LOOK AHEAD TOGETHER, ACT LOCALLY: LEARNING A SHARED PREDICTIVE BRAIN FOR MULTI-ROBOT MANIPULATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-robot manipulation requires robots to coordinate their actions while reliably performing their individual operations. Existing approaches often face a trade-off between decentralized policies that preserve per-robot controllers but lack global coordination context, and centralized policies that model team interactions but tightly couple coordination with joint action generation. In this paper, we introduce a multi-robot manipulation framework built on the principle of *Look Ahead Together, Act Locally*. Our method separates coordination from control through a shared predictive *Brain* and individual *Action Experts*. Given all robots' observations and a language instruction, the Brain learns a shared prediction of the collaborative future through alignment with future multi-view 3D features, and derives corresponding guidance for each Action Expert. The frozen Brain then guides independently adapted single-robot policies, enabling coordinated action while preserving their modularity. Experiments on RoboFactory demonstrate an 85.5% average success rate across six tasks, consistently outperforming decentralized and centralized baselines, while preserving a modular interface that enables individual experts to be replaced without retraining the Brain or other controllers.

1 INTRODUCTION

Recent advances in robot learning have enabled increasingly capable single-robot manipulation policies (Kim et al., 2024; Team et al., 2024; Black et al., 2024). Many tasks, however, can benefit from multiple robots acting across a larger workspace, jointly supporting shared objects, or executing independent operations in parallel (Lai et al., 2025; Liu et al., 2025; Jiang et al., 2023; Qin et al., 2025). For example, positioning a large panel during assembly may require robots to support and control it from different locations (Chen et al., 2026; Wang & Schwager, 2016; Nunziante et al., 2025). Realizing these benefits requires more than capable individual manipulation. The robots must also coordinate where and when they act so that their operations remain spatially and temporally compatible (Qin et al., 2025; He et al., 2025; Dong et al., 2025; Park et al., 2026; Chen et al., 2026). Even when each robot can perform its operation reliably, the team may fail if these operations occur in the wrong order or at incompatible positions (Qin et al., 2025; Chen et al., 2026).

A natural way to extend single-robot capabilities to a robot team is to deploy a separate policy for each robot, with each policy using its local observations to predict its own actions. This *decentralized* design (Figure 1a) retains a per-robot observation–action interface, making it easier to reuse pretrained single-robot policies and adapt them to different robot roles (Doshi et al., 2026). However, local observations may not reveal a teammate's progress or changes to shared objects, making it difficult for each robot to determine when and how to act (Park et al., 2026; Chen et al., 2026). Concurrent work CLS-DP (Park et al., 2026) addresses this challenge by learning a locally inferred coordination latent supervised by future team kinematics. Its inference nevertheless remains decentralized, requiring each robot to infer the team's future behavior from its own partial observations. When relevant teammate states or shared-object changes fall outside this local view, these predictions may not reflect the team's actual progress. A robot may therefore begin its next operation while its teammate is still completing the preceding one.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

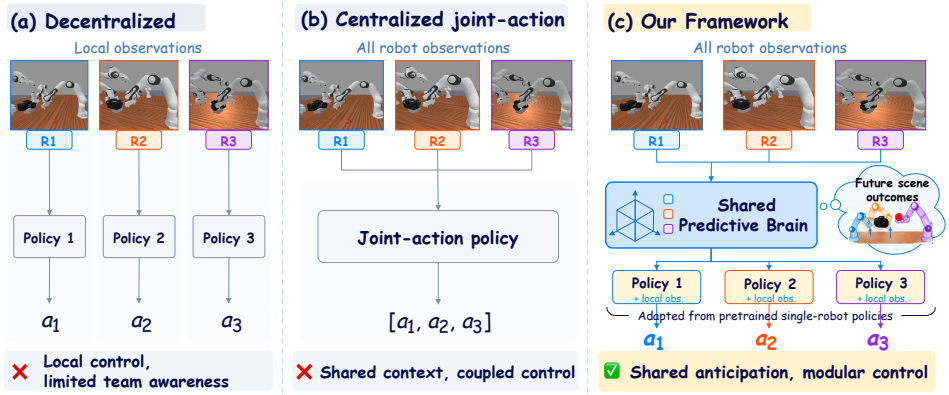

Figure 1: **Illustration of different coordination paradigms for multi-robot manipulation.** Decentralized policies in (a) retain independent per-robot control but have limited team awareness due to local observations, while centralized joint-action policies in (b) exploit shared observations at the cost of coupled control. Our framework in (c) combines shared anticipation with modular control through a predictive Brain and independently adaptable single-robot policies.

Aggregating observations across robots can reveal task-relevant information missing from an individual robot's view. Centralized joint-action policies (Figure 1b) use these shared observations to model dependencies across robot motions (Wang et al., 2026; Fu et al., 2024). GauDP (Wang et al., 2026) first reconstructs a shared 3D representation of the current scene from all robots' views, and a single policy then uses this representation to jointly predict the actions of the entire team. However, these joint-action policies couple coordination with individual action generation within a single model, making it less straightforward to retain pretrained single-robot policies as independently adaptable controllers (Im et al., 2026; Lu et al., 2025). In particular, they do not expose a separate action model for each robot that can be replaced while keeping the other controllers fixed. Moreover, GauDP's reconstruction objective grounds its shared representation in the current scene without explicitly supervising future scene outcomes involving robots and shared objects (Wang et al., 2026). When two robots carry a panel, for example, anticipating its future position and orientation can guide both robots toward compatible motions. These considerations motivate a shared predictive model that integrates observations across robots to provide robot-specific guidance, while allowing each controller to be adapted independently.

In this paper, we propose a multi-robot manipulation framework built on the principle of *Look Ahead Together, Act Locally*. A shared predictive Brain anticipates collaborative scene outcomes, while individual action experts generate each robot's actions (Figure 1c). Given all robots' current RGB observations and the language instruction, the Brain learns solely through future 3D feature alignment. For each robot, we jointly use representations from multiple Brain depths to predict features of its future view extracted by a frozen 3D foundation model (Wang et al., 2025). These targets provide supervision for anticipating scene outcomes involving both robot configurations and changes to shared objects. Future observations and the 3D foundation model are used only during Brain training. We then freeze the Brain and adapt a pretrained single-robot policy for each robot, conditioning its action generation on the corresponding Brain guidance and local observations. This modular design also allows an individual expert to be replaced by adapting a new action model to the same guidance, without retraining the Brain or the other experts.

