IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

TactileAloha: Learning Bimanual Manipulation With Tactile Sensing

Ningquan Gu, Kazuhiro Kosuge, *Fellow, IEEE*, and Mitsuhiro Hayashibe, *Senior Member, IEEE*

***Abstract*—Tactile texture is vital for robotic manipulation but challenging for camera vision-based observation. To address this, we propose TactileAloha, an integrated tactile-vision robotic system built upon Aloha, with a tactile sensor mounted on the gripper to capture fine-grained texture information and support real-time visualization during teleoperation, facilitating efficient data collection and manipulation. Using data collected from our integrated system, we encode tactile signals with a pre-trained ResNet and fuse them with visual and proprioceptive features. The combined observations are processed by a transformer-based policy with action chunking to predict future actions. We use a weighted loss function during training to emphasize near-future actions, and employ an improved temporal aggregation scheme at deployment to enhance action precision. Experimentally, we introduce two bimanual tasks: zip tie insertion and Velcro fastening, both requiring tactile sensing to perceive the object texture and align two object orientations by two hands. Our proposed method adaptively changes the generated manipulation sequence itself based on tactile sensing in a systematic manner. Results show that our system, leveraging tactile information, can handle texture-related tasks that camera vision-based methods fail to address. Moreover, our method achieves an average relative improvement of approximately 11.0% compared to state-of-the-art method with tactile input, demonstrating its performance.**

***Index Terms*—Imitation learning, bimanual manipulation, hardware-software integration in robotics.**

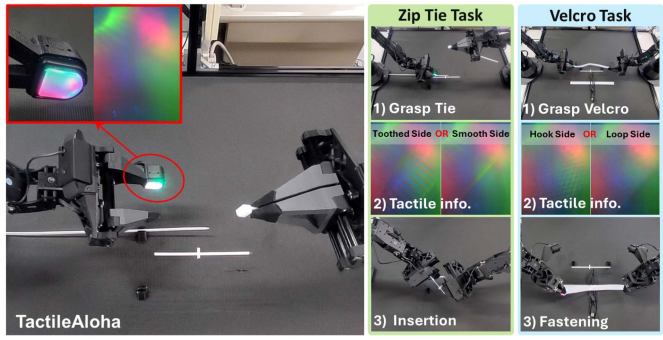

Fig. 1. Robotic Manipulation with Tactile Sensing. TactileAloha: A GelSight sensor is mounted on the Aloha robot system using a 3D-printed bracket to perceive the texture of objects during manipulation and enhance the robot's manipulation capabilities. We introduce two tasks: zip tie insertion and Velcro fastening. The zip tie task involves inserting a randomly placed zip tie into the head of another, while the Velcro task involves attaching a randomly placed Velcro strip to another fixed Velcro strip. In both tasks, the facing of the objects, i.e., their front and back sides, is crucial for successful manipulation. (1) The two robotic arms grasp the objects; (2) The GelSight sensor observes the orientation of the object to be manipulated, as different orientations correspond to different manipulation strategies; (3) The robot arms execute the task, one end of a zip tie is inserted into the head of another, while the correct side of the Velcro strip is aligned and fastened with the other Velcro strip.

I. Introduction

ROBOTIC manipulation has seen significant advancements [1], [2], [3], [4], [5] in recent years, driven by improvements in vision-based learning approaches. While vision provides rich spatial and environmental context, it alone is often insufficient for precise manipulation, particularly in tasks requiring fine contact, e.g., distinguishing between the toothed and smooth sides of a grasped zip tie, the hook and loop sides of Velcro, or differentiating between different cloth textures. In human dexterity, touch plays a crucial role in recognizing object properties, and ensuring stable interactions with the environment. Similarly, integrating tactile sensing into robotic systems has the potential to enhance perception, improve robustness, and expand capabilities.

Recently, works [6], [7], [8] have started to incorporate tactile information into robotic manipulation and have demonstrated success. However, these sensors mainly focus on the force of touch [7] rather than the texture of objects or are mainly applied to object detection [6], followed by heuristic-based manipulation, which has limited generalization.

In this work, we introduce the TactileAloha system by mounting a GelSight tactile sensor [10] onto the gripper of the Aloha platform [2], as shown in Fig. 1. TactileAloha captures fine-grained contact features, such as surface texture and object orientation, to support precise manipulation. For example, in the zip tie insertion task, it helps perceive the orientation of the toothed side, which determines the correct insertion trajectory.

Received 3 March 2025; accepted 18 June 2025. Date of publication 2 July 2025; date of current version 10 July 2025. This article was recommended for publication by Associate Editor M. Burke and Editor A. Faust upon evaluation of the reviewers' comments. The work of Ningquan Gu was supported by GP-Mech International Joint Graduate Program, Tohoku University. This work was supported in part by the Innovation and Technology Commission of the HKSAR Government through the InnoHK initiative and in part by the JC STEM Lab of Robotics for Soft Materials through the Hong Kong Jockey Club Charities Trust. *(Corresponding author: Mitsuhiro Hayashibe.)*

Ningquan Gu and Mitsuhiro Hayashibe are with the Neuro-Robotics Laboratory, Department of Robotics, Graduate School of Engineering, Tohoku University, Sendai 980-8577, Japan (e-mail: hayashibe@tohoku.ac.jp).

Kazuhiro Kosuge is with the Department of Electrical and Electronic Engineering, Faculty of Engineering, University of Hong Kong, Hong Kong, also with the Centre for Transformative Garment Production, Hong Kong Science Park, Hong Kong, and also with the JC STEM Lab of Robotics for Soft Materials, Department of Electrical and Electronic Engineering, Faculty of Engineering, University of Hong Kong, Hong Kong.

The project website is: https://guningquan.github.io/TactileAloha/.

This article has supplementary downloadable material available at https://doi.org/10.1109/LRA.2025.3585396, provided by the authors.

