Article | Received 13 November 2025; Revised 15 December 2025; Accepted 25 December 2025; Published 7 January 2026
https://doi.org/10.55092/rl20260001

TLA: tactile-language-action model for contact-rich manipulation

Peng Hao[1],†, Chaofan Zhang[1],†, Dingzhe Li[2], Xiaoge Cao[1], Xiaoshuai Hao[3], Shaowei Cui[1],* and Shuo Wang[1]

[1] The State Key Laboratory of Multimodal Artificial Intelligence Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China
[2] Samsung Research China–Beijing (SRC-B), Beijing, China
[3] Xiaomi EV, Beijing, China

† These authors contributed equally to this work.

* Correspondence author; E-mail: shaowei.cui@ia.ac.cn.

Highlights:

- We propose a novel Tactile-Language-Action model that fuses sequential tactile feedback with natural language priors, enabling grounded action generation for contact-rich manipulation tasks.
- We construct the TLA dataset, a tactile-action instruction dataset that provides a valuable benchmark for studying language-conditioned tactile manipulation.
- We validate the proposed framework through real-world peg-in-hole assembly experiments, demonstrating its practical applicability and strong generalization across varying clearances and shapes.

Abstract: Significant progress has been made in vision-language models. However, language-conditioned robotic manipulation for contact-rich tasks remains underexplored, particularly with respect to tactile sensing. To address this gap, we propose the Tactile-Language-Action (TLA) model, which processes sequential tactile feedback through cross-modal language grounding to enable robust policy generation in contact-intensive scenarios. We also construct a large-scale and low-cost dataset comprising 24 k pairs of tactile-action instructions in simulation, tailored for fingertip-based peg-in-hole assembly. Experimental results show that TLA significantly outperforms traditional imitation learning methods (e.g., diffusion policy) in terms of action accuracy. Moreover, robotic insertion experiments demonstrate that TLA generalizes well to unseen peg shapes and clearances, achieving a success rate exceeding 85%, and achieves zero-shot sim-to-real transfer for real-world deployment. Project website: https://sites.google.com/view/tactile-language-action/.

Keywords: vision-language-action model; tactile perception; contact-rich manipulation

1. Introduction

Tactile perception plays a critical role in contact-rich robotic manipulation tasks [1]. For instance, in fine assembly scenarios, robots must detect subtle variations on object surfaces to ensure accurate alignment [2]. Tactile sensing enables fine-grained adjustments to contact poses, helping avoid damage or misalignment during interaction [3]. Such precise perception of contact is indispensable in complex tasks like insertion and force-sensitive manipulation [4]. Prior studies have demonstrated that incorporating tactile feedback significantly enhances the flexibility and robustness of manipulation policies [5,6], particularly in unstructured or uncertain environments where visual signals alone are insufficient. Despite these benefits, most existing approaches rely on task-specific models trained on narrowly scoped datasets [7,8]. This results in limited generalization and prevents these methods from achieving the versatility of emerging general-purpose frameworks [9,10].

Recently, large language models have achieved significant breakthroughs in human-like reasoning [11], driving rapid progress in the development of vision-language-action (VLA) models [12,13]. Through cross-modal language grounding, VLA models have demonstrated superior generalization compared to traditional imitation learning approaches, particularly across diverse robot platforms and task settings. The π_0 and $\pi_{0.5}$ models [14] released by Physical Intelligence augment VLMs with a flow-matching action module to support continuous action output. Trained on multi-source heterogeneous data, these models mitigate the forgetting of general knowledge and further enhance generalization performance across varying robot platforms and observation modalities. Notably, existing VLA models are predominantly vision-centric [15], lacking tactile modalities that are essential for contact-rich manipulation tasks [16].

In recent months, integrating the tactile modality into existing VLA architectures has emerged as a research hotspot in the field. For instance, ForceVLA [17] leverages a Mixture-of-Experts (MoE) mechanism to directly fuse force-sensing features with those of Vision-Language Models (VLMs). Meanwhile, models such as Tactile-VLA [18] and TA-VLA [19] integrate tactile modality into the π_0 architecture, enabling high-precision robotic manipulation tasks including blackboard erasing and USB insertion, and thus achieving more accurate robotic operations compared with vision-only approaches. While these efforts represent valuable explorations, most of them simply embed low-dimensional tactile information into existing VLA frameworks, and they have not yet explored the potential of combining the tactile modality itself with large language models (LLMs) for action generation in scenarios where visual input is absent.

To address the aforementioned challenges, we construct a tactile-action instruction dataset specifically designed for fingertip-based peg-in-hole assembly tasks [20]. In addition, we introduce the TLA model, a novel fine-tuning paradigm for learning tactile-driven robotic skills. By leveraging cross-modal supervision to bridge language and tactile information, TLA enables robots to perform a variety of contact-rich insertion tasks under natural language guidance.

Our proposed Tactile-Language-Action framework consists of a high-fidelity tactile manipulation dataset and a language-conditioned tactile-action model, as illustrated in Figure 1. Building upon previous work [21], we construct a simulated dataset using a high-fidelity tactile simulator, comprising diverse assembly tasks with varying peg shapes and clearances. Specifically, 24 k tactile-language-action samples

are collected and reformulated into an instruction dataset. Furthermore, we present the TLA model to integrate language priors into a tactile-based control policy. To address the challenge of interpreting sequential tactile signals generated from a single contact event, TLA encodes the tactile sequence into a composite image. This fusion strategy leverages the spatial perception capabilities of VLMs, while mitigating their limitations in handling temporal data [22]. We conduct extensive experiments on both the TLA dataset and robotic assembly tasks. The results demonstrate that TLA significantly outperforms baseline methods in the TLA dataset and achieves notably higher success rates in robotic assembly tasks. Remarkably, despite being trained exclusively on data with a 2.0 mm clearance, TLA generalizes well to tighter clearances of 1.6 mm and 1.0 mm, achieving success rates exceeding 85%. The TLA model trained using simulation data could be deployed in the real world without finetuning. Our work makes the following key contributions:

- We propose a novel Tactile-Language-Action model that fuses sequential tactile feedback with natural language priors, enabling grounded action generation for contact-rich manipulation tasks.
- We construct the TLA dataset, a tactile-action instruction dataset that provides a valuable benchmark for studying language-conditioned tactile manipulation.
- We validate the proposed framework through both simulated and real-world peg-in-hole assembly experiments, demonstrating its practical applicability and strong generalization across varying clearances and shapes.