Our main contributions are summarized as follows.

- We introduce a multi-robot manipulation framework that separates centralized multi-view anticipation by a shared predictive Brain from decentralized action generation by individual robot policies.
- We develop multi-depth future 3D feature alignment to train the Brain to anticipate collaborative scene outcomes involving both robots and shared objects.
- We adapt pretrained single-robot policies to collaborative tasks by training each policy to use predictive guidance from a frozen Brain.

2 Related Work

Generalist Robot Learning. Generalist robot policies such as Octo (Team et al., 2024), OpenVLA (Kim et al., 2024), and π_0 (Black et al., 2024) learn transferable manipulation capabilities from demonstrations spanning diverse tasks and robot embodiments. Their pretrained visual representations and action models can be adapted to new tasks and configurations with less task-specific training (Team et al., 2024). This transferability makes them a natural foundation for multi-robot systems, where pretrained single-robot policies can be reused as separate controllers for different robots (Doshi et al., 2026). Our method builds on this modularity by retaining independently adaptable single-robot policies while augmenting them with predictive guidance from a shared Brain.

Multi-Robot Collaborative Manipulation. Existing multi-robot manipulation policies differ in how coordination information is shared and how robot actions are generated (Qin et al., 2025; Doshi et al., 2026; Park et al., 2026; Wang et al., 2026). *Decentralized* approaches preserve separate per-robot controllers, making individual policies easier to reuse and adapt, but each robot must coordinate from locally available information. CHORUS (Doshi et al., 2026) adapts a shared pretrained VLA using per-robot training examples and deploys an independent copy on each robot, conditioned on local observations and a robot-identifying prompt. Concurrent work CLS-DP (Park et al., 2026) improves such local coordination by distilling future team kinematics into coordination latents, but each robot still infers these latents from its own observation and the shared instruction. In contrast, *centralized* approaches can directly exploit observations across the team, but typically couple coordination with joint action generation. GauDP (Wang et al., 2026), for example, reconstructs shared 3D Gaussian context from all robots' views and uses the resulting representation to jointly predict actions for the entire team. Our method combines global context with modular action generation through a shared predictive Brain that integrates observations from all robots to produce robot-specific predictive guidance, while separate robot experts generate actions and can be adapted independently.

Predictive and Geometric Representations for Robot Control. Prior work improves robot control by incorporating predictive supervision or geometric priors into policy representations. For future-aware control, CoT-VLA (Zhao et al., 2025) and ThinkAct (Huang et al., 2026b;a) guide action generation with visual subgoals and latent plans, respectively, while WorldVLA (Cen et al., 2025) learns action-conditioned future image prediction. FLARE (Zheng et al., 2025) instead aligns policy features with embeddings of future observations. Complementary work injects geometric structure into policy representations using 3D models. Spatial Forcing (Li et al., 2026) supervises policy features with VGGT (Wang et al., 2025) representations, while ROCKET (Sun et al., 2026) extends such alignment across multiple network depths. These methods improve a single policy's representation from its own observations, and applying them independently to multiple robots would still require each robot to infer future scene evolution from a partial view. Our method instead trains a shared Brain from all robots' observations using multi-depth future 3D feature alignment, allowing each robot policy to receive predictive guidance for its own view informed by the entire team.

3 Method

3.1 Problem Formulation

Consider N robots operating in a shared workspace under a language instruction l. At time t, each robot $i \in \{1, \dots, N\}$ receives an RGB observation $o_{i,t}$ and its proprioceptive state $s_{i,t}$, with $O_t = (o_{1,t}, \dots, o_{N,t})$ denoting the multi-robot RGB observations. Our goal is to generate an action chunk $a_{i,t}$ for each robot so that the robots jointly accomplish the task specified by l. We omit within-chunk indices for simplicity. To address this task, we propose a framework comprising a shared predictive Brain and a separate action Expert for each robot (Figure 2). The Brain jointly processes all robots' observations O_t and the instruction l, producing robot-specific predictive guidance $G_{i,t}$ for each robot. Each Expert combines this guidance with its local observation $o_{i,t}$, proprioceptive state $s_{i,t}$, and the instruction l to generate an action chunk $a_{i,t}$. We train our framework in two stages. In the first stage, we learn the Brain through future 3D feature alignment (Section 3.2). In the second stage, we initialize each Expert from a pretrained single-robot policy and adapt it to use

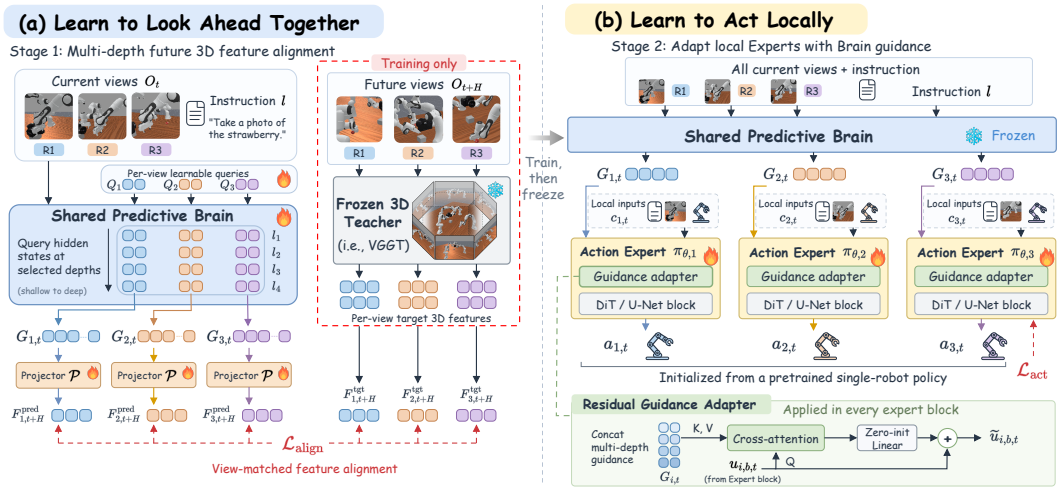

Figure 2: **Overview of our method.** **(a)** The shared predictive Brain maps all robots' current observations O_t and instruction l to robot-specific guidance $G_{i,t}$, trained by aligning multi-depth hidden representations with future 3D features from a frozen teacher. **(b)** Each Action Expert is initialized from a pretrained single-robot policy and incorporates its Brain guidance through residual adapters, while the Brain remains frozen.

the frozen Brain guidance (Section 3.3). Section 3.4 describes the learning strategy and inference procedure.