Digital Object Identifier 10.1109/LRA.2025.3585396

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

We utilize the ResNet [11] model with pre-trained weights to encode tactile information, which is then combined with encoded camera images and robot joint values, and then fed into the end2end ACT [2] imitation learning method that uses the action chunking technique. To enhance prediction accuracy, we use a weighted loss function that emphasizes earlier steps within each predicted action chunk by applying exponentially decaying weights across the chunk length.

To ensure smooth and precise motion during manipulation, we adopt an improved temporal ensembling strategy, Temporal Proximity Ensembling. At each timestep, the policy generates an action chunk, i.e., a sequence of predicted future actions. To execute the current action, we aggregate the predicted actions at the same relative index across a temporal span of recent chunks, assigning higher weights.

To showcase the integrated system and algorithm, we design two tasks: the zip tie insertion task and the Velcro fastening task. The results demonstrate that our system can perceive the texture of the grasped object through tactile sensing and use it to guide downstream actions, which are capabilities that vision-only systems lack. We achieve an average of approximately 11.0% relative improvement in the success rate compared to state-of-the-art (SOTA) in the two manipulation tasks. Moreover, we conduct an ablation study on our method and analyze the contribution of each improved component.

In summary, the main contributions of this study include:

1) We propose TactileAloha, an open-source robotic system that extends the Aloha platform by integrating a GelSight tactile sensor into the gripper to provide tactile information during manipulation.
2) Leveraging the data collected with our proposed integrated system, we explore the ACT-based method to incorporate tactile information. Furthermore, we use a weighted loss function and improved temporal ensembling to enhance manipulation performance.
3) Experiments demonstrate the performance of our approach, while ablation studies highlight the contribution of each improved component.

II. Related Work

A. Tactile Sensing for Robotic Manipulation

Tactile information plays a key role in robotic manipulation. Recent works [6], [7], [8], [12] have begun exploring the use of tactile sensors [10], [13] to overcome the limitations of solo vision sensor-based methods, which often fail to address certain challenges. For example, Tirumala et al. [7] explored tactile feedback to grasp a specific number of cloth layers. They used a sensor based on magnetic sensing [14] to detect normal and shear forces, which were computed from distortions in the magnetic field during the grasping of different numbers of cloth layers. However, their method cannot be used to recognize the texture of objects. Sunil et al. [6] used GelSight [10] to complete a cloth edge sliding action. They used a supervised learning method to recognize the state of the cloth edge grasped by the GelSight during sliding. However, their method requires manual labeling and uses a heuristic pipeline based on cloth texture detection, which often fails in complex scenarios. In contrast, we utilize a CNN to decode the grasped texture information without any human labeling, while the end2end imitation learning we employ can adaptively generate manipulation trajectories based on tactile sensing.

B. Imitation Learning for Robotics

Imitation learning (IL) [15] allows robots to learn directly from expert demonstrations. A widely used approach is Behavior Cloning (BC) [16], which maps observations to actions in an end2end manner. Recent studies have advanced IL in terms of policy architectures, platforms, and data processing etc. For example, Zhao et al. [2] introduced Aloha, a low-cost robotic platform, along with Action Chunking with Transformers (ACT), which predicts a sequence of future actions as a chunk. Fu et al. [17] extended this system to mobile settings using a co-training framework. However, both approaches rely solely on visual inputs and struggle with texture-sensitive tasks. In addition, ACT equally weights all predicted actions during training, despite the greater importance of recent ones in manipulation. In contrast, we extend Aloha with tactile sensing to enable multimodal manipulation and use a weighted loss function that emphasizes recent actions to improve precision. The tasks we design include multi-stage or multi-strategy manipulation tasks, such as selecting different strategies based on zip tie or Velcro orientation, or executing sequential subtasks in long-horizon scenarios. To address such challenges, Burke et al. [18] proposed Switching Density Networks (SDNs), which leverage a categorical reparameterization mechanism to extend model-based hierarchical behavior cloning methods [19]. Their method models manipulation behavior as a composition of sub-controllers and benefits from explicit structure in the form of mode-based control laws. In contrast, we adopt a model-free, end2end imitation learning framework that learns a continuous policy from raw multimodal inputs, implicitly discovering multi-phase manipulation strategies and generating goal-directed behavior in a systematic manner.

III. TactileAloha: Description of the Robot System

To address the challenge of tactile sensing in robotic manipulation, we introduce TactileAloha, built upon Aloha [2]. The key upgrades are illustrated in Fig. 3. We install a tactile sensor, GelSight, which is launched by ROS and used for data collection and control. The hardware installation model and code can be found in the project documentation. For the camera setup, we use three Logitech C922n webcams, positioned for top, left-arm, and right-arm views. Furthermore, we upgrade the top-view camera installation method. The reason is that we found Aloha's mounting method problematic: during robot manipulation, the robot's movement causes vibrations in the metal frame, leading to significant shaking of the top-view camera, which reduces manipulation performance. Therefore, we mount the top-view camera on a fixed boom, which prevents movement-induced vibrations from affecting the robot. Fig. 4 shows the complete TactileAloha system along with several other upgrades. The upgrade items for TactileAloha cost approximately $450. During dataset collection, the teleoperator performed the task by visually observing the scene and monitoring sensor outputs

