Agilex)	`Fold Cloth`
π_0 [14]	20/30 (66.7%)
X-VLA [39]	24/30 (80.0%)
Dex-BEV	**28/30 (93.3%)**
Task 3 (W1*)	`Scoop Popcorn`
π_0 [14]	18/30 (60.0%)
X-VLA [39]	21/30 (70.0%)
Dex-BEV	**26/30 (86.7%)**
Task 4 (W1)	`Handover Book`
π_0 [14]	12/30 (40.0%)
X-VLA [39]	21/30 (70.0%)
Dex-BEV	**28/30 (93.3%)**
Task 5 (A1)	`Fold Cloth`
π_0 [14]	19/30 (63.3%)
X-VLA [39]	23/30 (76.7%)
Dex-BEV	**29/30 (96.7%)**

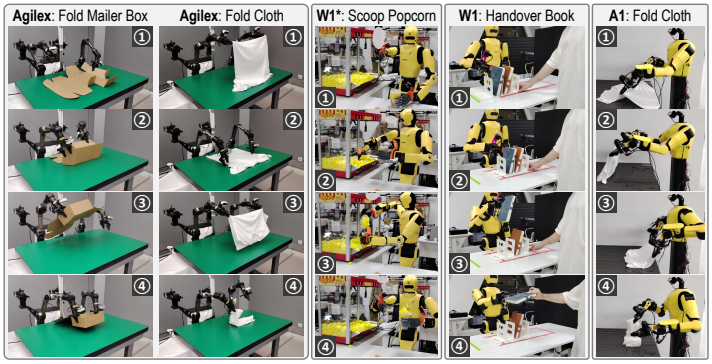

Figure 5: **Qualitative real-world rollouts across different long-horizon complex tasks.** Distinct keyframes demonstrate successful autonomous executions on diverse bimanual robotic platforms involving articulated, deformable, and granular objects (from left to right): `Fold Mailer Box` and `Fold Cloth` on Agilex, `Scoop Popcorn` and `Handover Book` on the DexForce W1 humanoid, and `Fold Cloth` on the DexForce A1 semi-humanoid.

Qualitative rollout sequences, displayed in Fig. 5, highlight the model's remarkable closed-loop reactivity to environmental alterations and its OOD generalization. Specifically, for the folding tasks (`Fold Mailer Box` and `Fold Cloth`), although the training demonstrations were limited to a fixed set of canonical items (e.g., white T-shirts), Dex-BEV achieves successful zero-shot adaptation to completely unseen colors, sizes, and rigidities. The `Scoop Popcorn` task demands fine-grained control over granular materials where the model must dynamically correct its trajectory despite unexpected manual displacements of the target cup. For the `Handover Book` task, Dex-BEV effortlessly handles dynamic human-in-the-loop interactions, accurately tracking a human partner's moving hand and submitting the object despite hand occlusions and unpredictable timing. To sum up, the resilience to dynamic workspace disturbances and large geometric shifts confirms that our framework models the underlying 3D spatial mechanics of a task rather than memorizing superficial 2D visual patterns. Due to page constraints, complete details regarding data collection protocols, teleoperation setups, and additional hardware specifications are expanded in the **Appendix**, with full dynamic executions provided in the **Supplementary Videos**.

5 Conclusion

This paper introduces Dexterity-BEV (Dex-BEV), a framework that establishes a unified input-output 3D alignment for generalizable and dexterous robotic manipulation. We bring in both vertex map and vertex spectrum as input representation for these end-to-end manipulation policies. Then, we designate the BEV frame and propose to construct BEV images, as steps towards spatial transparency and viewpoint invariance. We further propose to align trajectories temporally to mitigate the variance among different robots, tele-operators and datasets. Systematically, we implement a data processing pipeline that combines GUI-assisted manual operations, rule-based algorithms, and vision foundation models for spatial and temporal alignment. Extensive experiments in simulation and real-world demonstrate the efficacy and superiority of our method.

