Helping Hand: Completing Collaborative Tasks with Dexterous Hands

Author Names Omitted for Submission

I. Introduction

Robots are autonomous mechanical systems designed to assist humans with complex tasks. In many scenarios, effective human-robot collaboration is essential, whether through physical assistance or more nuanced forms of interaction. However, building robots that can collaborate naturally and intuitively with humans remains a significant challenge. Directly training control policies on specific tasks often leads to overfitting and fails to capture high-level task semantics or human intent.

Large language models (LLMs) have recently demonstrated impressive capabilities in reasoning, generalization, and multimodal understanding, making them promising candidates for enabling more flexible robotic behaviors. Yet, directly applying LLMs to real-world collaborative robotics remains impractical for two key reasons: (1) LLMs lack the mechanisms to bridge the gap between abstract reasoning and low-level control, and (2) they rely heavily on explicit language prompting, which introduces latency and inefficiency in real-time interactions.

To enable smoother and more intuitive collaboration, we envision a policy that minimizes the need for language-based prompting and instead infers human intent directly from motion cues — enabling a robot to act through **tacit understanding**.

In this project, we propose a novel approach that enables pre-trained vision-language-action (VLA) models for dexterous collaborative tasks based on body-language. We introduce several methods to improve adaptation to real-world human-robot interaction: leveraging pre-trained vision encoders, incorporating human pose priors, and re-designing the model's action space. Our approach enhances the robot's ability to perceive, interpret, and respond to human behaviors in a context-aware and data-efficient manner. Real-world evaluations demonstrate the effectiveness of the proposed method.

II. Related Work

A. Human-Robot Interaction

Human-robot interaction (HRI) is a longstanding area of research aimed at improving the ways in which robots assist and collaborate with humans. Prior work has explored a variety of methods to enhance robot responsiveness, intention understanding, and physical cooperation. For example, Roveda et al. [1] employed fuzzy controllers to support humans in industrial settings. Yan et al. [2] used long short-term memory (LSTM) networks for intention recognition in human-robot interaction. Similarly, Zhang et al. [3] applied recurrent models to predict human motion during assembly tasks to facilitate handovers. More recently, Wojtak et al. [4] proposed using neural fields for learning object handover behaviors, while Ji et al. [5] and Wang et al. [6] explored foundation model-based approaches for collaborative assembly and tabletop interaction, respectively. However, these methods often depend on handcrafted robotic APIs and suffer from high inference latency, limiting their real-time applicability.

In contrast, we propose a system that enables real-time, smooth human-robot collaboration by directly generating robot actions from multimodal observations, without relying on predefined action schemas.

B. Learning from Demonstrations

Learning from demonstrations (LfD), also known as imitation learning, is a widely adopted paradigm in robotic learning [7]. By mimicking human behavior, robots can acquire complex skills without requiring manually designed reward functions. Classical approaches include Behavior Cloning (BC), which maximizes the likelihood of expert actions given observed states, and Inverse Reinforcement Learning (IRL) [8], which infers the underlying reward function from demonstrations. DAgger [9] addresses distributional shift by iteratively querying the expert in an online setting.

To improve data efficiency and handle imperfect demonstrations, more recent methods incorporate probabilistic and generative modeling. Huang et al. [10] proposed a Gaussian Mixture Model (GMM)-based framework for few-shot learning in long-horizon tasks, while Bütepage et al. [11] used generative models for imitation in human-robot interaction scenarios.

The emergence of large-scale robotic datasets [12], [13] has enabled the development of generalist policies trained with simple imitation objectives. These datasets support scaling imitation learning to diverse tasks and environments.

C. Vision-Language-Action (VLA) Models

Recent advances in large language models (LLMs) [14], [15] have demonstrated strong capabilities in reasoning, abstraction, and multimodal alignment. This has motivated efforts to apply LLMs to robotics, where they could bridge perception and action through natural language.

Preliminary works such as Text2Motion [16] and VoxPoser [17] have explored this direction. Building on large-scale multimodal datasets and vision-language pretraining [18]–[20], researchers have introduced VLA models that process visual and linguistic inputs to directly generate tokenized

Fig. 1: The snapshots of the robot carrying out successful real-world inference. The rows from top to bottom are the front and top view of task *pass cube*, front and top view of task *pick up cube* and front and top view of the combined long-horizon task. Each sequence is executed from left to right.

robot actions [21]–[23]. These models are trained using next-token prediction over sequences of multimodal inputs and demonstrations.