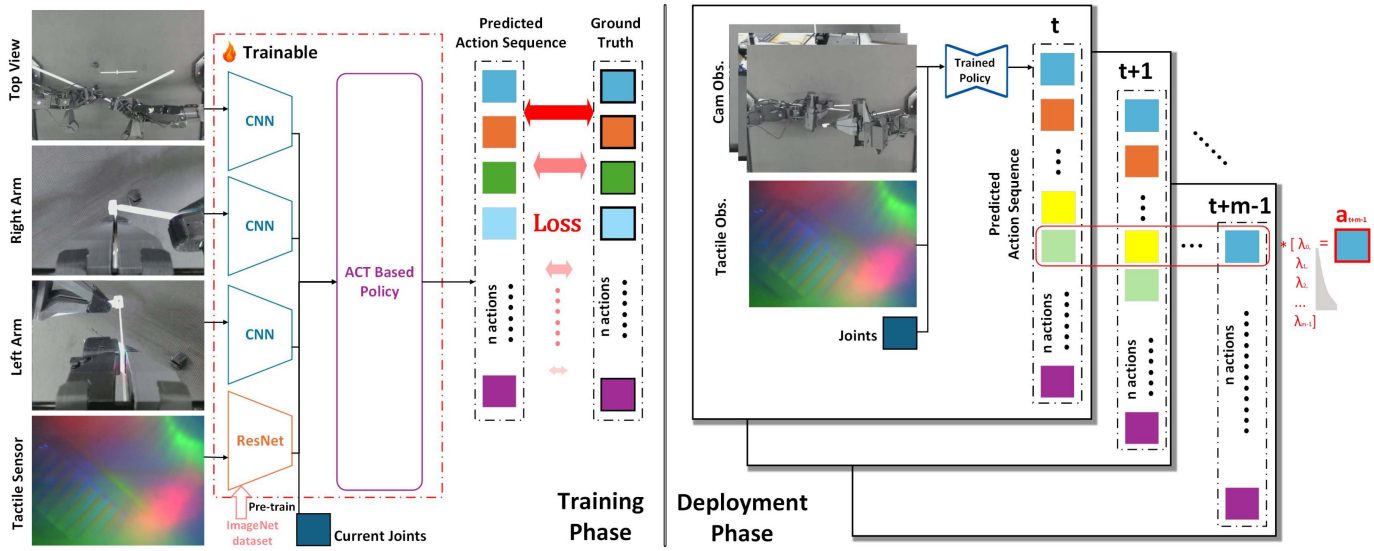

Fig. 2. Approach Overview. During the training phase, the input includes camera images from the top view, right-arm, and left-arm cameras, tactile information from a GelSight sensor, and the current robot joint states. The tactile information is encoded by ResNet pre-trained on ImageNet [9] and connected with the encoded camera images and the robot's proprioceptive information (i.e., joint states). These connected data serve as inputs to the algorithm and then output an action chunk, a sequence of predicted action, for the next n timesteps, which is used to compute the loss against the ground truth n-step actions. For loss computation, we use a weighting scheme that emphasizes near-future actions in the predicted n-step sequence by applying exponentially decaying weights over time. The components enclosed by the red dashed box are trainable and receive gradient updates during training. During the deployment phase, we provide camera images, tactile observations, and two robot joint values as input to the policy. We query the policy iteratively at each timestep and output the next n predicted actions. For an action that needs to be executed at timestep m, we aggregate the predicted actions with the same timestep index from previous predictions using exponential coefficients, where the most recent prediction receives the highest coefficient, while earlier predictions receive lower coefficients. This aggregation method ensures smooth operation while maintaining real-time performance and accuracy in robotic manipulation.

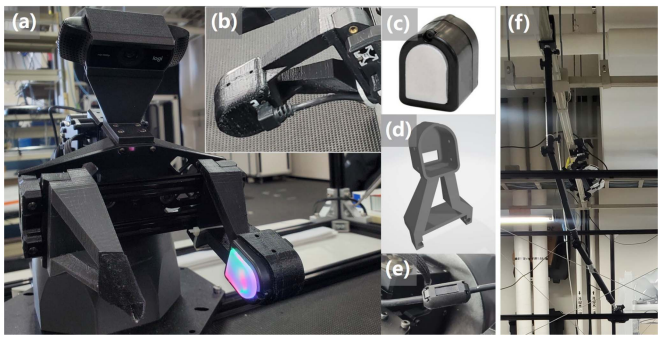

Fig. 3. TactileAloha Key Composition. (a) shows the view of the main upgraded part, while (b) shows its back view. We utilize a GelSight sensor (c), which provides the tactile texture of the object, while we fabricate a model (d) by 3D printing to mount it. We recommend using a right-angle micro USB cable, which reduces collision risk during manipulation. Moreover, we install noise absorbers (e) for the tactile sensor cable. Additionally, we redesign the top-view camera installation method. While Aloha mounts them on their metal frame, we install them on a fixed boom (f) to better stabilize the camera.

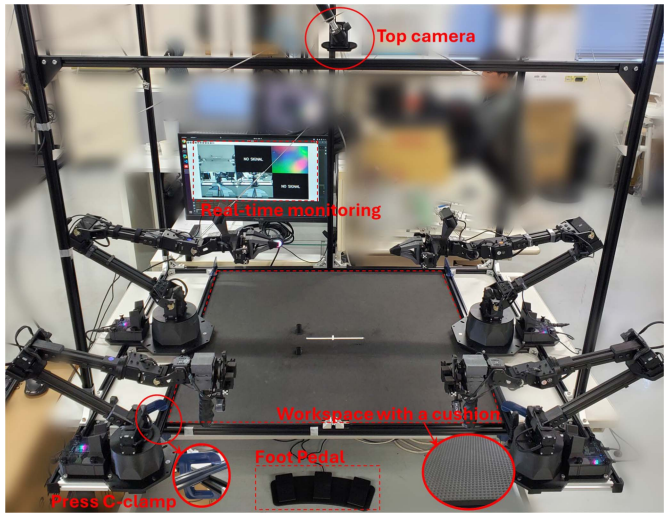

Fig. 4. Overview of Our TactileAloha Platform. Except for the updates to the tactile sensor and camera, we further use a real-time monitoring window that displays real-time camera and sensor observations during dataset collection to ensure dataset accuracy. The workspace is covered with an 820*780*6 mm cushion to protect the robot gripper during manipulation. Four C-clamps are fixed at the four corners of the platform to reduce the impact of vibrations caused by the robot's movement. A foot pedal facilitates dataset collection.

on the screen, particularly the visualized tactile information from GelSight. The foot pedal was used to confirm actions during teleoperation. For more details, please refer to the project website.