3.2 Learning the Brain through Future 3D Feature Alignment

To provide predictive guidance for coordinated action, we first train the Brain to anticipate future collaborative scene outcomes. We obtain supervision from a frozen 3D foundation model (Wang et al., 2025), which extracts scene features from future multi-view observations in demonstrations. By predicting these future features from the current observations and language instruction, the Brain learns representations that capture how the collaborative scene is expected to evolve.

Forming Robot-Specific Guidance with Learnable Queries. The Brain is a Vision-Language Model (VLM) that jointly processes all robots' current RGB observations O_t and the language instruction l. Frozen visual encoders extract patch tokens from each observation view, which are projected into the language model's hidden space and concatenated with the instruction tokens. We then append N sets of learnable query tokens, $Q_1, \ldots, Q_N$, one for each robot, where

$$Q_i = (q_{i,1}, \ldots, q_{i,M}) \tag{1}$$

contains M learnable queries associated with robot i. All query tokens attend bidirectionally to one another and to the shared visual and language context, allowing each robot's queries to integrate information across all observation views. The remaining tokens retain the backbone's original causal attention structure. We maintain the same robot ordering across observation views, query sets, and supervision targets, so that Q_i is trained to produce guidance for robot i.

Rather than reading guidance only from the final Brain layer, we collect the hidden states corresponding to Q_i from a subset of L layers at different depths. Let $h_{i,t}^{(j)}$ denote the query hidden states for robot i at the j-th selected layer. We concatenate these states to form its predictive guidance,

$$G_{i,t} = \text{Concat}\left(h_{i,t}^{(1)}, \ldots, h_{i,t}^{(L)}\right). \tag{2}$$

The resulting $G_{i,t}$ serves as the interface through which the shared Brain provides robot-specific information to the corresponding action Expert.

Learning Predictive Guidance from Future 3D Features. The learnable queries above define where robot-specific guidance is read out, but do not make this guidance predictive. We therefore

train their hidden states to predict representations of future collaborative scenes. Given a demonstration at time t, let O_{t+H} denote the joint observations H frames later. A frozen 3D foundation model (i.e., VGGT (Wang et al., 2025)) jointly processes O_{t+H} and produces a geometry-grounded representation $F^{\text{future}}_{i,t+H}$ for each view i.

These future features provide a geometrically informed target for learning the guidance. By jointly reasoning over multiple views to infer 3D scene structure, VGGT captures cross-view geometric relationships. Accordingly, although $F^{\text{future}}_{i,t+H}$ corresponds to a particular robot view, it is contextualized by all future views and thus reflects the collaborative scene rather than that view in isolation. Aligning $G_{i,t}$ with this target encourages the Brain to anticipate future robot configurations and changes to shared objects that may depend on the actions of multiple robots.

We use a shared QFormer-style projector $\mathcal{P}$ to map the multi-depth guidance into the target feature space,

$$F^{\text{pred}}_{i,t+H} = \mathcal{P}\left(G_{i,t}\right). \tag{3}$$

The Brain is trained by aligning this prediction with the corresponding future 3D features,

$$\mathcal{L}_{\text{align}} = \frac{1}{N}\sum_{i=1}^{N}\left\| F^{\text{pred}}_{i,t+H} - F^{\text{future}}_{i,t+H}\right\|_F^2. \tag{4}$$

This objective jointly trains the Brain and projector while keeping the visual encoders and 3D foundation model frozen. Future observations are used only as training supervision. At inference time, the Brain produces $G_{i,t}$ solely from the current multi-robot observations O_t and the language instruction l.

3.3 Adapting Local Experts with Brain Guidance

With the Brain learned and frozen, we adapt a pretrained single-robot policy into each robot's Expert by conditioning its action generation on the corresponding predictive guidance. We introduce lightweight residual guidance adapters into the Expert backbone, allowing its intermediate action features to query the Brain guidance while retaining the pretrained policy as initialization.

Adapting Experts with Residual Brain Guidance. For robot i, let $c_{i,t} = (o_{i,t}, s_{i,t}, l)$ denote its Expert's conditioning inputs at time t. The Expert generates an action chunk $a_{i,t}$ conditioned on $c_{i,t}$ and the corresponding Brain guidance $G_{i,t}$. To incorporate $G_{i,t}$, we attach a residual guidance adapter to each backbone block. As shown at the bottom of Fig. 2, let $u_{i,b,t}$ denote the intermediate features at block b of robot i's Expert. The adapter uses these features to query the corresponding Brain guidance through cross-attention,

$$A_b(u_{i,b,t}, G_{i,t}) = \text{Linear}_{\text{zero},b}\left(\text{CrossAttn}_b\left(u_{i,b,t}, G_{i,t}, G_{i,t}\right)\right). \tag{5}$$

Here, $u_{i,b,t}$ provides the queries, while $G_{i,t}$ provides the keys and values. This allows each backbone block to selectively retrieve predictive scene information from the Brain according to its current intermediate representation. The attended features are then projected to the block's channel dimension through a linear layer whose weights and bias are initialized to zero (Zhang et al., 2023). The adapter therefore initially introduces no change to the pretrained Expert, and its output is incorporated through residual addition,

$$\widetilde{u}_{i,b,t} = u_{i,b,t} + A_b(u_{i,b,t}, G_{i,t}). \tag{6}$$

We adapt each Expert on action demonstrations using the imitation-learning objective of its underlying single-robot policy. Let $\pi_{\theta,i}$ denote robot i's Expert, where θ comprises its backbone and guidance-adapter parameters. Conditioned on the local inputs $c_{i,t}$ and Brain guidance $G_{i,t}$, the Expert is trained to reproduce the demonstrated action chunk $a_{i,t}$ using

$$\mathcal{L}_{\text{act}} = \mathbb{E}_{i,t}\left[\ell_{\text{IL}}\left(\theta; c_{i,t}, G_{i,t}, a_{i,t}\right)\right], \tag{7}$$

where ℓ_{IL} denotes the imitation-learning objective of the chosen Expert. For example, when the Expert is instantiated as a diffusion policy (Chi et al., 2025), ℓ_{IL} corresponds to its standard action-denoising objective. The Brain remains frozen throughout Expert adaptation.

