RoboTron-Mani: All-in-One Multimodal Large Model for Robotic Manipulation

Supplementary Material

6. Related Work

Robotic Datasets. In the early stages of robotics research, it is typically necessary to collect specific datasets for each robot, task, and environment, such as RLBench [27] and CALVIN [43]. Although these datasets are highly customized and of high quality, they are limited in quantity and have poor generalization capabilities. To further enhance model performance and generalization, researchers have collected large amounts of data through teleoperation methods, such as RT-1 [8] and RH20T [20]. These large-scale datasets cover more scenarios and tasks, supporting multi-task learning, but also bring high data annotation costs. As research progresses, methods for integrating multiple datasets, such as Open X-Embodiment [53] and DROID [29], have been proposed to improve model generalization and data utilization efficiency by merging data from different sources. However, these methods also face issues of data inconsistency and potential biases. This paper proposes *RoboData*, which efficiently integrates multiple datasets and unifies the input and output spaces, thereby addressing data heterogeneity. Additionally, it breaks the limitation of training for a single specific task, providing a unified benchmark for robotic manipulation.

Robotic Policies. Previous works such as R3M [48], VC-1 [40], ACT [70], and HULC++ [44] typically employ strategies with a small number of parameters. Subsequent models like RoboFlamingo [35], Corki [26], and RoboUniView [37] have built on multimodal large models but have only fine-tuned on limited datasets. Despite advancements in multi-task learning and few-shot learning, recent models such as RT-X [53], Octo [61], HPT [64], CrossFormer [18], GR-2 [12], and OpenVLA [31] have trained vision-language-action robotic policies on various datasets. However, these works often pre-train on data from real robots [20, 29], human videos [21, 48], and simulation domains [43, 71], neglecting the uniformity of physical space, and achieve good performance only after fine-tuning on specific datasets. Given that robots operate in 3D physical environments, their perception and interaction capabilities must integrate 3D sensing, akin to the requirements of autonomous driving systems.

7. Real world experiment

To evaluate *RoboTron-Mani* performance in real-world scenarios, we constructed a physical evaluation system as illustrated in Figure 8. The robotic platform comprises a Dalu mobile base and an UR3 robotic arm. During experiments, the mobile base remains stationary, and only the robotic arm retaining degrees of freedom. The system is equipped with essential perception and actuation components, including a Robotiq two-finger gripper, an Intel D435 depth camera mounted on the wrist of the arm (denoted as cam_{wrist}), and an Orbbec Gemini Pro depth camera fixed to a stand on the ground to the left of the arm (denoted as cam_{static}).

Since *RoboTron-Mani* requires camera extrinsic parameters, we perform hand-eye calibration prior to experiments. Using the image data and intrinsic/extrinsic parameters from both cam_{wrist} and cam_{static}, we constructed point clouds of the scene, which were used for occupancy (OCC) supervised training to enable the model to learn 3D geometric structures. The system operates on a ROS1-based communication framework to enable efficient interaction between the Nvidia 3090 server and the robotic arm.

In Figure 8, we design ten real-world tasks grouped into three difficulty levels. Easy: pick or push apple, pick banana or Coke bottle. Medium: open drawer, place lid on pot, pour Coke into cup. Hard: open drawer and place object inside, group similar items, store plush toys. We collect 100 teleoperation demonstrations per task (1000 total) for training. For evaluation, each task undergoes 10 trials with manual resets and clear success criteria (e.g., lifting an apple at least 5 cm). More details are provided in the supplementary due to space limits. We compare two baseline models: RoboFlamingo (2D) and RoboUniView (3D), which, like *RoboTron-Mani*, are both 3D models.

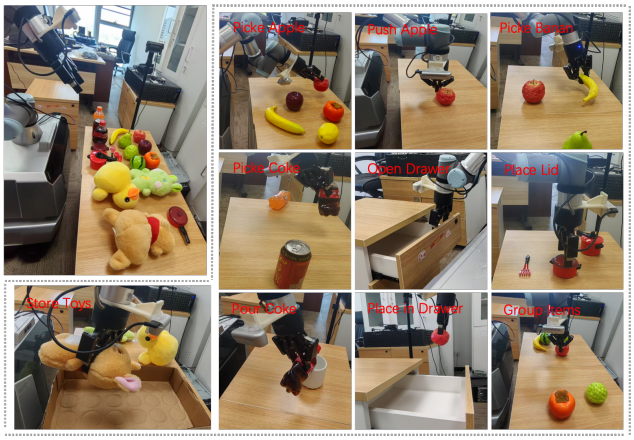

Figure 8. Real-world setup with 10 tasks, featuring a fixed Dalu base, UR3 arm, Robotiq gripper, wrist-mounted Intel D435, and ground-mounted Orbbec Gemini Pro. Hand-eye calibration and point cloud fusion enable OCC-supervised training. The system runs on ROS1 with an NVIDIA RTX 3090 GPU.