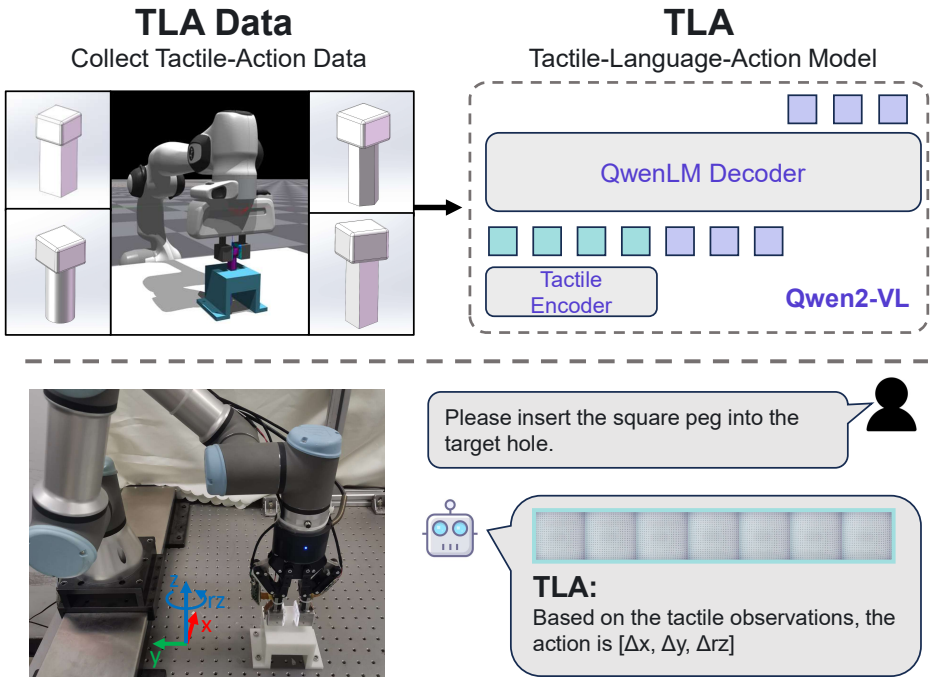

Figure 1. Overview of the proposed Tactile-Language-Action dataset and model. The TLA model is a 7B-parameter multimodal language model trained on our curated tactile-action dataset, specifically collected for insertion manipulation tasks. Through supervised fine-tuning, TLA learns to control a robot to perform various peg-in-hole assembly tasks.

2. Related work

2.1. VLM for robot manipulation

Recently, VLMs have demonstrated strong perception and reasoning capabilities. By treating actions as language tokens, RT-2 [10] introduces the Vision-Language-Action model based on a pre-trained VLM. However, RT-2 is closed-source. RoboFlamingo [23] and OpenVLA [12] adopt similar principles and are both open-sourced, making them widely adopted in the robotics community. GR-1 [24] and GR-2 [25] are pre-trained on Internet-scale video datasets to encode implicit physical and dynamic knowledge. These models are then fine-tuned on robot manipulation datasets. RDT-1B [26] and PI [27] further extend the VLA framework by incorporating diffusion-based objectives, showing improved action generation performance. Despite these advances, existing VLA models remain predominantly vision-based and are limited to relatively simple tasks such as pushing, pulling, or grasp-and-place. In contrast, our work investigates tactile sensing as an observation modality for language-conditioned action generation in contact-rich manipulation tasks.

2.2. Tactile-language model in robotics

Recent studies have investigated the integration of tactile information into language models [28,29]. Fu *et al.* [29] construct a texture-tactile dataset using ChatGPT and introduce the Tactile-Vision-Language Model to evaluate language models' capabilities in material understanding. However, the limited scale of their dataset restricts generalization to real-world scenarios. To address this limitation, Cheng *et al.* [30] propose Touch100k, a large-scale dataset for tactile-vision-based material classification. Despite these advances, prior work has not integrated tactile information into robotic manipulation tasks. Jones *et al.* [31] propose FuSe, which incorporates tactile sensing into a VLA model to learn a multimodal robotic policy. In contrast, our work focuses on establishing a direct connection between tactile perception and robot actions based on a pre-trained language model.

Recently, integrating the tactile modality into existing VLA architectures has emerged as a research hotspot in the field. Yu *et al.* [17] present ForceVLA, which leverages a MOE mechanism to fuse force modality features with visual-language features for subsequent action generation tasks. Huang *et al.* [18] develop Tactile-VLA, which integrates array-based tactile sensors with the π_0 architecture to enable high-precision manipulation tasks such as blackboard erasing and USB insertion. Additionally, Zhang *et al.* [19] propose TA-VLA and demonstrate that tactile signals contribute more substantially to action generation when incorporated at the decoder stage. The key distinction of our work from previous studies lies in our focus on the feasibility of combining high-dimensional, information-rich tactile signals with large language models for motion generation, while most prior studies are limited to low-dimensional tactile inputs.

2.3. Robotic peg-in-hole assembly

Peg-in-hole assembly is a representative contact-rich task commonly found in industrial applications, such as mechanical part fitting and precision manufacturing. Traditional methods typically rely on compliant control strategies [32], including force-based and impedance control. While effective in structured

environments, these approaches often require manual parameter tuning and struggle to generalize across varying geometries and tolerances. More recently, learning-based methods have achieved promising results by leveraging sensory feedback and data-driven policies [33]. However, such approaches often lack strong priors, leading to time-consuming training and limited generalization to unseen assembly conditions. In contrast to prior work, we incorporate language-based priors into tactile-action policy learning for the peg-in-hole task.

3. Dataset

In this section, we introduce a tactile–action instruction dataset for the peg-in-hole assembly task. Due to the scarcity of tactile–language–action instruction data, directly learning language-conditioned tactile policies remains challenging. In this setting, a robot equipped with GelStereo 2.0 visuotactile sensors [34] attempts to insert a peg into the hole based on natural language instructions. To enable efficient data collection, we construct a simulated peg-in-hole task in NVIDIA Isaac Gym and employ TacFlex [21] for visuotactile simulation, which ensures the generation of realistic and high-fidelity tactile observations throughout the entire insertion process.

The peg-in-hole task proceeds as follows. The robot gripper first grasps a peg with a specified shape and moves above the corresponding hole with a random 3-DOF misalignment along the x-axis, y-axis, and rotation around the z-axis (denoted as rz). The gripper then moves downward to attempt insertion. If a collision occurs during the downward motion, the attempt is considered a failure. The gripper then retracts and prepares for the next attempt. During each failed attempt, the sequence of tactile images recorded during contact is used to infer a corrective action $(\Delta x, \Delta y, \Delta rz)$ to adjust the peg pose. If no collision occurs and the gripper reaches the target depth, the insertion is considered successful. Each episode allows a maximum of 15 attempts; if insertion is not achieved within this limit, the task fails.