Limitations: Despite these results, Dex-BEV currently relies on camera calibration, which may limit its immediate deployment in unstructured environments where extrinsic parameters are unknown. Future research might explore calibration-free BEV lifting through end-to-end geometric prior learning. Alternatively, advances in foundation models for 3D reconstruction [29, 30, 57] can be used to obtain camera parameters, although our experience in data processing indicates these models might need more effort towards universally reliable for online, reactive robotic manipulation applications. Scaling this architecture to more heterogeneous datasets will further solidify BEV representations as a universal and scalable interface for embodied intelligence.

Acknowledgments

This work was funded by the Key-Area Research and Development Program of Guangdong Province, China under Grant 2024B0101040004, and the Shenzhen Science and Technology Program under Grant KJZD20240903104008012 and ZDCY20250901113000001.

Beyond that, this work was supported by the major leadership and directional guidance of Kui Jia. We sincerely thank all the contributors for their dedication: co-first authors Huayi Zhou and Wei Gao conceptualized the framework and drafted the manuscript, with Huayi Zhou conducting Agilex real-world experiments, and Wei Gao leading the simulation benchmarks, real-world deployment, and core data infrastructure; Dekun Lu and Jian Chen assisted with the data infrastructure and hardware testing; Ruiji Liu, Zhanqi Zhang, and Ziyang Zhang managed the real-robot evaluations on the A1 semi-humanoid and W1 humanoid configurations; Wenlve Zhou, Sheng Xu, and Yongyi Su contributed to text polishing and technical discussions; and Shumin Li, Kangyi Guo, Shichen Xu, and Zixin Huang supported the large-scale real-world teleoperation data collection.

References

[1] 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 (IJRR)*, 44(10-11):1684–1704, 2025.

[2] S. Levine, C. Finn, T. Darrell, and P. Abbeel. End-to-end training of deep visuomotor policies. *The Journal of Machine Learning Research*, 17(1):1334–1373, 2016.

[3] Y. Zhu, Z. Wang, J. Merel, A. Rusu, T. Erez, S. Cabi, S. Tunyasuvunakool, J. Kramár, R. Hadsell, N. de Freitas, et al. Reinforcement and imitation learning for diverse visuomotor skills. *arXiv preprint*, 2018.

[4] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[5] J. Bai, S. Bai, Y. Chu, Z. Cui, K. Dang, X. Deng, Y. Fan, W. Ge, Y. Han, F. Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

[6] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[7] P. Wang, S. Bai, S. Tan, S. Wang, Z. Fan, J. Bai, K. Chen, X. Liu, J. Wang, W. Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

[8] H. Liu, C. Li, Q. Wu, and Y. J. Lee. Visual instruction tuning. *Advances in Neural Information Processing Systems (NeurIPS)*, 36:34892–34916, 2023.

[9] H. Liu, C. Li, Y. Li, and Y. J. Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 26296–26306, 2024.

[10] L. Maes, Q. L. Lidec, D. Scieur, Y. LeCun, and R. Balestriero. Leworldmodel: Stable end-to-end joint-embedding predictive architecture from pixels. *arXiv preprint arXiv:2603.19312*, 2026.

[11] Y. Gao, H. Guo, T. Hoang, W. Huang, L. Jiang, F. Kong, H. Li, J. Li, L. Li, X. Li, et al. Seedance 1.0: Exploring the boundaries of video generation models. *arXiv preprint arXiv:2506.09113*, 2025.

[12] K. Team, J. Chen, Y. Ci, X. Du, Z. Feng, K. Gai, S. Guo, F. Han, J. He, K. He, et al. Kling-omni technical report. *arXiv preprint arXiv:2512.16776*, 2025.

[13] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. P. Foster, P. R. Sanketi, Q. Vuong, et al. Openvla: An open-source vision-language-action model. In *Conference on Robot Learning (CoRL)*, pages 2679–2713. PMLR, 2025.

[14] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter, et al. π_0: A vision-language-action flow model for general robot control. In *Proceedings of Robotics: Science and Systems (RSS)*, LosAngeles, CA, USA, June 2025. doi:10.15607/RSS.2025.XX.010.