In Table 4 Right, *RoboTron-Mani* outperforms baselines across all difficulty levels, especially on medium and hard tasks, demonstrating stronger generalization and

3D reasoning. While RoboUniView performs reasonably on easy tasks, it struggles in more complex scenarios. RoboFlamingo, lacking explicit 3D scene modeling, shows weaker overall performance.

Model	Easy	Medium	Hard
RoboTron-Mani	82.5%	70.0%	46.6%
RoboUniView*	75.0%	36.6%	23.3%
RoboFlamingo*	35.0%	6.6%	6.6%

Table 4. Success rates by task difficulty, * indicates reproduced results.

8. *RoboTron-Mani* detailed information

In summary, the parameters used in the study are as follows: $H = 12$, $N = 3$, $H = 256$, $W = 256$, $L = 80$, $B = 80$, $P = 40$, $C = 1024$, $\lambda_{\text{image}} = 0.1$, $\lambda_{\text{occ}} = 0.1$, $\lambda_g = 0.01$, and $\lambda_{\text{rgb}} = 0.5$. The optimization strategy employs AdamW, while the learning rate schedule utilizes cosine annealing, with an initial learning rate of 10^{-4} and a termination rate of 10^{-6}. The model is trained for a total of 10 epochs unless otherwise specified.

9. Action Representation

Different datasets have different methods for obtaining actions. For example, given the poses at two consecutive time steps $P^t = (p^t, r^t_{quat})$ and $P^{t+1} = (p^{t+1}, r^{t+1}_{quat})$, which are represented by 3D coordinates and quaternions, respectively.

9.1. Euler Angle Difference Method (EADM)

The Euler Angle Difference Method is a way to describe rotational transformations by calculating the difference in Euler angles between two poses (or orientations). The specific steps are as follows:

1. Convert the quaternions r^t_{quat} and r^{t+1}_{quat} to Euler angles r^t_{euler} and r^{t+1}_{euler}, respectively.
2. Compute the differences in the 3D coordinates and Euler angles to obtain the action:

$$A_t = (p^{t+1} - p^t, r^{t+1}_{euler} - r^t_{euler}). \tag{12}$$

This method is intuitive and easy to understand, but it may encounter gimbal lock issues when dealing with large-angle rotations or multiple rotations.

9.2. Composite Rotation Matrix Method (CRMM)

The Composite Rotation Matrix Method describes complex rotational transformations by multiplying multiple rotation matrices. A rotation matrix is a linear algebra tool used to represent rotations in three-dimensional space. The specific steps are as follows:

1. Convert the quaternions r^t_{quat} and r^{t+1}_{quat} to rotation matrices r^t_{matrix} and r^{t+1}_{matrix}, respectively.
2. Compute the composite rotation by multiplying the rotation matrices to obtain the action:

$$A_t = (p^{t+1} - p^t, r^{t+1}_{matrix} \cdot Inv(r^t_{matrix})) \tag{13}$$

This method is advantageous because it can conveniently handle any complex combination of rotations and avoids the gimbal lock problem.

9.3. Pose Composition Method (PCM)

The pose composition method is a way to describe the position and orientation of an object in space. By combining the poses at two consecutive time steps, complex motions can be described. The specific steps are as follows:

1. Convert the quaternions r^t_{quat} and r^{t+1}_{quat} to rotation matrices r^t_{matrix} and r^{t+1}_{matrix}, respectively.
2. Combine the poses to obtain the action:

$$A_t = \left(Inv(R^t_{matrix}) \cdot (p^{t+1} - p^t), Inv(R^t_{matrix}) \cdot R^{t+1}_{matrix}\right) \tag{14}$$

This method is advantageous because it can conveniently describe and compute complex motions of objects in space and is widely used in robotics and computer vision.