In simulation, we adopt a random insertion policy to collect data for the peg-in-hole task. For each insertion attempt, we record the sequence of tactile images along with the peg-to-hole pose errors. Based on these pose errors (e_x, e_y, e_{rz}), we generate action labels $(\Delta\hat{x}, \Delta\hat{y}, \Delta\hat{rz})$ used for training. The labels are computed as follows,

$$\Delta\hat{x} = \begin{cases} \min\left(0, \max\left(-\delta, -e_x + \frac{c}{2}\right)\right), & \text{if } e_x \geq 0, \\ \min\left(\delta, \max\left(0, -e_x - \frac{c}{2}\right)\right), & \text{if } e_x < 0, \end{cases} \tag{1}$$

$$\Delta\hat{y} = \begin{cases} \min\left(0, \max\left(-\delta, -e_y + \frac{c}{2}\right)\right), & \text{if } e_y \geq 0, \\ \min\left(\delta, \max\left(0, -e_y - \frac{c}{2}\right)\right), & \text{if } e_y < 0, \end{cases} \tag{2}$$

$$\Delta\hat{rz} = \min\left(1.5^\circ, \max\left(-1.5^\circ, -e_{rz}\right)\right), \tag{3}$$

where c denotes the assembly clearance. The three equations are designed to constrain the actions within a predefined range, thereby improving the stability of the learned policy. We empirically set the threshold δ to 1 mm. Throughout this study, the dataset is collected exclusively using pegs with an assembly clearance of 2.0 mm.

To facilitate model training, the collected interaction data is formatted into an instruction-based structure. Each dialogue round is enclosed by special tokens <lim_start|> and <lim_end|>, while tactile image inputs are wrapped between <|vision_start|> and <|vision_end|>. A natural language instruction specifies the task goal, describes the peg type, and outlines the insertion requirements. The corresponding robot actions during data collection are stored as ground-truth labels. An example of the TLA dataset format is given in Figure 2.

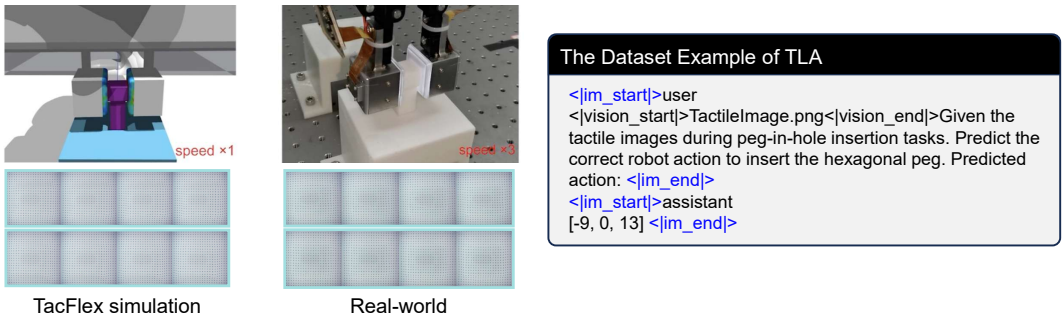

Figure 2. An example of TLA dataset.

4. Tactile-language-action model

This section presents the proposed Tactile-Language-Action model, as illustrated in Figure 3. TLA is built upon Qwen2-VL [35], which integrates a Vision Transformer (ViT) [36] for encoding visual inputs and the Qwen2 language model [37] for understanding multimodal information and generating responses. We first describe how sequential tactile information is encoded using ViT, followed by an explanation of how the language model predicts robotic actions from the fused tactile-language representation.

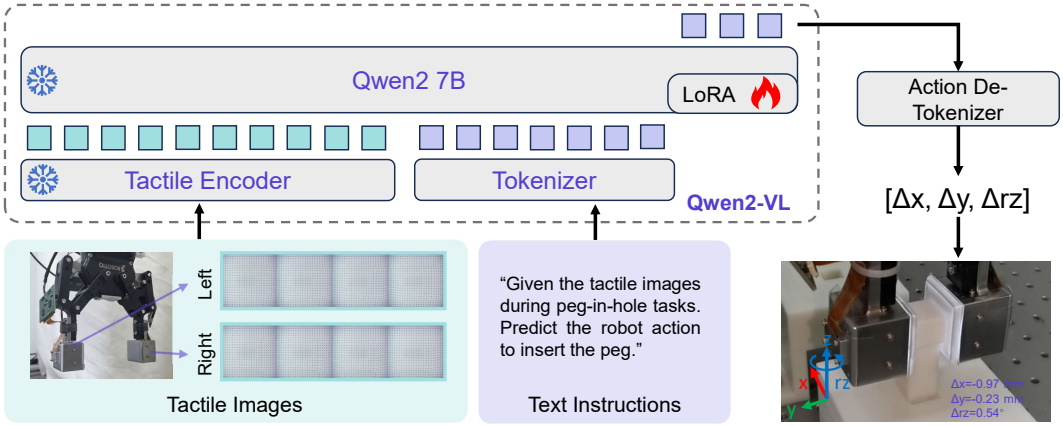

Figure 3. Architecture of the proposed Tactile-Language-Action model. Given a sequence of tactile images and a text instruction, TLA predicts 3-dimensional robotic actions. The model consists of two main modules: (1) Tactile Encoder extracts temporal tactile representations for the language model; (2) Language Model integrates tactile and textual information to infer the robot action.

4.1. Tactile encoder

As mentioned in the previous section, the tactile information from the visuotactile sensor is represented as images. During robotic manipulation, these tactile images vary continuously with changes in the contact

states between the gripper and the object. Therefore, the tactile encoder must process two temporally aligned sequences of tactile images to effectively capture contact dynamics over time.

The key challenge for the tactile encoder lies in modeling temporal variations from tactile image sequences, as most vision–language models are predominantly trained on images and exhibit limited temporal modeling capability. To address this issue, we synthesize two temporal sequences of tactile images into a single composite image, thereby converting temporal information into spatial structure for effective ViT-based feature extraction. Specifically, the input image set is denoted as $I = \{I_l^t, I_r^t \mid t = 0,1,2,3\}$, where I_l^t and I_r^t represent the tactile images from the left and right fingertips at time step t, respectively. These eight images are arranged into a 3×3 grid, with the remaining cell filled by a white placeholder image. The final composite image is resized to 616×616 pixels and used as the input to the model.