Table 1: Success rate comparison across six multi-robot manipulation tasks with two to four robots. **Bold** highlights the best result in each column.

Method	2 Robots			3 Robots		4 Robots	Avg
	Lift Barrier	Place Food	Stack Cube	Align Camera	Stack Cube	Take Photo	
Centralized-DP	9%	12%	6%	3%	0%	0%	5.00%
Decentralized-DP	58%	20%	20%	19%	22%	20%	26.50%
2D Dense Policy	3%	2%	0%	0%	0%	9%	2.33%
DP3 (XYZ)	30%	21%	1%	3%	0%	9%	10.67%
DP3 (XYZ+RGB)	31%	25%	1%	18%	0%	1%	14.33%
3D Dense Policy	28%	18%	0%	0%	0%	7%	8.83%
GauDP	72%	15%	2%	26%	0%	3%	19.67%
CLS-DP	61%	43%	39%	55%	20%	8%	38%
Ours	**99%**	**82%**	**79%**	**97%**	**84%**	**72%**	**85.5%**

3.4 Learning Strategy and Inference

We first train the Brain using $\mathcal{L}_{\text{align}}$. Each Expert is then initialized from a pretrained single-robot policy and augmented with the residual guidance adapters described above. For Expert adaptation, we freeze the Brain and cache its guidance $G_{i,t}$ for the training demonstrations. Using $\mathcal{L}_{\text{act}}$, we first optimize only the guidance adapters while keeping the pretrained backbone frozen, and then jointly fine-tune the adapters and backbone. Each Expert is adapted independently while the shared Brain remains frozen.

At inference time, the Brain processes the current multi-robot observations O_t and language instruction l in a single forward pass, producing predictive guidance $G_{i,t}$ for each robot. Robot i's Expert $\pi_{\theta,i}$ then conditions on its local inputs $c_{i,t}$ and corresponding guidance $G_{i,t}$ to generate an action chunk $a_{i,t}$. The guidance remains fixed during the generation of each action chunk. The robots execute an initial segment of their predicted chunks, observe the updated scene, and repeat the process with new Brain guidance and local inputs. Future observations, the 3D foundation model, and the alignment projector are used only for learning the Brain and are not required at inference time.

4 Experiments

4.1 Experimental Setup

Benchmarks and Metrics. We conduct experiments on RoboFactory (Qin et al., 2025), a simulation benchmark for multi-robot collaborative manipulation. Following the benchmark setting of GauDP and CLS-DP (Wang et al., 2026; Park et al., 2026), we evaluate six tasks involving two to four robots that cover tight synchronization, role-asymmetric coordination, and strict sequential dependencies. We use task success rate as the primary evaluation metric. Each configuration of our method is evaluated over 100 episodes per task, and we report both per-task performance and the average across all six tasks.

Baselines. We compare our method with image-based and point-cloud-based policies under centralized and decentralized execution. The image-based baselines include Centralized-DP and Decentralized-DP, both adapted from Diffusion Policy (Chi et al., 2025), 2D Dense Policy (Su et al., 2025), centralized GauDP (Wang et al., 2026), and decentralized CLS-DP (Park et al., 2026). The point-cloud-based baselines include DP3 with XYZ or XYZ+RGB inputs (Ze et al., 2024) and 3D Dense Policy (Su et al., 2025). We use published results under the RoboFactory evaluation protocol: Decentralized-DP from RoboFactory (Qin et al., 2025), the centralized baselines from GauDP (Wang et al., 2026), and CLS-DP from its original paper (Park et al., 2026).

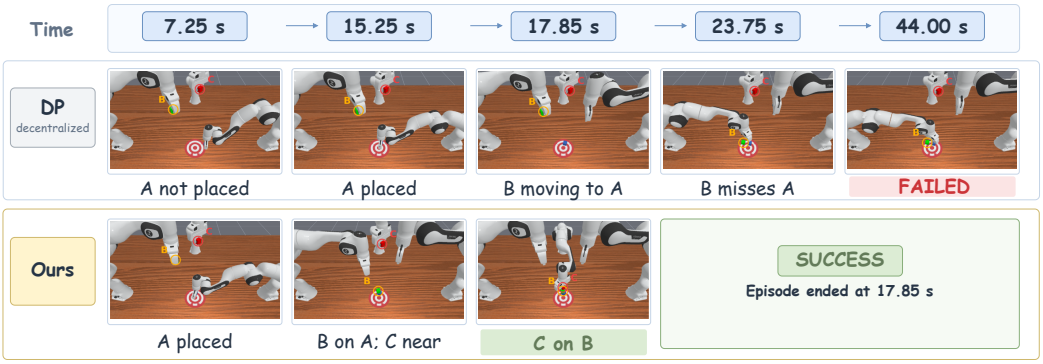

Figure 3: **Qualitative comparison on Three Robots Stack Cube.** The robots must place cube A on the target, stack B on A, and stack C on B. Each column shows both rollouts at the same elapsed time. Our method (bottom) completes the three placements, whereas Decentralized-DP (top) misses A when placing B and fails with C still held.

Implementation Details. Our implementation uses a Qwen2.5-0.5B Brain with frozen DINOv2 and SigLIP visual encoders, while each action Expert uses a ResNet-18 encoder and a 1D U-Net. All images are resized to 224×224. We align hidden states from Brain blocks 8, 13, 18, and 23 with teacher features extracted from the future observation at $t + H$, where $H = 8$. When $t + H$ exceeds the end of an episode, we use the final available observation as the future target. For the alignment loss, we normalize each view's error by its target token count, average over valid views within each sample, and then average across the batch. We train the Brain for 50 epochs before freezing it. The shared projector uses two self-attention layers followed by a linear projection to map the concatenated Brain states to the teacher feature space. We pretrain the single-robot action model for 200 epochs on three single-robot tasks, train only the guidance adapters for 50 epochs, and then fine-tune each Expert together with its adapters for 150 epochs. All training stages use a learning rate of 10^{-4} and a batch size of 256. At inference, each Expert conditions on two observation steps, predicts a 16-step action chunk, and executes eight action steps before replanning. We use VGGT (Wang et al., 2025) as the frozen 3D teacher for future feature supervision by default and evaluate Depth Anything 3 (DA3) (Lin et al., 2025) as an alternative teacher in Section 4.4.