VLA models exhibit strong generalization and compositionality, allowing them to handle open-ended, unstructured tasks. They can also be adapted to specific domains via fine-tuning, making them a promising foundation for learning collaborative robot behaviors from modest data.

III. Task Design

We design two toy tasks for demonstrating our method: "*pick up cube*" and "*pass cube*". These tasks were carefully selected because they are illustrative of core capabilities required for human-robot collaboration. Specifically:

1) They demonstrate the robot's ability to assist a human physically, through object manipulation and transfer.
2) They can be composed into a longer sequence — first picking up an object indicated by the human, then passing it back — showcasing the model's ability to execute long-horizon, goal-directed behavior.
3) They require the robot to interpret human body language rather than relying on explicit natural language instructions, aligning with our goal of enabling tacit understanding.

The "*pick up cube*" task involves two cubes placed on a table randomly, including one red and one blue. The human collaborator points to one cube, and the robot must infer the intention and pick up the designated object.

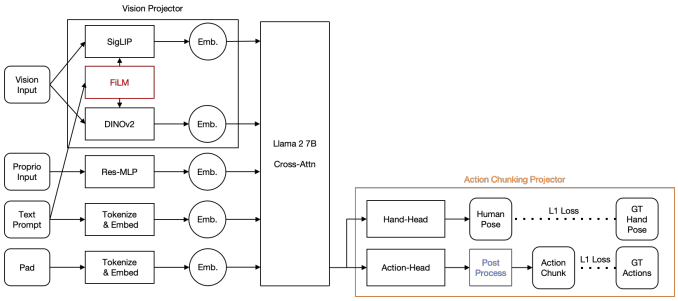


Fig. 2: The modified model structure. Red block represents the FiLM layers added to vision encoders, orange block represents the modified action chunking projector, blue block represents the modified action post-processing module.

The "*pass cube*" task begins with the robot already holding a cube. The robot is required to pass the object to the human collaborator and release it appropriately.

IV. Model Architecture

Our approach builds upon Open-VLA [22], a widely-used vision-language-action model. For visual perception, Open-VLA incorporates pre-trained encoders from SigLIP [24] and DINOv2 [25]. Language inputs are processed using a pre-trained LLaMA2-7B model [26]. These components are integrated into a unified multimodal transformer that fuses visual, linguistic, and proprioceptive inputs to generate robot actions.

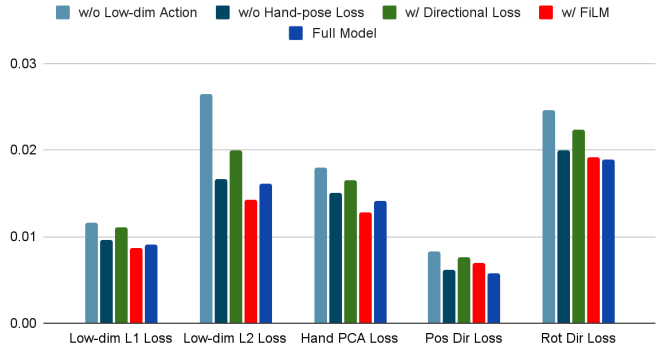


Fig. 3: Ablation study results. The "Full Model" includes action post-processing and hand-pose auxiliary loss, but excludes directional loss and FiLM conditioning.

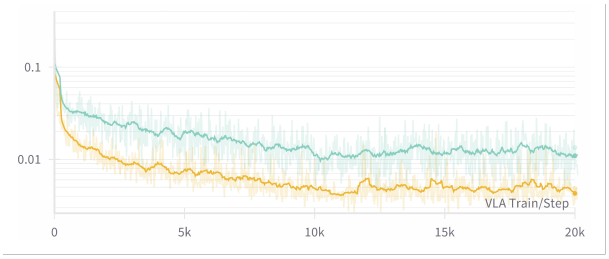


Fig. 4: Auxiliary loss when evaluated on on data from the same collaborator vs. different collaborators. Orange curve: same hand as training. Blue curve: different hand.