We adopt the ViT from Qwen2-VL [35] as the tactile encoder, as illustrated in Figure 3. The concatenated tactile image is first processed by the ViT to extract high-dimensional tactile features. To reduce the token dimension and enhance model efficiency, a Multi-Layer Perceptron (MLP) is applied to compress features within a local 2×2 patch region into a single token. Given the input tactile image $I \in \mathbb{R}^{616\times616}$ and a patch size of 14, the ViT produces $44 \times 44 = 1936$ patch tokens before MLP compression.

4.2. Action prediction with language model

This section describes how the Qwen2 language model is used to predict robotic actions, as illustrated in Figure 3. The input to the language model consists of tactile tokens and language tokens, which are obtained by encoding the multimodal inputs using the tactile encoder and a tokenizer, respectively. The backbone of our TLA model is Qwen2-7B [37], which is fine-tuned on the proposed TLA dataset to enable grounded action generation based on tactile and language inputs.

The encoding of continuous numerical values using discrete tokenizers can significantly impact model performance in number-related tasks [38]. Prior work [12] addresses this by overwriting infrequent tokens with dedicated "special tokens" and assigning them to discrete numerical bins. In contrast, we retain the original numerical encoding scheme to better leverage the numerical knowledge acquired during pre-training, ensuring its effective application in robotic manipulation tasks.

However, the tokenizer of Qwen2 encodes numerical values at the character level, which leads to redundant token sequences when representing decimal numbers. Since robotic action data often contains decimals, this redundancy increases the difficulty of model training. To alleviate this issue, we simplify the ground-truth representation by scaling all actions with predefined factors and rounding them to integers. The transformation is defined as: $A_{\text{gt}} = A_{\text{raw}} \cdot s$, where $A_{\text{gt}} \in \mathbb{R}^3$ denotes the scaled ground-truth action, $A_{\text{raw}} \in \mathbb{R}^3$ is the raw action, and $s \in \mathbb{R}^3$ represents the corresponding scale factors.

4.3. Training and inference

Previous studies have shown that VLMs achieve better performance when the visual encoder is kept frozen during training [35,39]. Following this insight, we freeze the parameters of the tactile encoder throughout the fine-tuning process. Additionally, we use Low-Rank Adaptation (LoRA) [40] to fine-tune

the Qwen2-7B language model. The model is trained using the ground-truth actions as supervision, and the training objective is formulated as a next-token prediction loss, defined as follows,

$$\mathscr{L}_{\text{NTP}} = -\sum_{t=1}^{T} p(y_t) \log P(y_t \mid y_{<t}, x), \tag{4}$$

where T denotes the length of the action sequence, and y_t represents the ground-truth token at step t. The notation $y_{<t}$ refers to the sequence of previously generated tokens, which are replaced with ground-truth tokens during training. The input x consists of tactile image tokens and text instruction tokens.

During inference, TLA sequentially predicts the probability distribution over robot actions conditioned on the tactile observations and instruction texts. The action sequence is generated using beam search until an end token is produced. Subsequently, the Action-De-Tokenizer maps the generated token probabilities to natural language text based on the vocabulary. It then converts the text into floating-point values that can be executed by the robot.

5. Experiment

In this section, we evaluate the effectiveness of the proposed TLA model for language-conditioned tactile learning in a fingertip peg-in-hole assembly task. The goal of our experiment is to answer the following questions:

- How does the TLA model perform on the dataset compared to the baseline methods?
- Does the TLA model effectively control the robot to accomplish various peg-in-hole assembly tasks with different shapes and clearances in the simulation environment?
- Can the TLA model deploy in the real world and control a real robot to perform various peg-in-hole assembly tasks?

5.1. Baseline and metrics

To answer the above questions, we compare the following baseline methods and ablation methods with the proposed TLA on various tasks.

- Behavior Cloning (BC) [41]: ResNet-50 [42] is used as the policy network, which takes the tactile image as input and outputs the corresponding robot action. The network is trained in a supervised manner.
- Diffusion Policy (DP) [43]: The diffusion policy [43] leverages a conditional denoising diffusion process to learn the peg-in-hole assembly policy.
- Single-Peg TLA (SP-TLA): The TLA model trained on the square peg insertion dataset.
- Multi-Peg TLA (MP-TLA): The TLA model trained on the square and triangular peg insertion dataset.

To evaluate different policies, we first assess the model's ability to generate effective actions using the Goal Convergence Rate (GCR), defined as the percentage of predicted actions that are correct in all three dimensions: x, y, and rz. In addition, we use the L1 distance to quantify the prediction error between the generated and ground-truth actions, which is calculated as follows:

$$L_1 = \frac{1}{m}\sum_{i=1}^{m}|y_i - \hat{y}_i|, \tag{5}$$

where m, y_i, $\hat{y}_i$ are the sample numbers in GCR, action prediction, and ground-truth, respectively. We compute the L1 distance for each action dimension separately and report the corresponding results.

5.2. Comparison with baseline methods

To address the first question, we conduct both single-peg and multi-peg experiments on the TLA dataset, comparing the performance of TLA with baseline methods.

Experimental Setup: For the single-peg setting, we use 8 k square peg insertion samples from the TLA dataset to compare the performance of TLA with baseline methods. Specifically, the dataset is split into 6 k samples for training and 2 k for testing. Each training sample consists of 8 tactile images, corresponding to the left and right fingertips over 4 time steps. The label is the robot action $(\Delta x, \Delta y, \Delta rz)$, representing the movement in the horizontal plane and the rotation around the z-axis. The SP-TLA model is trained for 10 epochs on 8 NVIDIA A6000 GPUs with a batch size of 128, a learning rate of 1e$-$3, and a training time of 1.6 hours. For the multi-peg setting, we use 16 k samples of square and triangular pegs for training. An additional 8 k samples are collected for evaluation, including 4 k from the seen pegs (square and triangular pegs) and 4 k from novel pegs (round and hexagonal pegs). The input, output, and evaluation remain consistent with the single-peg setting. The MP-TLA model is trained for 10 epochs on 8 NVIDIA A6000 GPUs with a batch size of 128, a learning rate of 1e$-$3, and a training time of 4.4 hours.

Results: The performance of different methods on the single-peg test set is reported in Table 1. The GCR results indicate that TLA produces a higher proportion of correct actions compared to the baselines. Moreover, the L1 distance results reveal that TLA predicts more accurate step lengths, particularly in the x-direction, where the L1 error is reduced by 78% compared to the second-best method. This superior L1 performance suggests that TLA can accomplish tasks with fewer operation steps, thereby demonstrating higher manipulation efficiency.

Table 1. Comparison of different methods on the single-peg test set.