10. *RoboData* detailed information

10.1. CALVIN Dataset

CALVIN is an open-source simulated benchmark specifically designed for learning long-horizon language-conditioned tasks in robotics. The dataset features four distinct environment splits, labeled A, B, C, and D. Each environment contains 6 hours of human-teleoperated recording data, resulting in over 2 million trajectories. However, only 1% of this data is annotated with language instructions, amounting to approximately 24,000 trajectories. Each environment split is uniquely configured with various objects and scenarios, allowing for comprehensive validation of the performance, robustness, and generality of the trained policies across different data combinations.

The benchmark utilizes a 7-degree-of-freedom (7-DOF) Franka Emika Panda robotic arm equipped with a parallel gripper. This robotic platform is enhanced with onboard sensors and captures images from two camera perspectives, enabling it to effectively execute complex sequences of language instructions. The coordinate system is based on the robot's body, represented as Right-Forward-Up, where the X-axis represents the right direction, the Y-axis denotes the forward direction, and the Z-axis indicates the upward direction.

For action representation, CALVIN employs EADM. To ensure that actions are appropriately scaled for network predictions, specific scaling factors are applied: 0.02 for the X,

Platform	Physics Engine	Robot	Coordinate (X-Y-Z)	Views	Camera Parameters	Action[a] Representation	Tasks	Episodes
CALVIN [43]	PyBullet	7-DOF Franka	Right-Forward-Up	Static, Gripper	No	EADM	34	20K
Meta-World [68]	MuJoCo	4-DOF Sawyer	Right-Forward-Up	behindGripper, corner, corner2, corner3, topview, gripperPOV	No	None	50	5K
Libero [36]	MuJoCo	7-DOF Franka	Forward-Left-Up	frontview, birdview, agentview, sideview	No	CRMM	130	6.5K
RoboMimic [41]	MuJoCo	7-DOF Franka	Forward-Left-Up	agentview, robot0_eye_in_hand	No	CRMM	8	1.6K
RoboCasa [50]	MuJoCo	12-DOF Franka	Forward-Left-Up	center, left, right, frontview, eye_in_hand	No	CRMM	100	5K
ManiSkill2 [22]	SAPIEN	7-DOF Franka	Forward-Right-Down	base_camera, hand_camera	No	PCM	20	30K
RoboCAS [71]	SAPIEN /Isaac	7-DOF Franka	Forward-Left-Up	gripper_camera, base_camera, static_camera	Yes	Absolute	3	7.3K
RLBench [27]	V-REP	7-DOF Franka	Forward-Left-Up	left_shoulder, right_shoulder, wrist, front	Yes	Absolute	18	1.8K
Colosseum [55]	PyRep	7-DOF Franka	Forward-Left-Up	left_shoulder, right_shoulder, wrist, front	Yes	Absolute	20	2K

Platform	Workspace $\mathbf{Min}_{[X,Y,Z]}, \mathbf{Max}_{[X,Y,Z]}$	Action Space $\mathbf{Min}_{[X,Y,Z,Pitch,Roll,Yaw]}, \mathbf{Max}_{[X,Y,Z,Pitch,Roll,Yaw]}$	Gripper (Open/Close)
CALVIN [43]	[-0.43, -0.57, 0.43], [0.37, -0.00, 0.80]	[-0.03, -0.03, -0.03, -6.28, -0.07, -6.27], [0.04, 0.02, 0.02, 6.28, 0.06, 6.28]	-1/1
Meta-World [68]	[-0.50, -0.10, 0.12], [0.48, 0.41, 0.60]	[-1.00, -1.00, -1.00], [1.00, 1.00, 1.00]	0.5/-0.5
Libero [36]	[-0.24, -0.43, 0.01], [0.86, 0.57, 0.90]	[-0.93, -0.93, -0.93, -0.33, -0.37, -0.37], [0.93, 0.93, 0.93, 0.37, 0.37, 0.37]	1/-1
RoboMimic [41]	[-0.17, -0.40, 0.90], [0.33, 0.33, 1.29]	[-1.0, -1.0, -1.0, -0.55, -1.0, -1.0], [1.0, 1.0, 1.0, 0.72, 0.45, 1.0]	1/-1
RoboCasa [50]	[-0.81, -1.35, 0.70], [0.85, 0.75, 1.83]	[-1.0, -1.0, -1.0, -1.0, -1.0, -1.0], [1.0, 1.0, 1.0, 1.0, 1.0, 0.89]	1/-1
ManiSkill2 [22]	[-0.26, -0.79, -1.17], [0.85, 0.76, 0.00]	[-0.14, -0.15, -0.16, -0.09, -0.09, -0.09,], [0.17, 0.16, 0.15, 0.09, 0.09, 0.09]	-1/1
RoboCAS [71]	[-0.70, -0.82, 0.062], [0.85, 0.67, 0.92]	[-0.04, -0.04, -0.04, -0.12, -0.10, 0.15], [0.03, 0.04, 0.03, 0.07, 0.09, 0.16, 0.08]	0/0.08
RLBench [27]	[-0.89, -0.72, 0.80], [0.56, 0.69, 1.89]	[-0.05, -0.04, -0.03, -1.0, -0.15, -1.0], [0.05, 0.06, 0.03, 1.0, 0.15, 1.0]	0/1
Colosseum [55]	[-0.68, -0.68, 0.83], [0.54, 0.70, 1.85]	[-0.04, -0.04, -0.04, -0.79, -0.12, -0.76], [0.03, 0.03, 0.04, 0.79, 0.35, 0.79]	0/1