To better adapt Open-VLA to collaborative settings, we introduce several key modifications, illustrated in fig. 2, and analyzed subsequently:

1) **FiLM conditioning** [27]: We insert FiLM layers into both vision encoders to improve cross-modal conditioning from text.
2) **Auxiliary intention loss**: We add an auxiliary prediction head to explicitly learn human intention by regressing collaborator hand pose.
3) **Action post-processing**: We constrain action predictions to a more compact and structured subspace, improving stability and learning efficiency.
4) **Directional loss**: We apply a directional loss on end-effector pose that emphasizes directional alignment while downweighting magnitude.

These modifications collectively improve the model's ability to interpret human cues and generate responsive, context-aware robot behavior in collaborative tasks.

A. FiLM Conditioning

Feature-wise Linear Modulation (FiLM) [27] is a technique for conditioning a vision encoder on additional inputs, typically text. FiLM layers apply affine transformations to feature maps, where the scale and bias are functions of the conditioning input. This enables the model to dynamically adjust visual representations based on linguistic context.

In the context of VLA models, FiLM layers allow the vision backbone to better align visual perception with task-specific language prompts. We incorporate FiLM conditioning into both vision encoders (SigLIP and DINOv2) and evaluate its impact on task performance in collaborative settings. We observed **consistent improvement** on loss function when FiLM conditioning is activated.

B. Auxiliary Loss

To enhance the model's understanding of human intent, we introduce human pose priors into training. A straightforward approach would be to extract pose-related features and feed them into the model via cross-attention. However, this method does not scale well: as the number of tasks and priors (e.g., grasp points, object bounding boxes) increases, it would require designing and maintaining multiple feature extractors.

Instead, we adopt an auxiliary loss formulation that encourages the model to implicitly learn human intention cues. Specifically, we add an auxiliary prediction head—referred to as the *hand head*—in parallel with the *action head*. This head receives the same model input and is trained to predict: (1) the 2D hand pose of the collaborator in each camera view, and (2) the color of the target cube. Hand pose annotations are extracted using MediaPipe [28], and the target object label is derived from task metadata.

The auxiliary loss is defined as the L2 distance between the predicted and ground-truth labels. During inference, the *hand head* is disabled, as it does not contribute to action generation. We observed **consistent improvement** when this loss is used.

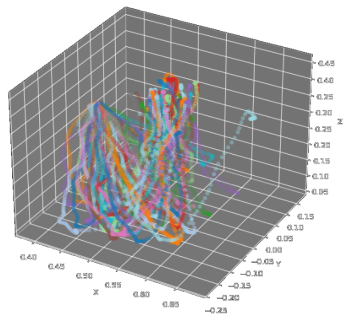
C. Action Post-processing

The original action space—comprising 3D position, 4D rotation (quaternion), and 16 joint positions—totals 23 dimensions. However, the underlying structure of valid actions likely lies on a lower-dimensional manifold, making it difficult for the model to learn effectively in the raw space.

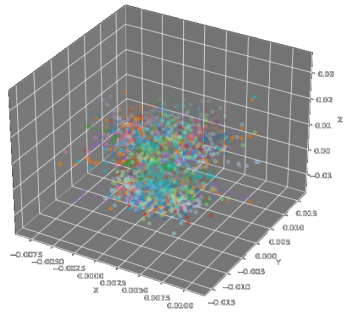
To address this, we reformulate the action space so that the model predicts actions in a compact, transformed space, which are then mapped back to the original representation via post-processing. This process consists of three stages:

- **Position**: Let p denote the current end-effector position, and a'_p the model's predicted delta. The final position command is computed as $a_p = p + a'_p$.
- **Rotation**: Let q be the current end-effector rotation in quaternion form, and $d = (\omega, x, y, z)$ be the a delta quaternion. The output rotation is computed as $a_r = q \cdot d$. The model predicts in the rotation vector form: $a'_r = \frac{(x,y,z)}{\sqrt{1-\omega^2}}$.
- **Hand joints**: We apply PCA to the 16-dimensional hand joint states in the training data and retain the top principal components. During inference, the model predicts in this low-dimensional PCA space, and the full joint configuration is reconstructed via inverse PCA.

We observed **significant improvement** when the action post-processing is enabled.



(a) Original action distribution



(b) Delta action distribution

Fig. 5: Differentiating the action sequence results in a smoother and more normally distributed action space.

D. Directional Loss

To improve the stability and relevance of end-effector motion, we design a **directional loss** that emphasizes the direction of movement rather than its magnitude. Let x denote the predicted delta pose, and y the ground-truth delta pose. We decompose x into two orthogonal components: $x_{\parallel}$ (parallel to y) and $x_{\perp}$ (orthogonal to y).