IV. Approach

A. Method Overview

Fig. 2 shows an overview of our approach. The learning policy is based on ACT [2] with our proposed TactileAloha robotic platform. In this work, we encode the tactile information during manipulation using a pre-trained ResNet and input it into the ACT-based policy. We further use a weighted exponentially decaying loss function (Section IV-B) to train the policy, encouraging it to focus more on near-future predicted actions. Finally, we employ an improved temporal ensembling method, Temporal Proximity Ensembling (Section IV-C), which

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

aggregates predicted actions across recent chunks to compute the current action, thereby improving manipulation precision while maintaining responsiveness.

B. Exponentially Decaying Loss Function

Chunking Action-based Policy [20], where the agent receives an observation, generates the next n sequential actions, named an action chunk, and formulates it as: $\pi_\theta(\hat{a}_{t:t+n} \mid s_t)$. Therefore, the reconstruction loss [2] for actions is computed based on the discrepancy between the predicted action chunk $\hat{a}_{t:t+n}$ and the ground-truth action chunk $a_{t:t+n}$, both consisting of n sequential actions. The loss in the original ACT is computed by averaging the discrepancy of the n actions. While in the iterative querying and aggregation manipulation setting (Section IV-C), nearer actions in the action chunk play a crucial role, while the farther ones in the chunk have less influence. Therefore, we use a weighted loss function that applies exponentially decaying weights to the discrepancies, actions nearer in the chunk are assigned higher weights, which decay exponentially. The weight u_i is defined as follows:

$$u_i = \frac{(1 - e^{-k})e^{-k \cdot i}}{1 - e^{-kn}} \tag{1}$$

where the denominator ensures that the weights are properly normalized, i.e., they sum to 1 over the given range. Here, u_0 is the weight for the nearest predicted action discrepancy in the list $\hat{a}_{t:t+i}$. While k controls the rate of exponential decay, a larger k results in a steeper decay, assigning significantly higher importance to the discrepancy of the nearest predicted action $\hat{a}_t$, while rapidly reducing the influence of the actions further away in the prediction sequence, i.e., $\hat{a}_{t+n}$.

Lastly, we formulate the loss [2] for action as follows:

$$L = \sum_{i=0}^{n-1} u_i \cdot \ell(\hat{a}_{t+i}, a_{t+i}) \tag{2}$$

where we use L1 to compute the loss ℓ.

C. Temporal Proximity Ensembling

Compounding error is a common challenge in long-horizon manipulation tasks. An action-chunking-based policy [20] can mitigate this issue by predicting n consecutive actions in a single forward pass and executing them as a single chunk. Furthermore, Zhao et al. [2] introduced an interactive querying mechanism and ensembled the predicted actions with the same timestep index across the predicted action chunk to improve smoothness and avoid discrete switching between action chunks. However, their method emphasizes the earliest predicted action, which is suitable for tasks involving a single manipulation strategy, but is limited when applied to tasks requiring multiple distinct strategies, such as zip tie insertion or Velcro fastening. The manipulation strategy may switch, and an aggregation scheme that overly favors earlier predictions prevents the policy from responding effectively to such changes. Their method fails to promptly suppress the influence of outdated predictions, causing lingering effects that may lead to task failure. To address this issue, we utilize an improved temporal ensembling strategy,

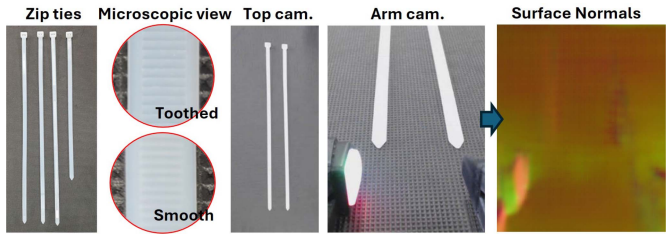

Fig. 5. Zip ties used for Training and Testing. We use four zip ties for the experiment, each with a width of 1 cm. The lengths include 40 cm and 30 cm, with the 40 cm ones inserted into the head of the 30 cm one. The zip tie has two sides: a toothed side and a smooth side. See the microscopic view image taken by a high-precision camera. However, even with a high-precision camera, the toothed and smooth sides are still difficult to distinguish due to the transparency of the plastic. RGB images from the top and arm cameras, as well as the derived surface normal representations, also fail to distinguish between the two surfaces.

called Temporal Proximity Ensembling module, during policy deployment, as shown on the right of Fig. 2. Here, "proximity" refers to temporal closeness, with more recent predictions receiving higher weights during aggregation, thereby improving responsiveness.

The Temporal Proximity Ensembling module queries the trained policy at every timestep, which causes different predicted action sequences to overlap with each other. Therefore, at a given timestep, there will be more than one predicted action. The maximum number of predicted actions with the same timestep index is n, which is equal to the length of the predicted action sequence. We aggregate the predictions that have the same timestep index within a pre-defined span. In the right part of Fig. 2, we define this span as m, where $m \leq n$, meaning that we only trace back m timesteps prediction, as older predictions may carry higher potential bias or inaccuracy. However, if m is too small, it will lead to unsmooth manipulation. For the aggregation process, to further focus the policy on recent, we incorporate an exponentially scaled coefficient scheme $\lambda_i = e^{(d \cdot i)}$, which assigns greater importance to recent predictions, i.e., the blue action box within the red rectangular frame of Fig. 2, while reducing the coefficients of earlier predictions exponentially, i.e., the green action box within the red rectangular frame. A higher d means more emphasis on recent predictions, which could make the manipulation more precise but may also lead to torpid and short-sighted robot motion. By tuning the traceback value m and the coefficient factor d, we can have better performance.

V. Experiments and Results

A. Tasks

We design two bimanual tasks for tactile sensing manipulation, involving two materials: zip ties (see Fig. 5) and Velcro (see Fig. 6). Both have two distinct textures on their two faces, which are difficult to observe with cameras but can be detected by the tactile sensor (see the tactile information in Fig. 1). To enhance visual insights of surface textures, we additionally provide surface normal representations [21] derived from the arm-mounted RGB images.