Table 5. Detailed information of CALVIN [43], Meta-World [68], LIBERO [36], Robomimic [41], RoboCAS [71], ManiSkill2 [22], RoboCasa [49], RLBench [27], and Colosseum [55].

Y, and Z axes, and 0.05 for the pitch, roll, and yaw angles. The states of the gripper are represented using -1 for open and 1 for closed, facilitating clear action commands.

Space Alignment: *RoboData* includes all 34 distinct tasks, providing 20,000 episodes with language instructions for training. Action representations are regenerated using CRMM, and camera parameters are obtained through replay. Since the other input spaces are consistent with those predefined by *RoboData*, no alignment adjustments are necessary.

The dataset evaluates 1,000 unique instruction chains, focusing primarily on sequential task execution. In each experiment, the robotic agent successfully completes a series of up to five language instructions in succession. The agent can only proceed to the next instruction after successfully achieving the current task, establishing a clear dependency on the completion of prior actions.

10.2. Meta-World Dataset

Meta-World is a tabletop manipulation benchmark designed to facilitate the training and evaluation of robotic manipulation policies in a simulated environment. This dataset focuses on the reinforcement learning domain and does not release training data. The simulator includes six perspectives: behindGripper, corner, corner2, corner3, topview, gripperPOV.

The benchmark utilizes a 4-degree-of-freedom (4-DOF) Franka Emika Panda robotic arm equipped with a parallel gripper, which does not allow end rotation. The gripper states are represented by the numbers 0.5/-0.5 for open/close, and the coordinate system is consistent with that of the CALVIN dataset.

Space Alignment: *RoboData* includes the ML-45 version, which consists of 45 distinct tasks. To address the lack of training data for simulation learning, we adopt the scripted policies from Yu et al. [68] and introduce Gaussian noise $N(0, 0.1)$ to the generated actions at each step, resulting in a total of 22,500 trajectories, with each task producing 500 successful trajectories. To align with *RoboData*'s predefined settings, we extract observations from the corner2 and gripperPOV perspectives. The rotational variables in the actions are zero-padded, and the gripper states are represented using -1 for open and 1 for closed, while other parameters remain unchanged.

For performance evaluation, we test on 20 unseen start and goal configurations for each task, totaling 900 unseen configurations. We report the average performance over these 900 trajectories, providing a comprehensive measure of the model's ability to generalize to new tasks and configurations.

10.3. LIBERO Dataset

LIBERO is a lifelong learning benchmark that includes multiple task suites involving various language-labeled rigid and articulated-body manipulation tasks. The dataset consists of a total of 130 tasks and 6,500 trajectories. The simulator includes four perspectives: frontview, birdview, agentview, sideview, all with a resolution of 256 × 256 pixels. The action representation differs from that used in CALVIN, employing CRMM to define actions.

