Contact SLAM: An Active Tactile Exploration Policy Based on Physical Reasoning Utilized in Robotic Fine Blind Manipulation Tasks

Gaozhao Wang, Xing Liu*, Zhenduo Ye, Zhengxiong Liu and Panfeng Huang

***Abstract*—Contact-rich manipulation is difficult for robots to execute and requires accurate perception of the environment. In some scenarios, vision is occluded. The robot can then no longer obtain real-time scene state information through visual feedback. This is called "blind manipulation". In this manuscript, a novel physically-driven contact cognition method, called "Contact SLAM", is proposed. It estimates the state of the environment and achieves manipulation using only tactile sensing and prior knowledge of the scene. To maximize exploration efficiency, this manuscript also designs an active exploration policy. The policy gradually reduces uncertainties in the manipulation scene. The experimental results demonstrated the effectiveness and accuracy of the proposed method in several contact-rich tasks, including the difficult and delicate socket assembly task and block-pushing task.**

I. Introduction

In robotic manipulation tasks, vision plays an irreplaceable role. Through visual servoing, a robot can identify the target of manipulation and control its manipulator to complete the task [1]. In this process, the tactile or force sensing is mainly used to interact with the environment after the vision has already provided the target position, leading the manipulator to achieve its operational goal with the dual guidance of vision and force sensing [2].

However, in certain tasks, when vision fails or is occluded, the robot can no longer obtain real-time scene state information via visual feedback. We refer to such manipulation tasks as "blind manipulation" tasks, in which the robot must rely solely on tactile and force information. Consequently, researchers have begun investigating how to transform contact data into state information about the scene [3], [4]. However, unlike directly contacting and measuring the environment, in general-purpose manipulation scenarios, the end-effector is often equipped with a two-finger gripper or a dexterous hand to handle different objects, and then uses these objects to interact with the environment, which means the force–tactile information captured cannot be directly used to determine the environment's state. Although some studies have investigated the perception process during manipulation with grippers holding objects [5], [6], they typically stop at the perception stage without conducting an in-depth study on task completion.

Gaozhao Wang, Xing Liu(Corresponding author), Zhenduo Ye, Zhengxiong Liu and Panfeng Huang are with the Research Center for Intelligent Robotics, the National Key Laboratory of Aerospace Flight Dynamics, and Shaanxi Province Innovation Team of Intelligent Robotic Technology, School of Astronautics, Northwestern Polytechnical University, Xi'an 710072, China. (e-mail: gaozhao_wang@mail.nwpu.edu.cn.)

During human manipulation, in addition to visual awareness of the process, there is also contact awareness. By judging the force sensed at the fingertips, people can estimate whether the manipulation goal has been reached and, if not, how far they are from the target state. In robotic manipulation, visual servoing generally determines the target object in the scene through active perception and cognition; however, in blind manipulation tasks, this mechanism is absent. Some researches have tried to transform the force or tactile signals into some forms of contact state [7], [8], but these methods usually have strong data dependency. Inspired by human manipulation, we introduce physically-driven contact cognition into robotic blind manipulation. By reasoning about the contact state between the grasped object and the environment from contact signals, the robot can continuously estimate and explore the environment until ultimately completing the manipulation task, as illustrated in Fig. 1.