As shown in Figs. 7 and 8, the zip tie insertion and Velcro fastening tasks each consist of three subtasks: **Grasp**, **Align**, and

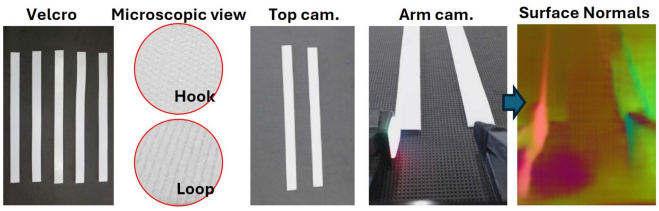

Fig. 6. Velcro Used for Training and Testing. We use five Velcro for the experiment, each measuring 30 cm*2 cm. The Velcro has two sides: a hook side and a loop side, as shown in the microscopic view image. However, the camera vision system cannot recognize the difference between the two sides.

Insert/Fasten. We illustrate the initialization process, outline each subtask, and describe the corresponding execution steps. Adapting to variations in given object orientation, the manipulation is done with different strategies by texture perception in a systematic manner. Among them, only the **Align** subtask relies on both tactile and visual feedback, while the others are standard vision-based manipulation tasks.

B. Details of Dataset, Training, and Deployment

For both tasks, we use the TactileAloha system to collect demonstrations. We collect 80 demonstrations for zip tie insertion, with zip ties randomly selected in each episode. Each demonstration consists of 900 timesteps (18 seconds). For Velcro fastening, we collect 50 demonstrations, each with 1100 timesteps (22 seconds), and randomly sample the Velcro strip in each episode.

In this work, we attempt to use the co-training method from MobileAloha [17], applying co-training with datasets from Aloha [2] and MobileAloha [17]. However, we find that it negatively impacts the performance, especially during grasping, where the gripper closes prematurely, leading to failures. For the video, please refer to the project website.

We train our policy on a server equipped with RTX A6000 GPUs. To encode tactile information, we adopt ResNet18 for its strong feature extraction capability. Following prior works [22], [23], the tactile encoder is initialized with ImageNet pre-trained weights, and is further fine-tuned jointly with the rest of the network during training. The whole policy is trained for 100 k iterations (approximately 50 hours) using the Adam optimizer with a learning rate of $1.0e-5$, a chunk size of 100, and a batch size of 32. The reconstruction loss [2] in our ACT-based policy is computed using the Mean Absolute Error with a decay coefficient of $k = 1.0e-2$, as defined in (1).

During deployment, we use a computer connected to the TactileAloha system, which consists of an RTX 3090 GPU, an Intel W5-2455X CPU running at 3.20 GHz, and a total of 64 GB of RAM for policy evaluation. Each time, we randomly select a zip tie and a piece of Velcro. For the Temporal Proximity Ensembling module, we set the hyperparameter d to $1.0e-2$ and m to 50 for zip tie insertion, while for Velcro fastening, we set d to $2.0e-3$ and m to 80. For the timesteps used in the deployment evaluation, it may increase the length to ensure the whole manipulation process is completed. In our case, we expand it by 1.2X based on the task dataset collection timesteps' length.

C. Comparsion

To demonstrate the performance of our method, we compare it with approaches that only input vision information: ACT [2] and Diffusion Policy [1]. ACT [2] is the original method on which our approach is based. It utilizes chunking actions and Transformers, achieving high manipulation precision and stability. The Diffusion Policy method [1] is a diffusion-based generative model for robotic manipulation, which is well-suited for high-dimensional action spaces. To explore the impact of enhanced visual representations, we further evaluate an ACT variant augmented with surface normal observations [21], denoted as ACT with Normals. This model receives mid-level geometric features derived from the arm-mounted RGB camera, aiming to improve the camera's perception of subtle surface textures. In the comparison, both algorithms rely solely on camera visual input, without tactile information, to evaluate whether vision alone in a robotic platform can distinguish differences and guide robotic manipulation. We further evaluate ACT with tactile input to compare against our full method and assess the impact of the additional improvements.

For the comparison metric, we report the success rate of each subtask. To ensure consistency, each task with each method is evaluated over 20 trials, with a random initial position. During manipulation, if the robot arm cannot complete the current subtask or is at risk of collision, we terminate the current action and report the success rates of the completed subtasks.

Table I shows the comparison result. By comparing ACT with Diffusion Policy, we can observe the importance of action chunking for manipulation, which improves the coherence of the generated trajectory [17]. We further evaluate ACT with Normals. However, this variant does not yield improvement in perceiving texture orientation compared to the standard ACT baseline, suggesting that surface normal representations alone are insufficient for capturing fine-grained surface information in this setting. In contrast, ACT with tactile input outperforms the standard ACT in the alignment subtask, which requires accurate recognition of grasped texture to determine the alignment pose. This demonstrates the crucial role of tactile sensing, which cannot be substituted by vision-based modalities. Moreover, our method achieves higher success rates in the final insertion subtask and the fastening subtask compared to ACT with tactile input. Averaged across all six sub-tasks from both tasks, it achieves an overall relative improvement of approximately 11.0%, demonstrating the effectiveness and precision of our method. Fig. 9 showcases the zip tie insertion and Velcro fastening evaluation processes and provides a qualitative comparison. For more video details, please visit the project website.