Space Alignment: *RoboData* includes the LIBERO-90 suite, which consists of 90 manipulation tasks, each with 50 demonstration trajectories collected through human teleoperation, providing a rich set of examples for training and evaluation. We select frontview and birdview as the observation perspectives, and camera parameters are obtained through replay. The coordinate system is defined as Forward-Left-Up. Due to differences in the coordinate system and workspace compared to the predefined settings in *RoboData*, we align them through rotation and translation:

$$W_{LIBERO} = \begin{bmatrix} 0 & 1 & 0 & 0.3 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -0.1 \\ 0 & 0 & 0 & 1 \end{bmatrix} W^{ori}_{LIBERO}.$$

The states of the gripper are similarly represented using -1 for open and 1 for closed.

During evaluation, we test on 20 unseen start and goal configurations for each task, totaling 1,800 unseen configurations. This approach allows for a comprehensive assessment of the agent's performance and generalization capabilities, ensuring that the evaluation reflects the agent's ability to adapt to new situations and previously unencountered scenarios.

10.4. RoboMimic Dataset

RoboMimic is a large-scale robotic manipulation benchmark designed to study imitation learning and offline reinforcement learning. The dataset includes 5 distinct manipulation tasks, each with a dataset of demonstrations teleoperated by proficient humans. These tasks are designed to enhance the learning effectiveness of robots through real human demonstrations.

Space Alignment: *RoboData* includes 3 of these tasks (Lift, Can, Square) and excludes the other two dual-arm tasks. Given that the characteristics of RoboMimic align with those of LIBERO, all alignment methods can refer to LIBERO.

During evaluation, we test on 20 unseen start and goal configurations for each task, totaling 600 unseen configurations.

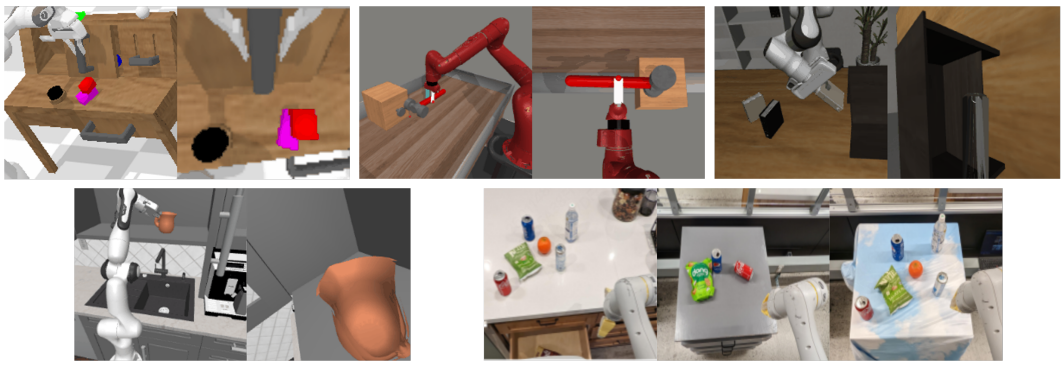

Figure 9. Evaluation Datasets. We evaluate *RoboTron-Mani* across five simulation benchmarks and present policy rollout visualizations of the experiments. From left to right, the benchmarks include CALVIN, Meta-World, LIBERO, RoboCasa, and Robomimic. Experiment details can be found in Section 4.1.

10.5. RoboCasa Dataset

RoboCasa is an open-source simulation benchmark designed to study robotic manipulation tasks in household environments, utilizing a 12-DOF Franka robot, where the first 7 degrees of freedom are related to manipulation and the remaining 5 are related to mobility. The dataset includes a simulation environment featuring 120 distinct real-world scenes, thousands of interactive objects, and household appliances, utilizing generative AI tools to create environmental textures and 3D objects. The RoboCasa dataset introduces 100 systematic evaluation tasks, consisting of 25 atomic tasks and 75 composite tasks generated with the guidance of large language models. Additionally, RoboCasa provides a large-scale multi-task dataset containing over 100,000 trajectories for model training, showcasing performance improvements achieved through behavior cloning training with synthetic data, as well as the applicability of simulation data in real-world tasks. These features make RoboCasa an important resource for researching and developing language-conditioned robotic technologies, laying a solid foundation for advancing intelligent applications of robots in household environments.

Space Alignment: *RoboData* includes 5,000 samples collected through remote control, utilizing two perspectives: front view and eye-in-hand. Only the degrees of freedom related to manipulation are retained. Given that the characteristics of RoboCasa align with those of LIBERO, all alignment methods can refer to LIBERO.

During evaluation, we test on 20 unseen start and goal configurations for each task, totaling 2,000 unseen configurations.