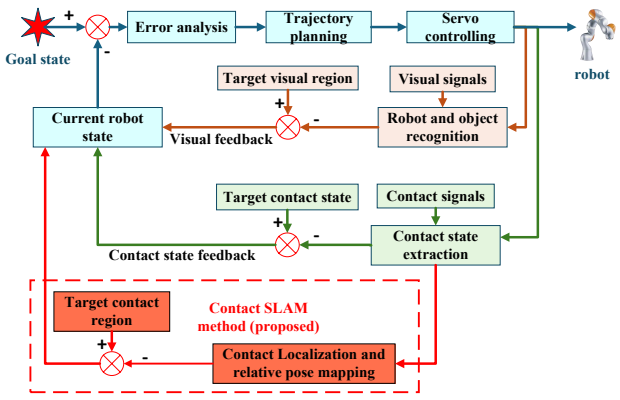

Fig. 1. The perception and cognition process in robot manipulation tasks. In general, the visual signals are used to understand the scene. When the vision is occluded, other kinds of information, such as tactile signals, are used to obtain contact states. However, the contact state is hard to be labelled and classified. We proposed the contact SLAM method, which doesn't pursue obtaining an exact contact state. Instead, it acquires the contact area with the environment through the cognitive process during the exploration of the environment.

The contributions of this paper are as follows:

1) **A tactile perception method for estimating the force and its position acting on the grasped object is proposed.** Given the precise geometric shape of the grasped object and the motion trends of tactile sensors mounted on the two-finger gripper, the proposed method can determine the forces exerted on the grasped object, as well as the contact position.

2) **A contact-based SLAM method for environment perception and localization is proposed**. This method further develops and extends the scope of indirect perception and tactile SLAM, enabling the perception and understanding of the scene state based on prior scene knowledge and real-time tactile feedback.
3) **A planning method for fine blind manipulation based on active tactile exploration is proposed**. This approach balances environmental exploration with task completion efficiency, plans optimal action strategies according to the current understanding of the environment, and reduces the uncertainties of the manipulation scene gradually.

II. Related Work

A. Tactile Perception During Manipulation Process

Compared with visual sensing, tactile sensing provides more comprehensive information about the contact state with the environment during manipulation, including the geometry of the contacted object [9], the contact force/torque and its distribution [10], and contact event triggering [11]. With these perceptual features, they can be further applied to robotic manipulation tasks [12], [13]. In terms of interaction with the environment, end-to-end policy generation is typically implemented through a sim-to-real approach [14]. Alternatively, tactile sensing can be used as an independent observer [15], whose results are then fed into the control loop.

B. Localization and Mapping Method based on Tactile Sensing

Sudharshan Suresh et al. [16] proposed using force–tactile sensing to simultaneously track an object's position and model its contour, a method they termed Tactile SLAM. Jialiang Zhao et al. [17] introduced the concept of FingerSLAM. Paloma Sodhi et al. [18] applied the SLAM modeling concept to object manipulation control, achieving tactile-based pushing operations. Daolin Ma et al. [5] proposed a method that uses visuo-tactile sensor to estimate the relative pose of an object grasped by a two-finger gripper, as well as the location of external contact points and contact lines. Building on this idea, Sangwoon Kim et al. [6] proposed a method for estimating the state of an object during manipulation to maintain its stability, along with an approach for perceiving external contact lines and enabling general peg-in-hole assembly [19].

C. Cognition and Reasoning During Manipulation Process

Object cognition primarily focuses on visual sensing. Chaofan Zhang et al. proposed VTLA [20], which integrates tactile modality into a large-scale manipulation model. Features representing contact constraints are generally obtained by two approaches: one is through force–tactile sensors, and the other is through a world model [21]. State cognition tends to focus on low-level state feedback and control, such as determining whether specific signals relevant to task completion are present during contact [22] or designing contact feedback control loops [23], [24] to replicate particular contact signals.

III. Task Formulation

In blind manipulation tasks, several kinds of objects need to be noticed. The grasped objects, which are grasped by the gripper, and the tactile sensors could only perceive the tactile signals between the grasped objects and the gripper. The manipulated objects, which are used to interact with the grasped objects, including unmovable obstacles and movable objects. The last is the target region, which the grasped objects or manipulated objects need to reach. The detailed scenes are shown in section V-A.

In order to represent the pose and of these things, the coordinates used in this manuscript are illustrated in Fig. 2(a), and the iteration between the grasped object and obstacles is shown in Fig. 2(b).