The directional loss is defined as:

$$\mathcal{L}_{\text{dir}} = r \cdot \frac{\|x_{\parallel}\|_2}{\|y\|_2 + r} + \|x_{\perp}\|_2,$$

where $r < 1$ is a scaling factor. When $\|y\|_2 \gg r$, the loss emphasizes directional alignment; when $\|y\|_2 \ll r$, it reduces to a standard L2 loss. However we see **no improvement** on the performance when this loss is enabled.

V. Findings

A. Action Space for Dexterous VLA

The motivation behind action post-processing is based on the assumption that the true action space lies on a low-dimensional manifold embedded in a high-dimensional space. Without explicitly modeling this structure, the model may struggle to learn meaningful mappings. By reducing the dimensionality, the action manifold can be transformed into a more compact and convex representation, facilitating more efficient learning.

We first analyze the action subspace related to end-effector pose. As shown in fig. 5a, the distribution of the raw xyz position components across trajectories is highly non-convex. However, when we differentiate the action sequence—i.e., consider relative rather than absolute motion—the resulting delta poses exhibit a much smoother and compact distribution fig. 5b.

Next, we examine the hand joint subspace, which has 16 dimensions. We hypothesize that despite this high dimensionality, the actual configuration space is low-dimensional. To test this, we perform principal component analysis (PCA) on all hand joint states in the training set. The results, shown in fig. 6, reveal that just four principal components account for 96% of the total variance. This suggests that PCA-reduced components can be effectively used as the action representation, replacing the original high-dimensional hand joint space.

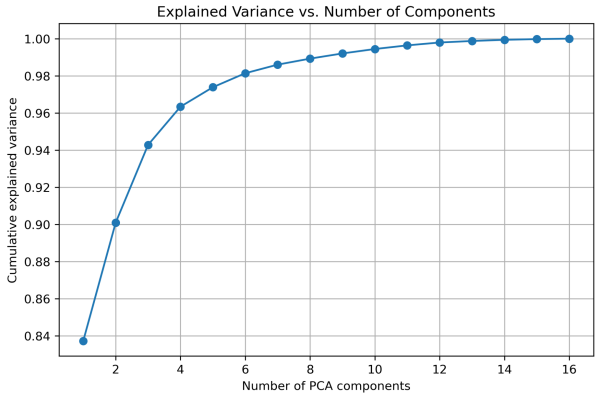


Fig. 6: PCA analysis of hand joint states. Four principal components explain 96% of the variance.

B. Auxiliary Predictions and Trainer Overfitting

When we evaluate the model in real world set ups, we discovered an interesting fact: when trained on data collected from one specific collaborator, the model accurately interprets their intentions during inference. However, when interacting with a different person, it fails to adapt and instead reverts to a fixed routine—behavior as if no meaningful commands were received.

We name this phenomenon as **trainer overfitting**: the model becomes overly specialized to the behavior of a single demonstrator. This overfitting is also common in intelligent creatures, for example, dogs only follow the commands of their owners [29]. To further analyze this phenomenon, we conducted an experiment that uses the auxiliary loss to quantify **trainer overfitting**. We trained the model with human collaborator A and tested the model on data from both collaborator A and collaborator B, and plotted the loss curve in fig. 4. The elevated loss confirms that the model fails to generalize across different collaborators.

This finding opens new possible research directions on how to reduce **trainer overfitting** in collaborative robots.

REFERENCES

[1] L. Roveda, S. Haghshenas, M. Caimmi, N. Pedrocchi, and L. Molinari Tosatti, "Assisting operators in heavy industrial tasks: On the design of an optimized cooperative impedance fuzzy-controller with embedded safety rules," *Frontiers in Robotics and AI*, vol. 6, p. 75, 2019.

[2] L. Yan, X. Gao, X. Zhang, and S. Chang, "Human-robot collaboration by intention recognition using deep lstm neural network," in *2019 IEEE 8th International Conference on Fluid Power and Mechatronics (FPM)*. IEEE, 2019, pp. 1390–1396.

[3] J. Zhang, H. Liu, Q. Chang, L. Wang, and R. X. Gao, "Recurrent neural network for motion trajectory prediction in human-robot collaborative assembly," *CIRP annals*, vol. 69, no. 1, pp. 9–12, 2020.