[15] S. Liu, L. Wu, B. Li, H. Tan, H. Chen, Z. Wang, K. Xu, H. Su, and J. Zhu. Rdt-1b: a diffusion foundation model for bimanual manipulation. In *The Thirteenth International Conference on Learning Representations (ICLR)*, volume 2025, pages 29982–30009, 2025.

[16] 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. S. Ryoo, G. Salazar, P. R. Sanketi, K. Sayed, J. Singh, S. Sontakke, A. Stone, C. Tan, H. Tran, V. Vanhoucke, S. Vega, Q. H. 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 *Proceedings of Robotics: Science and Systems (RSS)*, Daegu, Republic of Korea, July 2023. doi:10.15607/RSS.2023.XIX.025.

[17] B. Zitkovich, T. Yu, S. Xu, P. Xu, T. Xiao, F. Xia, J. Wu, P. Wohlhart, S. Welker, A. Wahid, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning (CoRL)*, pages 2165–2183. PMLR, 2023.

[18] S. Belkhale, T. Ding, T. Xiao, P. Sermanet, Q. Vuong, J. Tompson, Y. Chebotar, D. Dwibedi, and D. Sadigh. Rt-h: Action hierarchies using language. In *Proceedings of Robotics: Science and Systems (RSS)*, Delft, Netherlands, July 2024. doi:10.15607/RSS.2024.XX.049.

[19] A. Ali, J. Bai, M. Bala, Y. Balaji, A. Blakeman, T. Cai, J. Cao, T. Cao, E. Cha, Y.-W. Chao, et al. World simulation with video foundation models for physical ai. *arXiv preprint arXiv:2511.00062*, 2025.

[20] L. Li, Q. Zhang, Y. Luo, S. Yang, R. Wang, F. Han, M. Yu, Z. Gao, N. Xue, X. Zhu, et al. Causal world modeling for robot control. *arXiv preprint arXiv:2601.21998*, 2026.

[21] S. Gao, W. Liang, K. Zheng, A. Malik, S. Ye, S. Yu, W.-C. Tseng, Y. Dong, K. Mo, C.-H. Lin, et al. Dreamdojo: A generalist robot world model from large-scale human videos. *arXiv preprint arXiv:2602.06949*, 2026.

[22] G. Team, B. Wang, B. Li, C. Ni, G. Huang, G. Zhao, H. Li, J. Li, J. Lv, J. Liu, et al. Gigabrain-0.5 m*: a vla that learns from world model-based reinforcement learning. *arXiv preprint arXiv:2602.12099*, 2026.

[23] H. Zhen, X. Qiu, P. Chen, J. Yang, X. Yan, Y. Du, Y. Hong, and C. Gan. 3d-vla: A 3d vision-language-action generative world model. In *International Conference on Machine Learning (ICML)*, pages 61229–61245. PMLR, 2024.

[24] X. Li, L. Heng, J. Liu, Y. Shen, C. Gu, Z. Liu, H. Chen, N. Han, R. Zhang, H. Tang, et al. 3ds-vla: A 3d spatial-aware vision language action model for robust multi-task manipulation. In *Conference on Robot Learning (CoRL)*, pages 2344–2359. PMLR, 2025.

[25] L. Sun, B. Xie, Y. Liu, H. Shi, T. Wang, and J. Cao. Geovla: Empowering 3d representations in vision-language-action models. *arXiv preprint arXiv:2508.09071*, 2025.

[26] Z. Zhang, H. Li, Y. Dai, Z. Zhu, L. Zhou, C. Liu, D. Wang, F. E. H. Tay, S. Chen, Z. Liu, Y. Liu, X. Li, and P. Zhou. From spatial to actions: Grounding vision-language-action model in spatial foundation priors. In *The Fourteenth International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=fzmittHfq3.

[27] X. Fan, S. Deng, X. Wu, Y. Lu, Z. Li, M. Yan, Y. Zhang, Z. Zhang, H. Wang, and H. Zhao. Any3d-vla: Enhancing vla robustness via diverse point clouds. *arXiv preprint arXiv:2602.00807*, 2026.