4.2 Quantitative Results

Multi-Robot Manipulation. As shown in Table 1, our method achieves the highest success rate on all six tasks, with an average of 85.5%. It outperforms the best-performing decentralized baseline, CLS-DP, and centralized baseline, GauDP, by 47.5 and 65.8 percentage points in average success rate, respectively. The gains are most pronounced on tasks in which robots build on the outcomes of their teammates' earlier steps. On Three Robots Stack Cube and Take Photo, our method reaches 84% and 72%, whereas no baseline exceeds 22% on either task. These results demonstrate consistent improvements across teams of two to four robots and different coordination structures, with particularly large gains on tasks involving strict sequential dependencies.

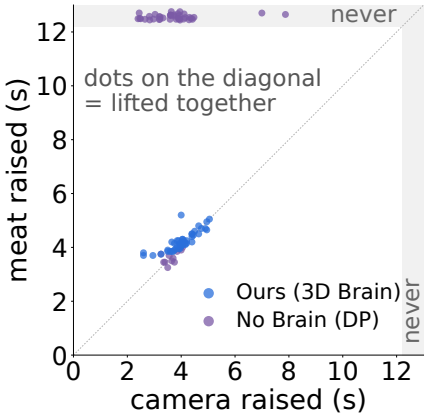

Figure 4: Lift timing on Align Camera

4.3 Qualitative Comparison

We examine the coordination behavior on Three Robots Stack Cube (Figure 3), where the robots must place cube A on the target, stack B on A, and then stack C on B. Our method completes the three placements in sequence, whereas Decentralized-DP misses A when placing B, leaving the robot

Table 2: Ablation study across six multi-robot manipulation tasks with two to four robots. The full model uses VGGT as the 3D teacher, and each subsequent row modifies one component.

Variant	2 Robots			3 Robots		4 Robots	Avg. ↑
	Lift Barrier	Place Food	Stack Cube	Align Camera	Stack Cube	Take Photo	
Full model	99%	82%	79%	97%	84%	72%	**85.5%**
w/o future-alignment (H=0)	96%	83%	10%	99%	53%	67%	68.0%
w/o alignment	59%	0%	0%	0%	0%	0%	9.8%
w/o multi-view input	97%	82%	73%	95%	78%	60%	80.8%
Alignment w/ DA3 (Lin et al., 2025)	96%	78%	49%	94%	61%	81%	76.5%

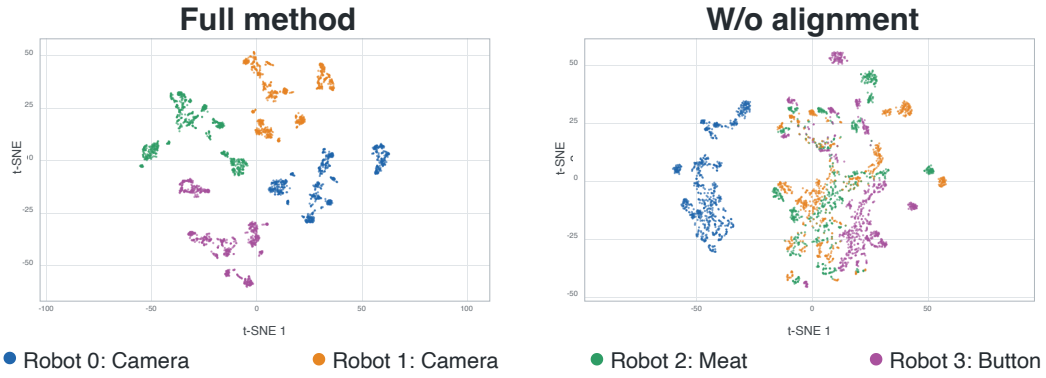

Figure 5: **t-SNE of Brain guidance on Take Photo.** Each point is the Brain guidance that one robot receives when predicting one action chunk. Robots 1 and 2 (blue and orange) handle the camera, and Robots 3 and 4 (green and purple) handle the meat and the button, respectively. With future 3D feature alignment (left), the guidance for each robot forms separate clusters. Without alignment (right), the guidance of Robots 2-4 is mixed.

holding C unable to proceed. This example highlights the benefit of shared predictive context. By observing all robots' views and learning from geometry-grounded future scene features, the Brain provides each Expert with scene information beyond its local observation. Each Decentralized-DP policy instead acts without such shared predictive guidance.

We also examine temporal coordination on Camera Alignment, where two robots raise the camera while the third raises the object to align with it (Figure 4). With Brain guidance, the third robot raises the object at nearly the same time as the camera. Removing the guidance weakens this synchronization, while Decentralized-DP fails to raise the object in most episodes.

4.4 Ablation Study

We ablate the design of the Brain on all six tasks (Table 2). Each variant changes one design choice of the Brain from the full model, which uses VGGT (Wang et al., 2025) as the 3D teacher.

Future 3D Feature Alignment. When feature alignment is removed, the Brain is instead trained jointly with the action Experts using only the action loss, and the average success rate drops from 85.5% to 9.8%. On Take Photo, its guidance for Robots 2–4 is mixed, whereas the full model yields separate clusters for each robot, even for the two robots that handle the same camera (Figure 5). With $H = 0$, the Brain aligns with features of the current observations instead, and the average drops to 68.0%, mostly on the two Stack Cube tasks (79% to 10% and 84% to 53%). Current-frame alignment supervises the Brain to represent the observed scene, whereas future alignment requires it to anticipate how the collaborative scene will evolve. This distinction is particularly important for stacking, where each placement builds on the outcome of a teammate's earlier one.