10.6. ManiSkill2 Dataset

ManiSkill2 is a unified benchmark designed for learning generalizable robotic manipulation skills, built on the SAPIEN platform. It includes 20 out-of-the-box task families, featuring over 2,000 distinct object models and more than 4 million demonstration frames. The dataset supports fast visual input learning algorithms, enabling a CNN-based policy to collect samples at approximately 2,000 frames per second (FPS) using just one GPU and 16 processes on a workstation. As the next generation of the SAPIEN ManiSkill benchmark, ManiSkill2 addresses critical pain points often encountered by researchers when utilizing benchmarks for developing generalizable manipulation skills, covering various task types, including stationary/mobile bases, single/dual-arm, and rigid/soft-body manipulation tasks. This extensive diversity of tasks and objects aims to enhance the robustness and applicability of robotic manipulation algorithms in real-world scenarios, making it an essential resource for advancing research in the field.

Space Alignment: *RoboData* includes 20 tasks related to single-arm manipulation from the ManiSkill2 dataset. The coordinate system and workspace are defined as Forward-Right-Down and [-0.26, -0.79, -1.17] to [0.85, 0.76, 0.00]. To ensure spatial consistency and compatibility, the corresponding coordinate transformations are applied:

$$W_{ManiSkill2} = \begin{bmatrix} 0 & 1 & 0 & 0.3 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} W^{ori}_{ManiSkill2}.$$

The action representation uses CRMM to replace PCM. Since the original data did not include the camera's intrinsic and extrinsic parameters, we replayed the data and saved the relevant parameters.

During evaluation, we test on 20 unseen start and goal configurations for each task, totaling 400 unseen configurations.

10.7. RoboCAS Dataset

RoboCAS is a benchmark proposed by Meituan's Embodied Intelligence Team, specifically designed for complex object arrangement scenarios in robotic manipulation. It is the first benchmark of its kind for such tasks and the first to employ a flexible and concise scripting strategy to collect samples in a cost-effective and efficient manner. RoboCAS showcases the handling of dispersed, ordered, and stacked objects within a highly realistic physical simulation environment, aiming to enhance robots' operational capabilities and performance across diverse settings. The benchmark provides a variety of proprioceptive observations and visual data, including RGB images and depth maps captured from the left gripper camera, base camera, and static camera.

Space Alignment: *RoboData* includes all samples, utilizing only the base camera and static camera. The coordinate system and workspace are defined as Forward-Left-Up and [-0.70, -0.82, 0.062] to [0.85, 0.67, 0.92]. To ensure spatial consistency and compatibility, the following coordinate transformation is applied:

$$W_{RLBench} = \begin{bmatrix} 0 & 1 & 0 & 0.3 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0.7 \\ 0 & 0 & 0 & 1 \end{bmatrix} W^{ori}_{RLBench}.$$

Since only end-effector positions are provided in the dataset, the research team utilized a composite rotation matrix to generate corresponding action representations, changing the gripper's open/close state from 0/0.04 to -1/1. Notably, the RGB images from this perspective are 480x640 pixels; to maintain consistency across all data in *RoboData*, we only extract the central region of 480x480 pixels.

During evaluation, we test on 20 unseen start and goal configurations for each task.

10.8. RLBench Dataset

RLBench is a challenging benchmark and learning environment specifically designed for robot learning. This benchmark features 18 completely unique, hand-designed tasks that range in difficulty from simple target reaching and door opening to more complex multi-stage tasks, such as opening an oven and placing a tray inside. RLBench provides a variety of proprioceptive observations and visual observation data, including RGB images, depth maps, and segmentation masks from the left shoulder, right shoulder, wrist, and front views.

Space Alignment: *RoboData* includes all experiments, totaling 1.8 experiments, with visual input extracted from the wrist and front views. The coordinate system in this dataset differs from that of other datasets, defined as Forward-Left-up, with a workspace range from [-0.89, -0.72, 0.80] to [0.56, 0.69, 1.89]. We apply spatial transformations to shift the data into a predefined coordinate system.

$$W_{RLBench} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0.7 \\ 0 & 0 & 0 & 1 \end{bmatrix} W^{ori}_{RLBench}.$$

Additionally, only end-effector positions are provided, so we use CRMM to transform action representations, changing the gripper's open/close state from 0/1 to -1/1.