Method	GCR (%)↑	L1 x (mm)↓	L1 y (mm)↓	L1 rz (deg)↓
BC [41]	10.4	0.803	0.302	0.205
DP [43]	8.5	0.370	0.382	0.568
SP-TLA (Ours)	**12.5**	**0.079**	**0.122**	**0.173**

The visualization results of model predictions are shown in Figure 4, which consists of three sub-figures representing pairwise combinations of action dimensions. In each sub-figure, the starting point of an arrow denotes the current pose deviation between the peg and the correct insertion position, while the arrow direction indicates the predicted action. Correct and incorrect predictions are colored in blue and orange, respectively. For the x-y pair, an action is considered correct if it reduces the position deviation in both dimensions. For the x-rz and y-rz pairs, an action is deemed correct if it reduces the deviation in at least one dimension. Experimental results show that TLA performs well in planar translation, effectively

guiding the peg toward the target hole. However, in translation-rotation visualizations, a larger number of incorrect predictions appear along the y-axis. We attribute this to the limitations of 2D tactile images: while our sensors capture tactile information parallel to the gripper (x-axis) effectively, they provide poor information for the perpendicular (y-axis) direction. This imbalance in feature representation increases the difficulty of accurate action prediction along the y-axis.

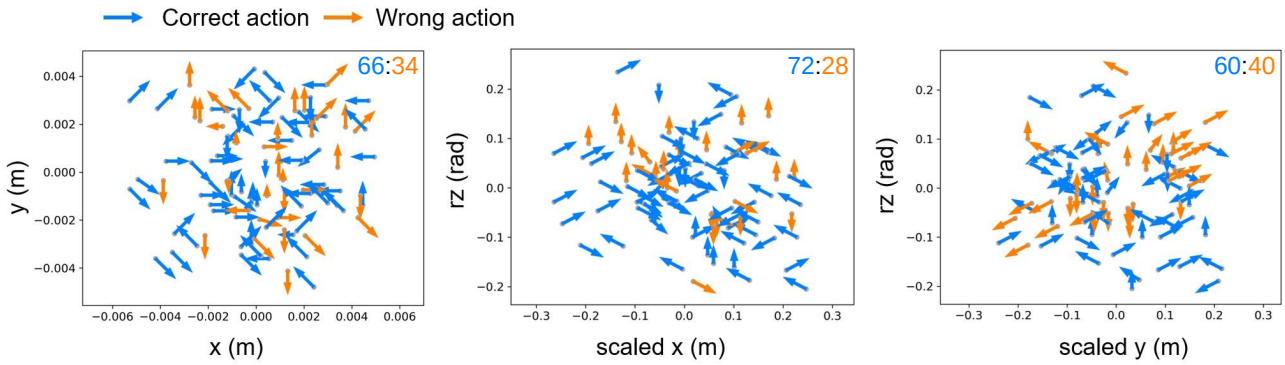

Figure 4. Visualization of action predictions on the single-peg test set. Three sub-figures illustrate pairwise combinations of action dimensions, where correct and incorrect actions are distinguished by colors. The term "Scaled" indicates that the raw data has been multiplied by a factor of 50 for visualization purposes. The results demonstrate that TLA accurately predicts actions across various deviations.

The comparison of different methods on the multi-peg test set is presented in Table 2. In the in-distribution (ID) setting, TLA achieves the lowest L1 error on seen peg shapes, indicating that its predicted actions are not only correct but also exhibit more precise and consistent step lengths during execution. In the out-of-distribution (OOD) setting, TLA maintains performance comparable to that on the ID set, demonstrating strong generalization to unseen peg shapes. Notably, TLA-OOD does not suffer from significant degradation compared to TLA-ID. This stands in stark contrast to traditional imitation learning baselines, which exhibit pronounced performance drops under distribution shifts. These results highlight the robustness of TLA to variations in peg shape and its effectiveness in transferring learned manipulation skills to novel configurations.

Table 2. Comparison of different methods on the multi-peg test set.

Method	ID				OOD			
	GCR↑ (%)	L1 x ↓ (mm)	L1 y ↓ (mm)	L1 rz ↓ (deg)	GCR↑ (%)	L1 x ↓ (mm)	L1 y ↓ (mm)	L1 rz ↓ (deg)
BC [41]	**18.4**	0.260	0.655	0.186	15.2	0.286	0.722	0.246
DP [43]	7.4	0.371	0.348	0.480	8.0	0.386	0.369	0.544
MP-TLA (Ours)	**18.4**	**0.102**	**0.114**	**0.135**	**16.5**	**0.121**	**0.102**	**0.184**

5.3. Robotic insertion in simulation

In this subsection, we evaluate the manipulation performance of different models on simulated robotic insertion tasks to answer the second research question.

Experimental Setup: We evaluate the manipulation performance of different models on two peg-in-hole assembly tasks. First, we assess model performance under varying assembly clearances by conducting square peg insertion experiments with clearances of 2.0 mm, 1.6 mm, and 1.0 mm. Second, we examine generalization across different geometries by testing the models on square, triangular, round, and hexagonal peg insertion tasks, with all clearances fixed at 2.0 mm. Each task is repeated 50 times, and the average success rate is reported as the final result.

At the beginning of each episode, the robot has already grasped the peg with its gripper, and the end-effector is randomly positioned near the target hole. The robot then performs an initial insertion attempt and collects tactile observations. Based on these tactile inputs, TLA or other baseline models predict the next action to guide the subsequent insertion. This process repeats until the insertion succeeds or the maximum number of 15 attempts is reached. To evaluate performance, we report both the overall success rate and the average number of steps in successful episodes.

Results: The quantitative comparison results of different methods across various assembly clearances are presented in Table 3. Compared with baseline methods, TLA achieves a significantly higher success rate and better manipulation efficiency, outperforming the second-best method by a margin of 50% in success rate. Even under the most challenging 1.0 mm clearance condition, TLA consistently surpasses the baselines, demonstrating strong generalization capability across different tolerances. While BC and DP achieve GCR scores comparable to TLA, their L1 performance is considerably worse. This discrepancy suggests that these models struggle to predict accurate step lengths, preventing them from reaching the target hole within the allowed number of attempts. In some cases, the inaccurate predictions result in repetitive behaviors around the hole, ultimately leading to insertion failure.

Table 3. Comparison of different methods in insertion tasks with various clearances.