[28] B. Liu, Y. Zhu, C. Gao, Y. Feng, Q. Liu, Y. Zhu, and P. Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 36:44776–44791, 2023.

[29] J. Wang, M. Chen, N. Karaev, A. Vedaldi, C. Rupprecht, and D. Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5294–5306, 2025.

[30] H. Lin, S. Chen, J. H. Liew, D. Y. Chen, Z. Li, Y. Zhao, S. Peng, H. Guo, X. Zhou, G. Shi, J. Feng, and B. Kang. Depth anything 3: Recovering the visual space from any views. In *The Fourteenth International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=yirunib8l8.

[31] Y. Liu, T. Wang, X. Zhang, and J. Sun. Petr: Position embedding transformation for multi-view 3d object detection. In *European Conference on Computer Vision (ECCV)*, pages 531–548. Springer, 2022.

[32] Y. Liu, J. Yan, F. Jia, S. Li, A. Gao, T. Wang, and X. Zhang. Petrv2: A unified framework for 3d perception from multi-camera images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3262–3272, 2023.

[33] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Proceedings of Robotics: Science and Systems (RSS)*, Daegu, Republic of Korea, July 2023. doi:10.15607/RSS.2023.XIX.016.

[34] Z. Fu, T. Z. Zhao, and C. Finn. Mobile aloha: Learning bimanual mobile manipulation using low-cost whole-body teleoperation. In *Conference on Robot Learning (CoRL)*, pages 4066–4083. PMLR, 2025.

[35] A. O'Neill, A. Rehman, A. Maddukuri, A. Gupta, A. Padalkar, A. Lee, A. Pooley, A. Gupta, A. Mandlekar, A. Jain, et al. Open x-embodiment: Robotic learning datasets and rt-x models. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 6892–6903. IEEE, 2024.

[36] A. Khazatsky, K. Pertsch, S. Nair, A. Balakrishna, S. Dasari, S. Karamcheti, S. Nasiriany, M. K. Srirama, L. Y. Chen, K. Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. In *Proceedings of Robotics: Science and Systems (RSS)*, Delft, Netherlands, July 2024. doi:10.15607/RSS.2024.XX.120.

[37] K. Wu, C. Hou, J. Liu, Z. Che, X. Ju, Z. Yang, M. Li, Y. Zhao, Z. Xu, G. Yang, S. Fan, X. Wang, F. Liao, Z. Zhao, G. Li, Z. Jin, L. Wang, J. Mao, N. Liu, P. Ren, Q. Zhang, Y. Lyu, M. Liu, H. Jingyang, Y. Luo, Z. Gao, C. Li, C. Gu, Y. Fu, D. Wu, X. Wang, S. Chen, Z. Wang, P. An, S. Qian, S. Zhang, and J. Tang. Robomind: Benchmark on multi-embodiment intelligence normative data for robot manipulation. In *Proceedings of Robotics: Science and Systems (RSS)*, LosAngeles, CA, USA, June 2025. doi:10.15607/RSS.2025.XXI.152.

[38] O. M. Team, D. Ghosh, H. Walke, K. Pertsch, K. Black, O. Mees, S. Dasari, J. Hejna, T. Kreiman, C. Xu, et al. Octo: An open-source generalist robot policy. In *Proceedings of Robotics: Science and Systems (RSS)*, Delft, Netherlands, July 2024. doi:10.15607/RSS.2024.XX.090.

[39] J. Zheng, J. Li, Z. Wang, D. Liu, X. Kang, Y. Feng, Y. Zheng, J. Zou, Y. Chen, J. Zeng, T. Wang, Y.-Q. Zhang, J. Liu, and X. Zhan. X-vla: Soft-prompted transformer as scalable cross-embodiment vision-language-action model. In *The Fourteenth International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=kt51kZH4aG.

[40] K. Black, N. Brown, J. Darpinian, K. Dhabalia, D. Driess, A. Esmail, M. R. Equi, C. Finn, N. Fusai, M. Y. Galliker, et al. $\pi_{0.5}$: a vision-language-action model with open-world generalization. In *Conference on Robot Learning (CoRL)*, pages 17–40. PMLR, 2025.