During evaluation, we test on 20 unseen start and goal configurations for each task, totaling 360 unseen configurations.

10.9. Colosseum Dataset

Colosseum is a benchmark that complements RLBench by addressing the limitations of single environmental variables. It features 20 diverse manipulation tasks that enable systematic evaluation of models across 14 axes of environmental perturbations. These perturbations include changes in the color, texture, and size of objects, as well as variations in tabletop surfaces, backgrounds, and the physical properties of objects. Additionally, lighting conditions, distractors, and camera poses are adjusted. All configurations align with those of RLBench, allowing researchers to test and compare the robustness and adaptability of their models under a wider range of environmental conditions.

Space Alignment: *RoboData* includes all samples, and the alignment method is consistent with that of RLBench.

11. Comparison with OpenVLA

To evaluate the superiority of our model architecture, we compare *RoboTron-Mani* with the currently best-performing OpenVLA. To ensure a fair comparison, we set the window size to 1, and train *RoboTron-Mani* from scratch, while OpenVLA is fine-tuned on the officially released weights.

As shown in Table 6, *RoboTron-Mani* outperforms OpenVLA (LoRA) in multiple metrics, particularly in Task 3-5 and average sequence length. This indicates that *RoboTron-Mani* can capture longer dependencies when handling tasks, thereby improving model accuracy. These results not only demonstrate the superior performance of the *RoboTron-Mani* architecture but also provide valuable references for future research.

	Task Completed in a Sequence					Avg Len
	1	2	3	4	5	
OpenVLA (LoRA)	78%	55%	29%	17%	8%	1.86
RoboTron-Mani(ours)	81%	54%	37%	25%	16%	2.15

Table 6. Performance comparison with OpenVLA on CALVIN.

12. Experiment Details

The success rates of the expert models in the Table 1 are organized from the following sources: The evaluation methods on the CALVIN [43] dataset are sourced from the official CALVIN leaderboard (url: http://calvin.cs.uni-freiburg.de/). In the Meta-World [68] dataset, the results of PAD [24], GR-1 [67], SuSIE [7], RT-2* [9], and RT-1 [8] come from PAD [24], while the results of PRISE [72] are derived from related papers. In the Libero [36] dataset, the results of QueST [47], VQ-BeT [32], PRISE [72], DiffusionPolicy [16], ACT [70], and ResNet-T [36] all come from QueST [47], while the results of MDT [58] and Distill-D [25] are sourced from MDT [58]; the results of MaIL [28], ATM [66], and MUTEX [59] come from their respective research papers. The results of RoboCasa [50] in the RoboCasa [50] dataset are sourced from related papers. In the Rt-1 [8] dataset, all results come from CogACT [34] paper.

The success rates of each task for *RoboData* on various datasets are shown in the Table 8, 9, 10, 11, 7.

Task Name	Success Rate
CoffeePressButton	100%
CoffeeServeMug	50%
CoffeeSetupMug	25%
CloseDoubleDoor	30%
CloseSingleDoor	100%
OpenDoubleDoor	0%
OpenSingleDoor	45%
CloseDrawer	100%
OpenDrawer	80%
TurnOffMicrowave	75%
TurnOnMicrowave	85%
PnPCabToCounter	45%
PnPCounterToCab	50%
PnPCounterToMicrowave	35%
PnPCounterToSink	40%
PnPCounterToStove	60%
PnPMicrowaveToCounter	70%
PnPSinkToCounter	40%
PnPStoveToCounter	40%
TurnOffSinkFaucet	70%
TurnOnSinkFaucet	55%
TurnSinkSpout	90%
TurnOffStove	35%
TurnOnStove	55%
PrepareCoffee	0%
ArrangeVegetables	0%
MicrowaveThawing	0%
RestockPantry	0%
PreSoakPan	0%

Table 7. *RoboTron-Mani* Success Rates on Various Tasks in RoboCasa [50].