In addition to reporting subtask success rates, we also analyzed failure cases. Grasping failures included weak or unstable grasps, object slippage during execution, or the gripper targeting an incorrect position. These were relatively evenly distributed, with no single type being particularly dominant. Alignment failures primarily occurred in settings without tactile sensing, highlighting the importance of contact feedback. We also observed alignment failures in multi-strategy tasks, where the current trajectory followed a new strategy but was influenced

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

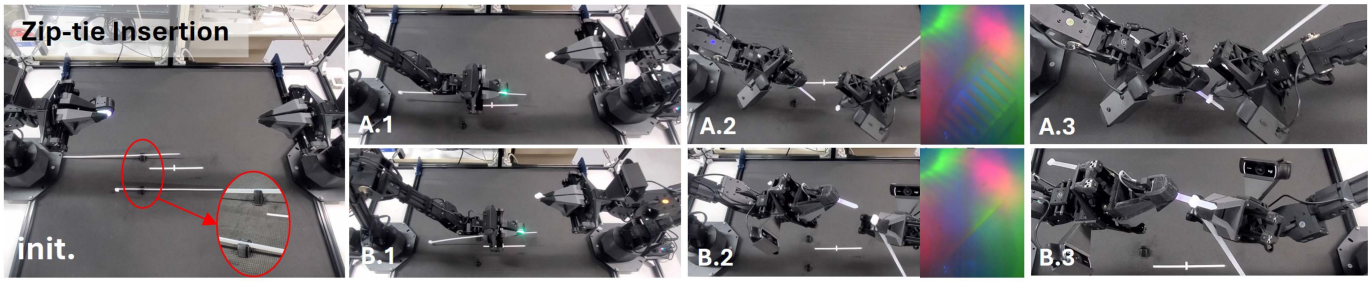

Fig. 7. Zip tie Insertion: Insert one zip tie into the head of another. Two zip ties are initially placed on the workspace. To facilitate grasping by the robot gripper, two supports are fabricated by 3D printing to keep the zip ties stand. The orientation of the near zip tie is fixed, while the facing direction of the far zip tie is random. Moreover, the head of the far zip tie cannot be observed because it is occluded by the robot base. During manipulation, the robot arms first **Grasp** the twoes (A.1, B.1). The tactile sensor then perceives the orientation, and the arms **Align** by adjusting posture to ensure that the toothed side aligns with the pawl of the receiving tie (A.2, B.2). Finally, the lefte is **Insert**ed into the right zip head (A.3, B.3) with precise motion. Depending on the zip tie's initial orientation, different manipulation sequence was automatically generated as shown in A and B based on tactile sensing.

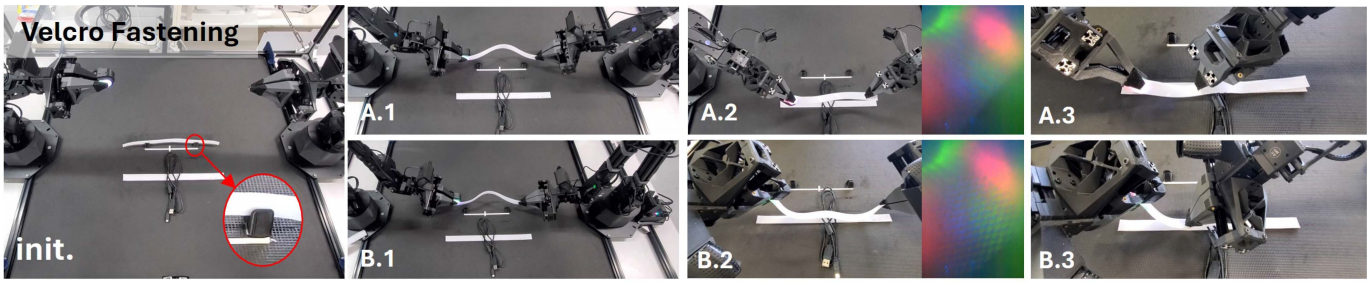

Fig. 8. Velcro Fastening: Engage one piece of Velcro with another to fasten the USB cable. One USB cable is placed on a piece of Velcro, whose position is fixed, with the hook side facing up. Another piece of Velcro is placed next to it, supported by two 3D-printed supports to make it stand, and its orientation is random. During manipulation, the robot arms first **Grasp** both Velcro edges (A.1, B.1). The tactile sensor perceives orientation, and the arms **Align** by adjusting posture and angle to ensure that the hook side matches the loop side (A.2, B.2). Finally, the arms **Fasten** the Velcro, with the right arm pressing to ensure engagement (A.3, B.3). Adapting to the given Velcro side, the tape handling is done with different ways for covering the tape so that the hook and loop sides of the Velcro tape contact each other.

TABLE I
RESULTS OF OUR METHOD AND BASELINES

Method	Zip Tie Insertion [%]			Velcro Fastening [%]		
	Grasp	Align	Insert	Grasp	Align	Fasten
Diffusion Policy	60	20	0	70	30	15
ACT	80	30	5	90	45	30
ACT with Normals	75	35	5	85	40	30
ACT with Tactile	80	75	25	95	95	85
Ours	**90**	**90**	**35**	**100**	**100**	**90**

Note: Bolded values highlight the best results across the compared methods.

by residual predictions from a previous one. This issue was particularly evident under the temporal ensembling scheme used in the original ACT [2]. As shown in Fig. 10, ensembling incompatible actions led to execution conflicts and, in some cases, collisions. These failures arose from the method's inability to promptly discard outdated predictions. Others, insertion failures were mostly due to misalignment with the small zip tie head opening. For Velcro fastening, failures typically arose during the pressing phase, where the Velcro failed to properly engage, sometimes due to alignment precision or interference with the tabletop.

To evaluate sensitivity to the hyperparameters k (Exponentially Decaying Loss), and d, m (Temporal Proximity Ensembling), we varied each individually while keeping the others fixed. We report the average success rate across the three subtasks of the zip tie insertion task, see Fig. 11.

D. Ablation

To evaluate the introduced aspects, we design three ablations to examine the respective contributions:

- Ours with ACT's TE (our method using the temporal ensemble from ACT): In this setting, we replace our Temporal Proximity Ensembling module with the temporal ensemble method proposed in ACT.
- W/O EDL (Without Exponentially Decaying Loss Function Module): In this setting, we don't use the introduced Exponentially Decaying Loss Function. The loss for the chunking action will be computed as the mean of n predicted values: $L = \frac{1}{n}\sum_{i=0}^{n-1} \ell(\hat{a}_{t+i}, a_{t+i})$.
- W/O TPE (Without Temporal Proximity Ensembling Module): We further exclude the utilization of the Temporal Proximity Ensembling Module, resulting in no aggregation with the previous prediction sequence actions. Consequently, the observe&action process is executed a total number of times equal to the whole horizon H divided by the action chunking size n, i.e., H/n.