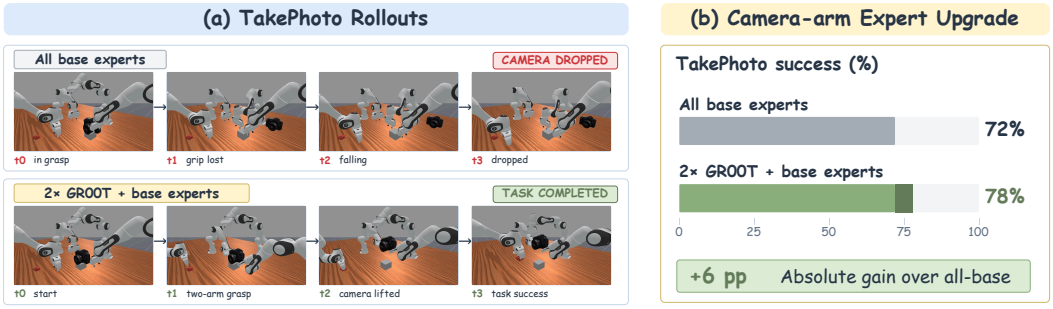

Figure 6: **Localized expert upgrade on Take Photo.** (a) A common failure pattern in Take Photo, in which the camera slips from the gripper after being grasped and falls to the table (top). Replacing only the action experts of the two camera-handling robots with GR00T yields a two-arm grasp that lifts the camera, and the team completes the task (bottom). (b) This localized upgrade raises the success rate from 72% to 78%, while the Brain and the other two action experts remain unchanged.

Shared versus Local Observations. In the local variant, we feed each robot's view to the Brain separately and keep the alignment target unchanged. The average success rate drops from 85.5% to 80.8%, most notably on the four-robot Take Photo task (72% to 60%). With only its own view, the Brain must anticipate changes caused by teammates and objects that may not be directly visible, whereas the shared Brain can observe them through the other robots' views.

3D Teacher. Replacing VGGT with Depth Anything 3 (DA3) (Lin et al., 2025) as the teacher lowers the average success rate from 85.5% to 76.5%, although this variant still outperforms the best-performing baseline on every task. The drop concentrates on the two Stack Cube tasks (79% to 49% and 84% to 61%), whereas DA3 improves Take Photo from 72% to 81%. Future 3D feature alignment thus remains effective with an alternative 3D teacher.

4.5 Localized Expert Upgrade

Because each robot has its own Action Expert, an individual action model can be upgraded without retraining the Brain or the other Experts. We examine this on Take Photo, where a common failure pattern is that the camera slips from the gripper after being grasped and falls to the table (Figure 6(a)). Since this failure arises in the two robots that handle the camera, we replace only their Action Experts with GR00T N1.7 (Bjorck et al., 2025), a pretrained vision-language-action model. We keep its vision-language backbone frozen and fine-tune only its action head, which receives Brain guidance through guidance adapters. This localized upgrade raises the success rate from 72% to 78%, while the Brain and the other two Experts remain unchanged (Figure 6(b)). These results demonstrate that the frozen Brain can guide action models with a different architecture, allowing selected Experts to be upgraded without modifying the rest of the team.

5 Conclusion

In this paper, we presented a multi-robot manipulation framework that separates team-level coordination from robot-specific action generation through a shared predictive Brain and individual Action Experts. The Brain integrates observations across robots and learns to anticipate collaborative scene outcomes through multi-depth alignment with future 3D features, producing robot-specific predictive guidance for each Expert. Each Expert then combines its guidance with local observations to generate actions, preserving the modularity of pretrained single-robot policies while enabling coordination through a shared understanding of the future scene. Experiments on RoboFactory demonstrate improved multi-robot task performance over decentralized and centralized baselines, while allowing individual Experts to be replaced or adapted without retraining the Brain or other controllers. These results demonstrate the effectiveness of combining shared predictive representations with independently adaptable controllers for multi-robot manipulation.

AI USE STATEMENT

Codex assisted with code implementation. The authors manually reviewed the AI-assisted code and used AI tools for an additional cross-check. ChatGPT helped edit the manuscript's language and assess its presentation. AI tools also surfaced potentially overlooked related work, which the authors evaluated while conducting the literature search and selecting citations. The authors supervised training, reviewed each paragraph of the manuscript, and made the final research and editorial decisions. They are responsible for the manuscript's content and claims, as well as any artifacts developed with AI assistance.

REFERENCES

Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, et al. Gr00t n1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025.

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.

Jun Cen, Chaohui Yu, Hangjie Yuan, Yuming Jiang, Siteng Huang, Jiayan Guo, Xin Li, Yibing Song, Hao Luo, Fan Wang, et al. Worldvla: Towards autoregressive action world model. *arXiv preprint arXiv:2506.21539*, 2025.

Yincong Chen, Ranpeng Qiu, Zihao Li, Yanan Zhou, Guoqiang Ren, and Weiming Zhi. Robots that collaborate: Sequential asymmetric imitation for learning coupled robot policies. *arXiv preprint arXiv:2606.16490*, 2026.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

Dayi Dong, Maulik Bhatt, Seoyeon Choi, and Negar Mehr. Mimic-d: Multi-modal imitation for multi-agent coordination with decentralized diffusion policies. *arXiv preprint arXiv:2509.14159*, 2025.

Ria Doshi, Tian Gao, Annie Chen, Chelsea Finn, and Jeannette Bohg. Chorus: Decentralized multi-embodiment collaboration with one vla policy. *arXiv preprint arXiv:2606.12352*, 2026.

Zipeng Fu, Tony Z Zhao, and Chelsea Finn. Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation. *arXiv preprint arXiv:2401.02117*, 2024.

Chengyang He, Gadiel Sznaier Camps, Xu Liu, Mac Schwager, and Guillaume Sartoretti. Latent theory of mind: A decentralized diffusion architecture for cooperative manipulation. *arXiv preprint arXiv:2505.09144*, 2025.

Chi-Pin Huang, Yunze Man, Zhiding Yu, Min-Hung Chen, Jan Kautz, Yu-Chiang Frank Wang, and Fu-En Yang. Fast-thinkact: Efficient vision-language-action reasoning via verbalizable latent planning. *arXiv preprint arXiv:2601.09708*, 2026a.

Chi-Pin Huang, Yueh-Hua Wu, Min-Hung Chen, Frank Wang, and Fu-En Yang. Thinkact: Vision-language-action reasoning via reinforced visual latent planning. *Advances in Neural Information Processing Systems*, 38:82782–82802, 2026b.

Hokyun Im, Euijin Jeong, Andrey Kolobov, Jianlong Fu, and Youngwoon Lee. Twinvla: Data-efficient bimanual manipulation with twin single-arm vision-language-action models. In *International Conference on Learning Representations*, volume 2026, pp. 61969–61995, 2026.