Method	2.0 mm		1.6 mm		1.0 mm		Total	
	Suc↑	Step↓	Suc↑	Step↓	Suc↑	Step↓	Suc↑	Step↓
BC [41]	44	2.60	32	4.00	18	4.44	31	3.68
DP [43]	58	**2.45**	43	4.35	20	4.70	40	3.83
SP-TLA	**96**	3.15	**94**	3.77	74	4.37	88	3.76
MP-TLA	94	3.04	86	**3.30**	**90**	**4.35**	**90**	**3.56**

The quantitative comparison results of different methods on various peg types are presented in Table 4. Both TLA variants consistently outperform the baseline methods in terms of success rate and the number of steps, demonstrating strong generalization across different peg geometries. Notably, the MP-TLA model achieves better performance than the SP-TLA model in the OOD setting. This suggests that training on a diverse set of peg geometries further improves the generalization capability of the TLA model.

Table 4. Comparison of different methods on insertion tasks with various peg types.

Method	ID				OOD			
	Square		Triangle		Round		Hexagon	
	Suc↑	Step↓	Suc↑	Step↓	Suc↑	Step↓	Suc↑	Step↓
BC [41]	60	3.37	84	2.67	62	2.84	66	**2.30**
DP [43]	60	3.60	60	3.40	56	2.89	54	2.37
SP-TLA	**96**	3.15	76	2.82	92	2.80	90	2.78
MP-TLA	94	**3.04**	**92**	**1.80**	**94**	**2.66**	**96**	3.00

Visualization: The visualization result of TLA controlling the robot to successfully insert the peg is shown in Figure 5. Initially, the robot fails to complete the insertion with the first action. Based on the tactile observations collected during the failed attempt, TLA predicts adjusted actions to guide the robot. After two rounds of correction, the robot successfully approaches and inserts the peg.

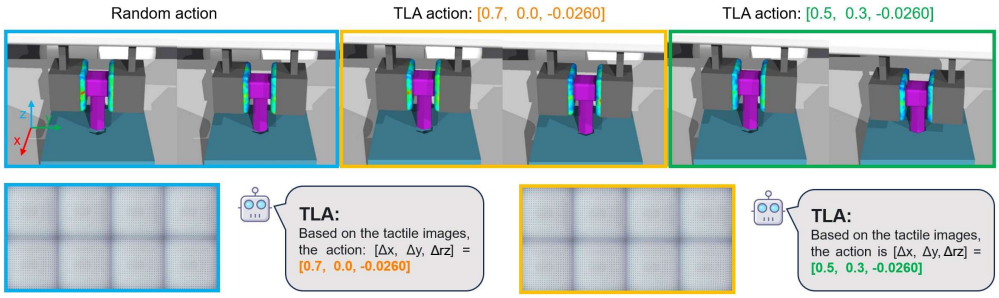

Figure 5. Snapshots of the robot successfully inserting a hexagonal peg into the target hole using the proposed TLA model. The sequence illustrates the robot's adaptive behavior under contact-rich conditions.

The visualization of a failure case is shown in Figure 6, where TLA fails to control the robot to insert the triangular peg into the hole. This failure is primarily attributed to the asymmetric tolerance characteristics of the triangular hole. Specifically, for an assembly clearance of d, the allowable deviation along the x-axis is approximately $3d$, whereas that along the y-axis is only about $2.3d$. In addition, since the current tactile information is represented as 2D images, the quality of feature representation in the direction perpendicular to the gripper (*i.e.*, the y-axis) is weaker than that in the x directions. This imbalance makes it more difficult for the model to interpret contact information along the y-axis.

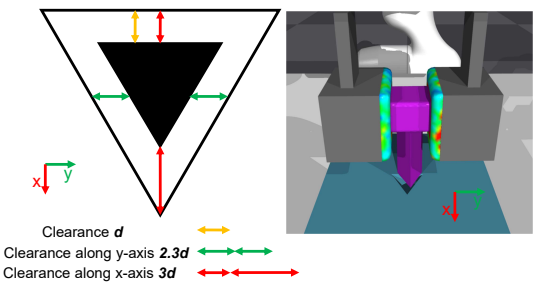

Figure 6. Visualization of a failure case. The TLA model fails to insert the triangular peg due to the direction-dependent clearance of the triangular hole, where the clearance in the y-direction is smaller than that in the x-direction. The sensor's limited feature representation along the y-axis further restricts the model's performance on the triangular peg insertion task.

5.4. Real-world robotic insertion

To answer the third question, we deploy TLA in real-world environments to evaluate its sim-to-real generalization, testing how well the policy learned in simulation transfers under real-world uncertainties.

Experimental Setup: We create the peg-in-hole task using a UR3 robot in the real world. Two GelStereo 2.0 sensors [34] are mounted onto the Robotiq-85 gripper. We evaluate the insertion performance of TLA using pegs with different clearances and geometries. The execution process of the task remains consistent with that in the simulation. Each task is repeated 20 times to compute the success rate. We adopt the MP-TLA model to control the real-world robot for performing the peg-in-hole assembly task.

Results: The quantitative results for peg-in-hole tasks with various clearances are demonstrated in Table 5, indicating that the TLA model trained entirely on simulated data can be directly deployed on the real-world robot. Compared to the simulated tasks, the success rate in the real-world scenario shows a decline. Specifically, with a clearance of 2.0 mm, the success rate is reduced by approximately 15%. While simulation results indicate a relatively consistent success rate across varying clearances, the success rate in real-world conditions deteriorates markedly for tasks involving tighter insertions. This discrepancy is likely due to complex physical phenomena—such as friction and surface viscosity on the sensor—that are not accurately modeled in the simulation. The real-world performance might be improved by developing advanced Sim2Real transfer methods or hybrid training incorporating both simulated and real data.

Table 5. Success rate of insertion tasks with various clearances in the real world (square peg).

Method	2.0 mm		1.6 mm		1.0 mm		Total	
	Suc↑	Step↓	Suc↑	Step↓	Suc↑	Step↓	Suc↑	Step↓
MP-TLA	80	2.56	70	1.71	55	2.27	68	2.18

The results for peg-in-hole tasks with different geometries are presented in Table 6, demonstrating an overall success rate of approximately 80%. The success rate for round pegs is relatively high, likely due to their insensitivity to rotational misalignment during insertion. The strong performance in real-world peg-in-hole tasks further indicates the TLA model's robust generalization to out-of-domain settings, including variations in peg-hole geometry and the sim-to-real gap. These results underscore the TLA model's great potential for practical, generalizable assembly applications.

Table 6. Success rate of insertion tasks with various peg types in the real world (2.0 mm clearance).

Method	ID				OOD			
	Square		Triangle		Round		Hexagon	
	Suc↑	Step↓	Suc↑	Step↓	Suc↑	Step↓	Suc↑	Step↓
MP-TLA	80	2.56	80	2.63	95	1.79	75	2.47

Visualization: The real-world insertion processes for different pegs are illustrated in Figure 7. The robot actions generated by the TLA model reduce the posture deviation between the peg and the hole. The task is successfully completed after several rounds.