[4] W. Wojtak, F. Ferreira, P. Vicente, L. Louro, E. Bicho, and W. Erlhagen, "A neural integrator model for planning and value-based decision making of a robotics assistant," *Neural Computing and Applications*, vol. 33, no. 8, pp. 3737–3756, 2021.

[5] Y. Ji, Z. Zhang, D. Tang, Y. Zheng, C. Liu, Z. Zhao, and X. Li, "Foundation models assist in human–robot collaboration assembly," *Scientific Reports*, vol. 14, no. 1, p. 24828, 2024.

[6] C. Wang, S. Hasler, D. Tanneberg, F. Ocker, F. Joublin, A. Ceravola, J. Deigmoeller, and M. Gienger, "Lami: Large language models for multi-modal human-robot interaction," in *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, 2024, pp. 1–10.

[7] M. Zare, P. M. Kebria, A. Khosravi, and S. Nahavandi, "A survey of imitation learning: Algorithms, recent developments, and challenges," *IEEE Transactions on Cybernetics*, 2024.

[8] S. Arora and P. Doshi, "A survey of inverse reinforcement learning: Challenges, methods and progress," *Artificial Intelligence*, vol. 297, p. 103500, 2021.

[9] S. Ross, G. Gordon, and D. Bagnell, "A reduction of imitation learning and structured prediction to no-regret online learning," in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2011, pp. 627–635.

[10] Y. Huang, J. Silvério, L. Rozo, and D. G. Caldwell, "Generalized task-parameterized skill learning," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 5667–5474.

[11] J. Bütepage, A. Ghadirzadeh, Ö. Öztimur Karadağ, M. Björkman, and D. Kragic, "Imitating by generating: Deep generative models for imitation of interactive tasks," *Frontiers in Robotics and AI*, vol. 7, p. 47, 2020.

[12] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu *et al.*, "Rt-1: Robotics transformer for real-world control at scale," *arXiv preprint arXiv:2212.06817*, 2022.

[13] 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: Open x-embodiment collaboration 0," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 6892–6903.

[14] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[15] 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.

[16] K. Lin, C. Agia, T. Migimatsu, M. Pavone, and J. Bohg, "Text2motion: From natural language instructions to feasible plans," *Autonomous Robots*, vol. 47, no. 8, pp. 1345–1365, 2023.

[17] W. Huang, C. Wang, R. Zhang, Y. Li, J. Wu, and L. Fei-Fei, "Voxposer: Composable 3d value maps for robotic manipulation with language models," *arXiv preprint arXiv:2307.05973*, 2023.

[18] R. Anil, A. M. Dai, O. Firat, M. Johnson, D. Lepikhin, A. Passos, S. Shakeri, E. Taropa, P. Bailey, Z. Chen *et al.*, "Palm 2 technical report," *arXiv preprint arXiv:2305.10403*, 2023.

[19] H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Visual instruction tuning," *Advances in neural information processing systems*, vol. 36, pp. 34 892–34 916, 2023.

[20] 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*, 2024, pp. 26 296–26 306.

[21] D. Driess, F. Xia, M. S. Sajjadi, C. Lynch, A. Chowdhery, A. Wahid, J. Tompson, Q. Vuong, T. Yu, W. Huang *et al.*, "Palm-e: An embodied multimodal language model," *arXiv preprint arXiv:2303.03378*, 2023.

[22] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi *et al.*, "Openvla: An open-source vision-language-action model," *arXiv preprint arXiv:2406.09246*, 2024.

[23] 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," *arXiv preprint arXiv:2405.12213*, 2024.

[24] X. Zhai, B. Mustafa, A. Kolesnikov, and L. Beyer, "Sigmoid loss for language image pre-training," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 11 975–11 986.

[25] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby *et al.*, "Dinov2: Learning robust visual features without supervision," *arXiv preprint arXiv:2304.07193*, 2023.

[26] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale *et al.*, "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint arXiv:2307.09288*, 2023.

[27] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville, "Film: Visual reasoning with a general conditioning layer," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[28] Google AI Edge, "Mediapipe," https://github.com/google-ai-edge/mediapipe, 2024, accessed: 2025-07-13.

[29] I. Merola, E. Prato-Previde, and S. Marshall-Pescini, "Dogs' social referencing towards owners and strangers," *PloS one*, vol. 7, no. 10, p. e47653, 2012.