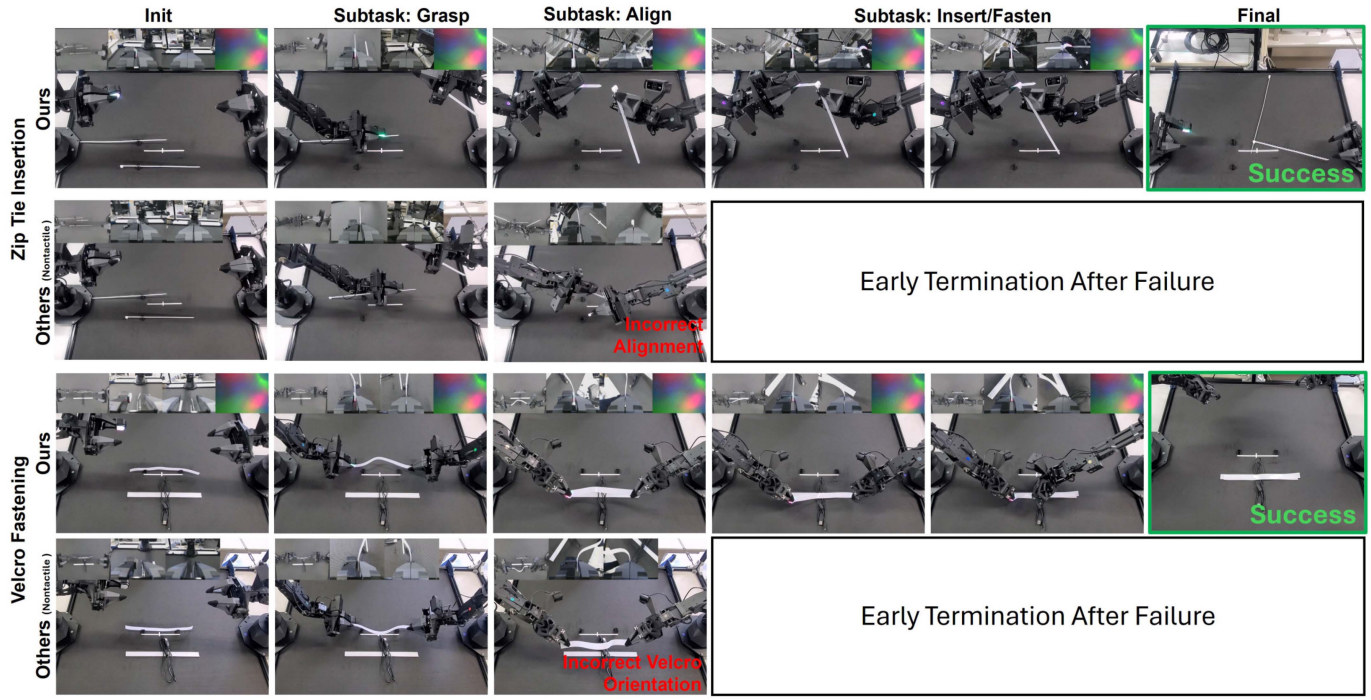

Fig. 9. Qualitative comparison of experimental evaluation on zip tie insertion and Velcro fastening.

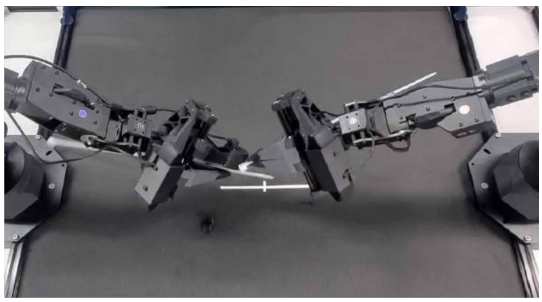

Fig. 10. Scene where the robot collision occurs in the method using ACT [2] temporal ensembling.

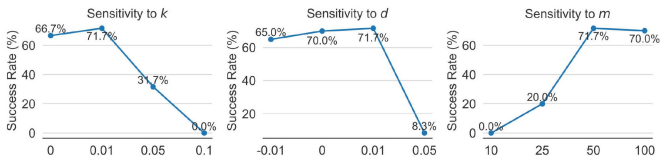

Fig. 11. Sensitivity analysis of the hyperparameters of our transformer-based policy with action chunking.

- W/O Tac. (Without Tactile sensing): We further train our policy without tactile information, so it only manipulates based on vision input from three cameras.
- EDL + TPE: Train and deploy the policy with EDL and TPE, without tactile sensing.

We conduct 20 trials for each ablation and evaluate only the zip tie insertion task. The results are presented in Table II. The ablations consistently perform worse than the full method. We observe that ACT's temporal ensembling module is not compatible with our Exponentially Decaying Loss function, resulting in a performance decrease. This may be because, during deployment, ACT's temporal ensembling scheme assigns greater weight to the earliest predicted actions, which are also the furthest in time within the predicted action chunk, while our exponentially decaying loss function reduces the training loss associated with these actions. For the other three ablations, we find that the Exponentially Decaying Loss module generally improves performance across all three subtasks of zip tie insertion. A comparison between the performance of W/O EDL and W/O TPE shows that TPE contributes more to the precise insert subtask, as the policy can receive real-time feedback and adjust its control accordingly. The comparison between W/O TPE and W/O Tac. demonstrates the influence of tactile information on changes in the alignment success rate. The results of EDL + TPE and the full method indicate that incorporating tactile information does not negatively affect performance on subtasks, i.e., grasp and insert, that do not rely on tactile sensing.