Daqi Jiang, Hong Wang, and Yanzheng Lu. Mastering the complex assembly task with a dual-arm robot: A novel reinforcement learning method. *IEEE Robotics & Automation Magazine*, 30(2): 57–66, 2023.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

Matthew Lai, Keegan Go, Zhibin Li, Torsten Kröger, Stefan Schaal, Kelsey Allen, and Jonathan Scholz. Roboballet: Planning for multirobot reaching with graph neural networks and reinforcement learning. *Science Robotics*, 10(106):eads1204, 2025.

Fuhao Li, Wenxuan Song, Han Zhao, Jingbo Wang, Pengxiang Ding, Donglin Wang, Long Zeng, and Haoang Li. Spatial forcing: Implicit spatial representation alignment for vision-language-action model. In *International Conference on Learning Representations*, volume 2026, pp. 132324–132345, 2026.

Haotong Lin, Sili Chen, Junhao Liew, Donny Y Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647*, 2025.

Wenhang Liu, Meng Ren, Kun Song, Michael Yu Wang, and Zhenhua Xiong. A planning framework for complex flipping manipulation of multiple mobile manipulators. *IEEE Robotics and Automation Letters*, 10(5):5162–5169, 2025.

Guanxing Lu, Tengbo Yu, Haoyuan Deng, Season Si Chen, Yansong Tang, and Ziwei Wang. Anybimanual: Transferring unimanual policy for general bimanual manipulation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 13662–13672. IEEE, 2025.

Luca Nunziante, Kentaro Uno, Gustavo H Diaz, Shreya Santra, Alessandro De Luca, and Kazuya Yoshida. Assembling solar panels by dual robot arms towards full autonomous lunar base construction. In *2025 IEEE/SICE International Symposium on System Integration (SII)*, pp. 1497–1502. IEEE, 2025.

Chanyoung Park, Minsung Yoon, Andrew Jeong, and Sung-eui Yoon. Distilling collaborative dynamics into latent space for implicit coordination in decentralized multi-agent manipulation. *arXiv preprint arXiv:2606.22982*, 2026.

Yiran Qin, Li Kang, Xiufeng Song, Zhenfei Yin, Xiaohong Liu, Xihui Liu, Ruimao Zhang, and Lei Bai. Robofactory: Exploring embodied agent collaboration with compositional constraints. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 10075–10085. IEEE, 2025.

Yue Su, Xinyu Zhan, Hongjie Fang, Han Xue, Hao-Shu Fang, Yong-Lu Li, Cewu Lu, and Lixin Yang. Dense policy: Bidirectional autoregressive learning of actions. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 14486–14495. IEEE, 2025.

Guoheng Sun, Tingting Du, Kaixi Feng, Chenxiang Luo, Xingguo Ding, Zheyu Shen, Ziyao Wang, Yexiao He, and Ang Li. Rocket: Residual-oriented multi-layer alignment for spatially-aware vision-language-action models. *arXiv preprint arXiv:2602.17951*, 2026.

Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, et al. Octo: An open-source generalist robot policy. *arXiv preprint arXiv:2405.12213*, 2024.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5294–5306. IEEE, 2025.

Zijian Wang and Mac Schwager. Kinematic multi-robot manipulation with no communication using force feedback. In *2016 IEEE international conference on robotics and automation (ICRA)*, pp. 427–432. IEEE, 2016.

Ziye Wang, Li Kang, Yiran Qin, Jiahua Ma, LEI BAI, Ruimao Zhang, et al. Gaudp: Reinventing multi-agent collaboration through gaussian-image synergy in diffusion policies. *Advances in Neural Information Processing Systems*, 38:5620–5639, 2026.

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.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3813–3824. IEEE, 2023.

Qingqing Zhao, Yao Lu, Moo Jin Kim, Zipeng Fu, Zhuoyang Zhang, Yecheng Wu, Zhaoshuo Li, Qianli Ma, Song Han, Chelsea Finn, et al. Cot-vla: Visual chain-of-thought reasoning for vision-language-action models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1702–1713. IEEE, 2025.

Ruijie Zheng, Jing Wang, Scott Reed, Johan Bjorck, Yu Fang, Fengyuan Hu, Joel Jang, Kaushil Kundalia, Zongyu Lin, Loic Magne, et al. Flare: Robot learning with implicit world modeling. *arXiv preprint arXiv:2505.15659*, 2025.

A SUPPLEMENTARY MATERIAL

A.1 TASK DESCRIPTIONS

We evaluate on six collaborative tasks from RoboFactory, summarized in Table 3. Each robot is a single arm with a 9-dimensional proprioceptive state and an 8-dimensional action. All robots in a task receive the same one-sentence instruction, such as "Two robots collaboratively lift a barrier." An episode counts as a success if the condition listed for its task is satisfied before the step limit is reached. Training uses the demonstrations provided by the benchmark, with 150 demonstrations for each task.

Table 3: **RoboFactory tasks used in our evaluation.**

Task	Robots	Robot roles	Success condition
Lift Barrier	2	Both robots grasp opposite ends of a long barrier and raise it together.	The barrier reaches the target height and stays stable.
Place Food	2	One robot removes the lid of a pot, and the other puts the food into the pot.	The food lies inside the pot, within a distance threshold of the pot center.
Two Robots Stack Cube	2	One robot moves the blue cube to a target position, and the other stacks the red cube on it.	The blue cube is at the target position, and the red cube rests on top of it.
Camera Alignment	3	Two robots hold a camera from both sides and raise it, while the third places an object at a target position aligned with the camera.	The camera reaches the target height, and the object is at its aligned position.
Three Robots Stack Cube	3	The robots place the blue, red, and green cubes in turn to build a stack at a target position.	The blue cube is at the target position, with the red cube on it and the green cube on top.
Take Photo	4	Three robots align the camera and the object as in Camera Alignment, and the fourth presses the shutter.	The Camera Alignment condition holds, and the fourth end effector is within a threshold of the shutter.

A.2 ADDITIONAL IMPLEMENTATION DETAILS

This section complements the implementation details in the main paper. Table 4 lists the configuration of each training stage.

Table 4: **Training and inference configuration.** Expert epochs are listed for single-robot pretraining, adapter-only training, and joint fine-tuning, respectively.