Task	Success Rate
rotate_blue_block_right	82.6%
move_slider_right	98.2%
lift_red_block_slider	87.9%
place_in_slider	26.9%
turn_off_lightbulb	89.7%
turn_off_led	96.9%
push_into_drawer	68.1%
lift_blue_block_drawer	100.0%
lift_pink_block_slider	86.2%
open_drawer	93.1%
lift_pink_block_table	87.0%
turn_on_lightbulb	94.4%
rotate_blue_block_left	96.6%
push_blue_block_left	87.1%
close_drawer	91.0%
rotate_red_block_right	80.3%
push_pink_block_right	64.9%
push_red_block_right	59.7%
push_red_block_left	87.1%
lift_blue_block_table	87.2%
place_in_drawer	97.3%
move_slider_left	91.9%
rotate_red_block_left	84.6%
turn_on_led	93.0%
lift_red_block_table	93.0%
stack_block	55.4%
push_pink_block_left	91.2%
lift_blue_block_slider	85.2%
unstack_block	100.0%
rotate_pink_block_left	90.2%
lift_pink_block_drawer	85.7%
rotate_pink_block_right	63.5%
lift_red_block_drawer	93.3%
push_blue_block_right	48.4%

Table 8. *RoboTron-Mani* Success Rates on Various Tasks in CALVIN [43].

Task	Success Rate
assembly-v2	100%
basketball-v2	100%
bin-picking-v2	70%
box-close-v2	85%
button-press-topdown-v2	100%
button-press-topdown-wall-v2	100%
button-press-v2	80%
button-press-wall-v2	85%
coffee-button-v2	90%
coffee-pull-v2	40%
coffee-push-v2	65%
dial-turn-v2	100%
disassemble-v2	80%
door-close-v2	100%
door-lock-v2	100%
door-open-v2	100%
door-unlock-v2	100%
hand-insert-v2	55%
drawer-close-v2	100%
drawer-open-v2	100%
faucet-open-v2	0%
faucet-close-v2	90%
hammer-v2	15%
handle-press-side-v2	100%
handle-press-v2	100%
handle-pull-side-v2	25%
handle-pull-v2	100%
lever-pull-v2	80%
peg-insert-side-v2	55%
pick-place-wall-v2	95%
pick-out-of-hole-v2	15%
reach-v2	75%
push-back-v2	100%
push-v2	90%
pick-place-v2	100%
plate-slide-v2	100%
plate-slide-side-v2	100%
plate-slide-back-v2	100%
plate-slide-back-side-v2	100%
peg-unplug-side-v2	25%
soccer-v2	20%
stick-push-v2	100%
stick-pull-v2	85%
push-wall-v2	100%
reach-wall-v2	85%
shelf-place-v2	45%
sweep-into-v2	95%
sweep-v2	100%
window-open-v2	60%
window-close-v2	100%

Table 9. *RoboTron-Mani* Success Rates on Various Tasks in Meta-World [68]

Method	Pick Coke Can	Move Near	Open / Close Drawer	Overall
RT-1-X	0.567	0.317	0.597	0.534
RT-2-X(55B)	0.787	0.779	0.250	0.607
Octo-Base	0.170	0.042	0.227	0.169
OpenVLA	0.163	0.462	0.356	0.248
HPT[1]	0.60	0.24	0.23	0.35
RoboTron-Mani	0.63	0.64	0.525	0.60

Table 10. SIMPLER evaluation results of different methods on RT-1. The "Overall" column reports the success rate averaged across the sub-tasks of all task types.

Task Index	Success Rate	Task Index	Success Rate
0	100%	45	80%
1	85%	46	90%
2	95%	47	100%
3	95%	48	95%
4	85%	49	95%
5	60%	50	95%
6	100%	51	35%
7	100%	52	95%
8	90%	53	100%
9	80%	54	95%
10	100%	55	100%
11	95%	56	100%
12	90%	57	100%
13	75%	58	100%
14	95%	59	90%
15	95%	60	100%
16	100%	61	95%
17	95%	62	95%
18	95%	63	100%
19	100%	64	80%
20	100%	65	100%
21	85%	66	100%
22	90%	67	100%
23	70%	68	95%
24	100%	69	95%
25	100%	70	100%
26	90%	71	100%
27	75%	72	95%
28	100%	73	70%
29	100%	74	90%
30	100%	75	70%
31	100%	76	100%
32	50%	77	100%
33	85%	78	85%
34	100%	79	100%
35	85%	80	90%
36	90%	81	60%
37	100%	82	100%
38	80%	83	95%
39	85%	84	90%
40	95%	85	95%
41	100%	86	90%
42	100%	87	100%
43	100%	88	100%
44	95%	89	95%

Table 11. *RoboTron-Mani* Success Rates on Various Tasks in Libero [36].