[41] S. Ye, Y. Ge, K. Zheng, S. Gao, S. Yu, G. Kurian, S. Indupuru, Y. L. Tan, C. Zhu, J. Xiang, et al. World action models are zero-shot policies. *arXiv preprint arXiv:2602.15922*, 2026.

[42] D. Qu, H. Song, Q. Chen, Y. Yao, X. Ye, J. Gu, Z. Wang, Y. Ding, B. Zhao, D. Wang, and X. Li. Spatialvla: Exploring spatial representations for visual-language-action models. In *Proceedings of Robotics: Science and Systems (RSS)*, LosAngeles, CA, USA, June 2025. doi:10.15607/RSS.2025.XX.011.

[43] T. Yuan, Y. Liu, C. Lu, Z. Chen, T. Jiang, and H. Zhao. Depthvla: Enhancing vision-language-action models with depth-aware spatial reasoning. *arXiv preprint arXiv:2510.13375*, 2025.

[44] S. Deng, M. Yan, Y. Zheng, J. Su, W. Zhang, X. Zhao, H. Cui, Z. Zhang, and H. Wang. Stereovla: Enhancing vision-language-action models with stereo vision. *arXiv preprint arXiv:2512.21970*, 2025.

[45] C. Li, J. Wen, Y. Peng, Y. Peng, and Y. Zhu. Pointvla: Injecting the 3d world into vision-language-action models. *IEEE Robotics and Automation Letters (RAL)*, 11(3):2506–2513, 2026.

[46] Q. Yu, X. Yuan, Y. Jiang, J. Chen, D. Zheng, C. Hao, Y. You, Y. Chen, Y. Mu, L. Liu, et al. Artgs: 3d gaussian splatting for interactive visual-physical modeling and manipulation of articulated objects. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 13170–13177. IEEE, 2025.

[47] F. Li, W. Song, H. Zhao, J. Wang, P. Ding, D. Wang, L. ZENG, and H. Li. Spatial forcing: Implicit spatial representation alignment for vision-language-action model. In *The Fourteenth International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=euMVC1DO4k.

[48] P. Li, Y. Chen, H. Wu, X. Ma, X. Wu, Y. Huang, L. Wang, T. Kong, and T. Tan. Bridgevla: Input-output alignment for efficient 3d manipulation learning with vision-language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2025. URL https://openreview.net/forum?id=ffBF6hYuQv.

[49] Y. Chen, S. Liu, X. Shen, and J. Jia. Dsgn: Deep stereo geometry network for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12536–12545, 2020.

[50] C. Reading, A. Harakeh, J. Chae, and S. L. Waslander. Categorical depth distribution network for monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8555–8564, 2021.

[51] Z. Li, W. Wang, H. Li, E. Xie, C. Sima, T. Lu, Y. Qiao, and J. Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In *European Conference on Computer Vision (ECCV)*, pages 1–18. Springer, 2022.

[52] Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, and M. Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations (ICLR)*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

[53] B. Wen, M. Trepte, J. Aribido, J. Kautz, O. Gallo, and S. Birchfield. Foundationstereo: Zero-shot stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5249–5260, 2025.

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

[55] Y. Mu, T. Chen, Z. Chen, S. Peng, Z. Lan, Z. Gao, Z. Liang, Q. Yu, Y. Zou, M. Xu, et al. Robotwin: Dual-arm robot benchmark with generative digital twins. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 27649–27660, 2025.

[56] T. Chen, Z. Chen, B. Chen, Z. Cai, Y. Liu, Z. Li, Q. Liang, X. Lin, Y. Ge, Z. Gu, et al. Robotwin 2.0: A scalable data generator and benchmark with strong domain randomization for robust bimanual robotic manipulation. *arXiv preprint arXiv:2506.18088*, 2025.

[57] J. Wang, M. Chen, S. Zhang, N. Karaev, J. Schönberger, P. Labatut, P. Bojanowski, D. Novotny, A. Vedaldi, and C. Rupprecht. Vggt-Ω. *arXiv preprint arXiv:2605.15195*, 2026.