Setting	Brain (Stage 1)	action experts
Optimizer	AdamW	AdamW
Learning rate	10^{-4}	10^{-4}
Weight decay	0.01	10^{-6}
Batch size	256	256
Epochs	50	200 / 50 / 150
Precision	bfloat16	float32
Selected Brain blocks	8, 13, 18, 23	–
Latent tokens per view	4	–
Future offset H	8 frames	–
Target features	VGGT final block, 261×2048 per view	–
Diffusion steps (train / inference)	–	100 / 100
Observation steps	–	2
Predicted / executed action steps	–	16 / 8

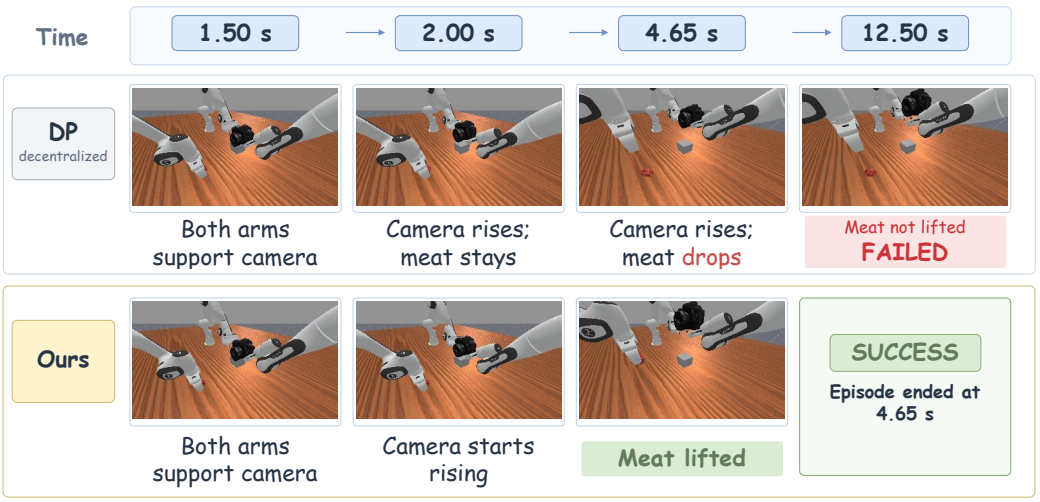

Figure 7: **Qualitative comparison on Camera Alignment.** Two robots must raise the camera while the third lifts the object into alignment with it. Each column shows both rollouts at the same elapsed time. Our method (bottom) lifts the object together with the camera, whereas Decentralized-DP (top) fails to lift the object.

Brain. The Brain is initialized from a Prismatic VLM checkpoint that combines Qwen2.5-0.5B, with 24 blocks and hidden size 896, and DINOv2 and SigLIP visual encoders. The visual encoders remain frozen, and the Brain has 385M trainable parameters. Each image is resized directly to 224×224 without cropping, following the preprocessing of the pretrained checkpoint. Because the number of visual tokens is fixed within a forward pass, each training batch contains samples with the same number of views. The shared projector $\mathcal{P}$ is implemented as a QFormer-style module with two self-attention layers, eight attention heads, and hidden width 896, followed by a linear projection to the VGGT feature dimension. The future offset $H = 8$ corresponds to 0.4 s at the 20 Hz control rate. We train the Brain on four GPUs with 10 warmup steps and gradient-norm clipping at 1.0.

Action expert. Each expert follows Diffusion Policy. A ResNet-18 with group normalization encodes each RGB frame into a 128-dimensional feature. A frozen SigLIP 2 text encoder embeds the instruction, which is then projected to 128 dimensions. The image features and proprioceptive states from the two most recent steps, together with the instruction embedding, form a 402-dimensional condition that enters the 1D U-Net through FiLM. The expert predicts the added noise under a squared-cosine noise schedule. Every block of the U-Net has its own guidance adapter, which attends to the concatenated Brain states from all selected blocks. Each adapter uses 4-head attention with width 128. The adapters add 3.65M parameters to the 80.1M parameters of the pretrained policy.

Training and inference. The single-robot policy is pretrained on the single-robot tasks of RoboFactory. These tasks share the proprioceptive and action spaces of the multi-robot tasks and contain no multi-robot demonstrations. The six evaluation tasks involve 16 robots in total, and each robot receives its own expert initialized from this policy. For Stage 2, we compute the guidance of the frozen Brain once and cache it. The optimizer state and the exponential moving average (EMA) of the expert weights carry over from adapter-only training to joint fine-tuning, and we evaluate the EMA weights.

B Additional Qualitative Results

We qualitatively compare our method with Decentralized-DP on Camera Alignment (Figure 7), where two robots raise the camera while the third lifts the object into alignment with it. Our method lifts the object together with the camera, whereas Decentralized-DP raises the camera but fails to

lift the object. This rollout is consistent with the lift-timing analysis in the main paper, where Decentralized-DP fails to raise the object in most episodes. We attribute this difference to the shared Brain. It observes all robots' views and is trained to anticipate the future scene, in which the object rises together with the camera. Its guidance therefore conveys the camera's progress to the third robot, which cannot infer it reliably from its own view. Each Decentralized-DP policy instead acts without a representation shared with its teammates.

C LIMITATIONS

Because the Shared Predictive Brain is trained solely through alignment with features from a frozen 3D foundation model, our method inherits the limitations of this teacher. Inaccuracies in the teacher features, such as imprecise geometry when objects are heavily occluded by robot arms or observed from few overlapping views, become part of the alignment targets, and the resulting guidance may misrepresent spatial relationships between robots and shared objects. Each action expert still receives its own observation, proprioceptive state, and instruction, which may partially compensate for inaccurate guidance. Since the teacher is used only during Brain training, it can be replaced by a stronger 3D foundation model without changing the architecture or the inference procedure.

D BROADER IMPACTS

This work aims to enable robot teams to coordinate on manipulation tasks that a single robot cannot complete alone, which may benefit applications such as manufacturing, logistics, and household assistance. Because our method reuses a policy pretrained on single-robot tasks, it may also reduce the cost of collecting multi-robot demonstrations. However, inaccurate guidance or poorly synchronized actions could cause collisions or dropped objects when deployed on physical robots near people, so real-world use would require safety constraints, collision monitoring, and human oversight beyond the simulated evaluation in this paper.