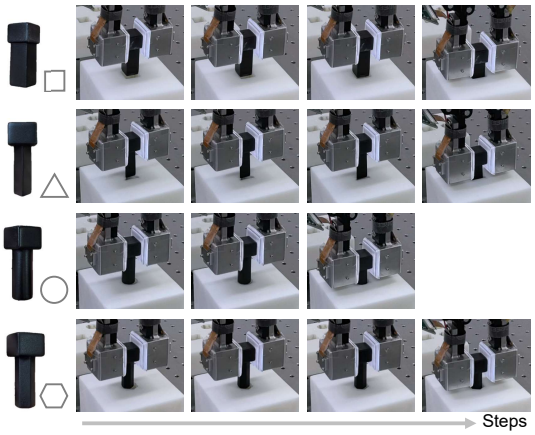

Figure 7. Snapshots of peg-in-hole tasks using the proposed TLA model in the real world.

6. Discussion and limitations

In this work, we propose TLA, a Tactile-Language-Action model and dataset designed for peg-in-hole assembly. Specifically, we present the TLA model that fuses sequential tactile feedback with natural language priors, enabling grounded action generation under complex contact conditions. To support research in this domain, we construct a large-scale tactile-action instruction dataset, serving as a valuable benchmark for language-conditioned tactile manipulation. Extensive experiments demonstrate that TLA not only outperforms traditional imitation learning methods, but also exhibits strong generalization to unseen peg geometries and varying assembly tolerances.

Despite the promising results, TLA has several limitations. A notable limitation is that the model does not explicitly capture the temporal dynamics of tactile signals. Instead, temporal information is encoded via spatial arrangements, which may not fully exploit the sequential characteristics inherent in tactile data. Future work will explore more effective representations and learning paradigms for sequential tactile perception. One promising direction is to perform cross-modal alignment between language and tactile signals [44], combined with fine-grained multi-modal alignment techniques [45], thereby enabling large language models to receive more structured and semantically grounded tactile inputs. In addition, the choice of tactile signal representation remains an open question. Different tactile modes—such as 2D tactile images, contact depth maps, and 3D tactile point clouds—offer distinct advantages. Exploring their integration with vision-language models for tactile skill learning could further enhance model capabilities. Finally, more advanced action de-tokenization mechanisms could be explored. Adopting flow-matching techniques [14] could better support continuous robotic control and improve action precision.

Data availability statement

The data or datasets that support the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgments

This work was supported in part by the National Key R&D Program of China under Grant 2025ZD1606200, in part by the National Natural Science Foundation of China under U24A20282, 62303455, 62373039, U23B2038, 62273342, and 62122087, in part by Beijing Natural Science Foundation under Grant L233006, L253006.

Authors' contribution

Peng Hao: conceptualization, methodology, data curation, validation, formal analysis, visualization and writing—original draft. Chaofan Zhang: conceptualization, data curation, validation, formal analysis, visualization and writing—original draft. Dingzhe Li: methodology, data curation and validation. Xiaoge Cao: validation and formal analysis. Xiaoshuai Hao: writing—review and editing. Shaowei Cui: writing—review and editing, project administration and funding acquisition. Shuo Wang: writing—review and editing and funding acquisition. All authors have read and agreed to the published version of the manuscript.

Conflicts of interests

The authors declare no conflict of interest.

References

[1] Billard A, Kragic D. Trends and challenges in robot manipulation. *Science* 2019, 364(6446):eaat8414.

[2] Cui J, Trinkle J. Toward next-generation learned robot manipulation. *Sci. Rob.* 2021, 6(54):eabd9461.

[3] Calandra R, Owens A, Jayaraman D, Lin J, Yuan W, *et al.* More than a feeling: learning to grasp and regrasp using vision and touch. *IEEE Rob. Autom. Lett.* 2018, 3(4):3300–3307.

[4] Li Q, Kroemer O, Su Z, Veiga FF, Kaboli M, *et al.* A review of tactile information: perception and action through touch. *IEEE Trans. Rob.* 2020, 36(6):1619–1634.

[5] Wang C, Wang S, Romero B, Veiga F, Adelson E. Swingbot: learning physical features from in-hand tactile exploration for dynamic swing-up manipulation. In *International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, USA, October 24–January 24, 2020, pp. 5633–5640.

[6] Qi H, Yi B, Suresh S, Lambeta M, Ma Y, *et al.* General in-hand object rotation with vision and touch. In *Proceedings of The 7th Conference on Robot Learning*, Atlanta, USA, November 6–9, 2023, pp. 2549–2564.

[7] Lee MA, Zhu Y, Zachares P, Tan M, Srinivasan K, *et al.* Making sense of vision and touch: learning multimodal representations for contact-rich tasks. *IEEE Trans. Rob.* 2020, 36(3):582–596.

[8] Dong S, Jha DK, Romeres D, Kim S, Nikovski D, *et al.* Tactile-RL for insertion: generalization to objects of unknown geometry. In *IEEE International Conference on Robotics and Automation (ICRA)*, Xi'an, China, May 30–June 5, 2021, pp. 6437–6443.

[9] Team OM, Ghosh D, Walke H, Pertsch K, Black K, *et al.* Octo: an open-source generalist robot policy. *arXiv* 2024, arXiv:2405.12213.

[10] Brohan A, Brown N, Carbajal J, Chebotar Y, Chen X, *et al.* Rt-2: vision-language-action models transfer web knowledge to robotic control. *arXiv* 2023, arXiv:2307.15818.

[11] Naveed H, Khan AU, Qiu S, Saqib M, Anwar S, *et al.* A comprehensive overview of large language models. *arXiv* 2023, arXiv:2307.06435.

[12] Kim MJ, Pertsch K, Karamcheti S, Xiao T, Balakrishna A, *et al.* Openvla: an open-source vision-language-action model. *arXiv* 2024, arXiv:2406.09246.

[13] Wen J, Zhu Y, Li J, Zhu M, Wu K, *et al.* Tinyvla: towards fast, data-efficient vision-language-action models for robotic manipulation. *arXiv* 2024, arXiv:2409.12514.

[14] Black K, Brown N, Darpinian J, Dhabalia K, Driess D, *et al.* $\pi_{0.5}$: a vision-language-action model with open-world generalization. In *9th Annual Conference on Robot Learning*, Seoul, Republic of Korea, September 27–30, 2025.