TABLE II
THE RESULTS OF ABLATIONS

Method	EDL	TPE	Tac.	Zip Tie Insertion [%]		
				Grasp	Align	Insert
Ours	✓	✓	✓	**90**	**90**	**35**
Ours with ACT's TE	✓	ACT's TE	✓	80	75	15
W/O EDL		✓	✓	85	85	30
W/O TPE			✓	70	60	10
W/O Tac.				65	25	5
EDL + TPE	✓	✓		**90**	40	15

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

VI. Conclusion

To facilitate tactile-aware robotic manipulation, we introduce TactileAloha, a bimanual manipulation system that extends the Aloha platform by integrating a GelSight tactile sensor into the control loop. This enables real-time tactile sensing during contact-rich manipulation. We explore an existing transformer-based policy framework (Action Chunking with Transformers) to incorporate tactile observations, which are encoded using a pre-trained ResNet and fused with visual and proprioceptive inputs to predict future action sequences. To improve the quality of these predictions, we use a weighted loss and an improved temporal aggregation strategy that emphasize near-future actions and enhance control accuracy. Experimentally, we design two tasks: zip tie insertion and Velcro fastening, both of which are highly sensitive to tactile information. In both tasks, there is an object orientation condition to manage the task by bimanual manipulation, so that the zip tie's toothed side aligns with the pawl of the receiving tie, and the hook side matches the loop side of the Velcro, respectively. Different texture orientations of handling objects could systematically result in different manipulation strategies based on tactile sensing with the proposed method. Our results demonstrate significant performance improvements, highlighting the effectiveness of our method toward tactile-based manipulation.

References

[1] C. Chi et al., "Diffusion policy: Visuomotor policy learning via action diffusion," in *Proc. Robot.: Sci. Syst.*, Daegu, Republic of Korea, Jul. 2023.

[2] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn, "Learning fine-grained bimanual manipulation with low-cost hardware," in *Proc. Robot.: Sci. Syst. XIX*, K. E. Bekris, K. Hauser, S. L. Herbert, and J. Yu, eds. Daegu, South Korea: Daegu Exhibition and Convention Center, 2023.

[3] C. Zhou et al., "Dual-arm robotic fabric manipulation with quasi-static and dynamic primitives for rapid garment flattening," *IEEE/ASME Trans. Mechatron.*, early access, Apr. 18, 2025, doi: 10.1109/TMECH.2025.3556283.

[4] N. Gu, R. He, and L. Yu, "Learning to unfold garment effectively into oriented direction," *IEEE Robot. Autom. Lett.*, vol. 9, no. 2, pp. 1051–1058, Feb. 2024.

[5] N. Gu, Z. Zhang, R. He, and L. Yu, "Shakingbot: Dynamic manipulation for bagging," *Robotica*, vol. 42, no. 3, pp. 775–791, 2024.

[6] N. Sunil, S. Wang, Y. She, E. Adelson, and A. R. Garcia, "Visuotactile affordances for cloth manipulation with local control," in *Proc. Conf. Robot Learn.*, 2023, pp. 1596–1606.

[7] S. Tirumala, T. Weng, D. Seita, O. Kroemer, Z. Temel, and D. Held, "Learning to singulate layers of cloth using tactile feedback," in *Proc. 2022 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2022, pp. 7773–7780.

[8] R. Proesmans, A. Verleysen, and F. Wyffels, "UnfoldIR: Tactile robotic unfolding of cloth," *IEEE Robot. Automat. Lett.*, vol. 8, no. 8, pp. 4426–4432, Aug. 2023.

[9] J. Deng, W. Dong, R. Socher, L. -J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. 2009 IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.

[10] W. Yuan, S. Dong, and E. H. Adelson, "Gelsight: High-resolution robot tactile sensors for estimating geometry and force," *Sensors*, vol. 17, no. 12, 2017, Art. no. 2762.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[12] X. Mao et al., "Learning fine pinch-grasp skills using tactile sensing from real demonstration data," 2023, *arXiv:2307.04619*.

[13] M. Lambeta et al., "DIGIT: A novel design for a low-cost compact high-resolution tactile sensor with application to in-hand manipulation," *IEEE Robot. Autom. Lett.*, vol. 5, no. 3, pp. 3838–3845, Jul. 2020.

[14] R. Bhirangi, T. Hellebrekers, C. Majidi, and A. Gupta, "Reskin: Versatile, replaceable, lasting tactile skins," in *Proc. 5th Annu. Conf. Robot Learn.*, 2021, vol. 164, pp. 587–597.

[15] A. Hussein, M. M. Gaber, E. Elyan, and C. Jayne, "Imitation learning: A survey of learning methods," *ACM Comput. Surveys*, vol. 50, no. 2, pp. 1–35, 2017.

[16] D. A. Pomerleau, "Alvinn: An autonomous land vehicle in a neural network," in *Proc. Adv. Neural Inf. Process. Syst.*, 1988, pp. 305–313.

[17] Z. Fu, T. Z. Zhao, and C. Finn, "Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation," in *Proc. Conf. Robot Learn.*, P. Agrawal, O. Kroemer, and W. Burgard, Eds., Nov. 2024, vol. 270, pp. 4066–4083.

[18] M. Burke, Y. Hristov, and S. Ramamoorthy, "Hybrid system identification using switching density networks," in *Proc. Conf. Robot Learn.*, 2020, pp. 172–181.

[19] R. Goebel, R. G. Sanfelice, and A. R. Teel, "Hybrid dynamical systems," *IEEE Control Syst. Mag.*, vol. 29, no. 2, pp. 28–93, Apr. 2009.

[20] L. Lai, A. Z. X. Huang, and S. J. Gershman, "Action chunking as policy compression," *Cognition*, vol. 264, 2025, Art. no. 106201.

[21] B. Chen et al., "Robust policies via mid-level visual representations: An experimental study in manipulation and navigation," in *Proc. 4th Conf. Robot Learn.*, 2020, pp. 2328–2346.

[22] V. Dave, F. Lygerakis, and E. Rueckert, "Multimodal visual-tactile representation learning through self-supervised contrastive pre-training," in *Proc. IEEE Int. Conf. Robot. Automat.*, Yokohama, Japan 2024, pp. 8013–8020.

[23] H. -G. Chi, J. Barreiros, J. Mercat, K. Ramani, and T. Kollar, "Multi-modal representation learning with tactile data," in *Proc. 2024 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2024, pp. 9660–9667.