[15] O'Neill A, Rehman A, Maddukuri A, Gupta A, Padalkar A, *et al.* Open X-embodiment: robotic learning datasets and RT-X models: open X-embodiment collaboration[0]. In *IEEE International Conference on Robotics and Automation (ICRA)*, Yokohama, Japan, May 13–17, 2024, pp. 6892–6903.

[16] Zhou H, Yao X, Meng Y, Sun S, Bing Z, *et al.* Language-conditioned learning for robotic manipulation: a survey. *arXiv* 2023, arXiv:2312.10807.

[17] Yu J, Liu H, Yu Q, Ren J, Hao C, *et al.* ForceVLA: enhancing VLA models with a force-aware MoE for contact-rich manipulation. *arXiv* 2025, arXiv:2505.22159.

[18] Huang J, Wang S, Lin F, Hu Y, Wen C, *et al.* Tactile-VLA: unlocking vision-language-action model's physical knowledge for tactile generalization. *arXiv* 2025, arXiv:2507.09160.

[19] Zhang Z, Xu H, Yang Z, Yue C, Lin Z, *et al.* TA-VLA: elucidating the design space of torque-aware vision-language-action models. *arXiv* 2025, arXiv:2509.07962.

[20] Chen W, Xu J, Xiang F, Yuan X, Su H, *et al.* General-purpose sim2real protocol for learning contact-rich manipulation with marker-based visuotactile sensors. *IEEE Trans. Rob.* 2024, 40:1509–1526.

[21] Zhang C, Cui S, Hu J, Jiang T, Zhang T, *et al.* TacFlex: multi-mode tactile imprints simulation for visuotactile sensors with coating patterns. *IEEE Trans. Rob.* 2025, 41:3965–3985.

[22] Bordes F, Pang RY, Ajay A, Li AC, Bardes A, *et al.* An introduction to vision-language modeling. *arXiv* 2024, arXiv:2405.17247.

[23] Li X, Liu M, Zhang H, Yu C, Xu J, *et al.* Vision-language foundation models as effective robot imitators. *arXiv* 2023, arXiv:2311.01378.

[24] Wu H, Jing Y, Cheang C, Chen G, Xu J, *et al.* Unleashing large-scale video generative pre-training for visual robot manipulation. *arXiv* 2023, arXiv:2312.13139.

[25] Cheang CL, Chen G, Jing Y, Kong T, Li H, *et al.* Gr-2: a generative video-language-action model with web-scale knowledge for robot manipulation. *arXiv* 2024, arXiv:2410.06158.

[26] Liu S, Wu L, Li B, Tan H, Chen H, *et al.* Rdt-1b: a diffusion foundation model for bimanual manipulation. *arXiv* 2024, arXiv:2410.07864.

[27] Black K, Brown N, Driess D, Esmail A, Equi M, *et al.* π_0: a vision-language-action flow model for general robot control. *arXiv* 2024, arXiv:2410.24164.

[28] Yang F, Feng C, Chen Z, Park H, Wang D, *et al.* Binding touch to everything: learning unified multimodal tactile representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision*

and Pattern Recognition, Seattle, USA, June 17–21, 2024, pp. 26340–26353.

[29] Fu L, Datta G, Huang H, Panitch WCH, Drake J, *et al.* A touch, vision, and language dataset for multimodal alignment. *arXiv* 2024, arXiv:2402.13232.

[30] Cheng N, Guan C, Gao J, Wang W, Li Y, *et al.* Touch100k: a large-scale touch-language-vision dataset for touch-centric multimodal representation. *arXiv* 2024, arXiv:2406.03813.

[31] Jones J, Mees O, Sferrazza C, Stachowicz K, Abbeel P, *et al.* Beyond sight: finetuning generalist robot policies with heterogeneous sensors via language grounding. *arXiv* 2025, arXiv:2501.04693.

[32] Kang SC, Hwang YK, Kim MS, Lee CW, Lee KI. A compliant motion control for insertion of complex shaped objects using contact. In *Proceedings of International Conference on Robotics and Automation*, Albuquerque, USA, April 25, 1997, pp. 841–846.

[33] Hao P, Lu T, Cui S, Wei J, Cai Y, *et al.* Meta-residual policy learning: zero-trial robot skill adaptation via knowledge fusion. *IEEE Rob. Autom. Lett.* 2022, 7(2):3656–3663.

[34] Zhang C, Cui S, Wang S, Hu J, Cai Y, *et al.* Gelstereo 2.0: an improved gelstereo sensor with multimedium refractive stereo calibration. *IEEE Trans. Ind. Electron.* 2023, 71(7):7452–7462.

[35] Wang P, Bai S, Tan S, Wang S, Fan Z, *et al.* Qwen2-vl: enhancing vision-language model's perception of the world at any resolution. *arXiv* 2024, arXiv:2409.12191.

[36] Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, *et al.* An image is worth 16x16 words: transformers for image recognition at scale. *arXiv* 2020, arXiv:2010.11929.

[37] Yang A, Yang B, Hui B, Zheng B, Yu B, *et al.* Qwen2 technical report. *arXiv* 2024, arXiv:2407.10671.

[38] Qian L, Li J, Wu Y, Ye Y, Fei H, *et al.* Momentor: advancing video large language model with fine-grained temporal reasoning. *arXiv* 2024, arXiv:2402.11435.

[39] Karamcheti S, Nair S, Balakrishna A, Liang P, Kollar T, *et al.* Prismatic vlms: investigating the design space of visually-conditioned language models. In *Forty-first International Conference on Machine Learning*, Vienna, Austria, July 24–27, 2024.

[40] Hu EJ, Shen Y, Wallis P, Allen-Zhu Z, Li Y, *et al.* Lora: low-rank adaptation of large language models. *ICLR* 2022, 1(2):3.

[41] Torabi F, Warnell G, Stone P. Behavioral cloning from observation. *arXiv* 2018, arXiv:1805.01954.

[42] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, Las Vegas, USA, June 27–30, 2016, pp. 770–778.

[43] Chi C, Xu Z, Feng S, Cousineau E, Du Y, *et al.* Diffusion policy: visuomotor policy learning via action diffusion. *Int. J. Rob. Res.* 2025, 44(10,11):1684–1704.

[44] Ma W, Cao X, Zhang Y, Zhang C, Yang S, *et al.* CLTP: contrastive language-tactile pre-training for 3D contact geometry understanding. *arXiv* 2025, arXiv:2505.08194.

[45] Tschannen M, Gritsenko A, Wang X, Naeem MF, Alabdulmohsin I, *et al.* Siglip 2: multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv* 2025, arXiv:2502.14786.