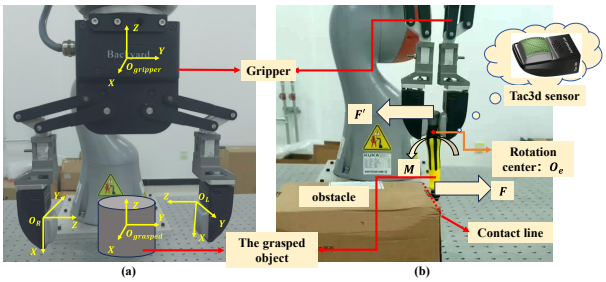

Fig. 2. (a) represents the coordinates used in this manuscript. The left and right sensor coordinates are opposite to the left and right positions in space because of sensor setting. (b) represents the corresponding forces and torques applied to the object when contact happens.

To achieve the task objectives, we make the following assumptions:

1) There exists a relative motion tendency between the grasped object and the gripper, but no relative translational displacement.
2) The contact between objects is quasi-static.
3) The precise geometric dimensions and shapes of the grasped object and the manipulated object are known in advance.
4) All objects in the environment are static and do not move autonomously; the only moving bodies are the gripper and the objects it holds.

IV. Methods

The complete contact SLAM method includes tactile perception, localization, mapping and active exploration process, as illustrated in Fig. 3.

A. Contact Force and Pose Perception of the Grasped Object

The Tac3D sensor used in this study provides the three-dimensional force distribution, the resultant three-dimensional force, and the resultant three-dimensional torque in the sensor coordinate frame. These enable researchers to determine not only the magnitude but also the trend of forces acting on the grasped object.

When a two-finger gripper grasps an object, all forces acting on the grasped object are reflected in the tactile sensors mounted on the gripper fingers. The two-finger gripper and the

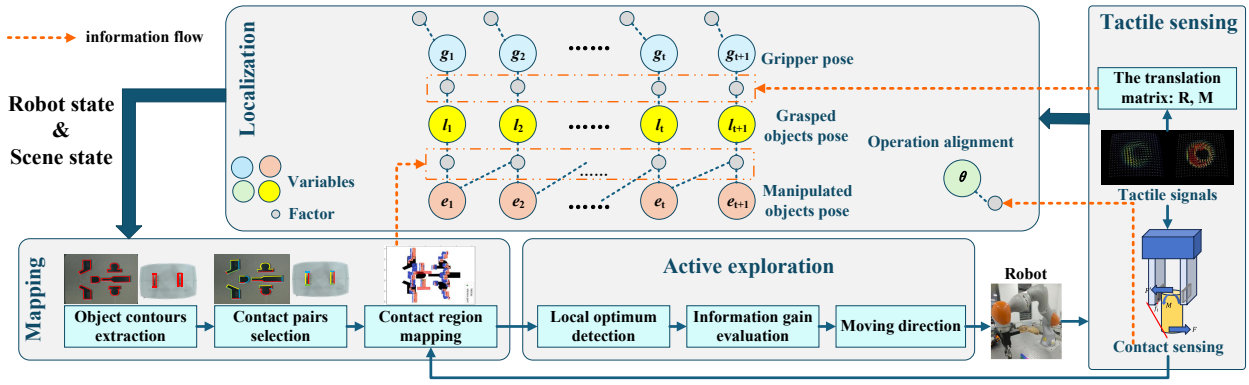

Fig. 3. The complete process of the proposed method. The method regards the manipulation process as an SLAM problem. In the localization phase, it focuses on determining the positions of the robot end-effector, the grasped object, the target object, and other objects of interest in the environment. In the mapping phase, it focuses on representing the current relative contact information together with the tactile information between the grasped object and the environment. After one SLAM process, generate the action strategy based on the information gain evaluation in different movement directions, then the robot executes the strategy, the tactile sensing converts the tactile signals into physical information and sends them to the SLAM process.

grasped object could be modeled as a beam fixed at one end. When an external force is applied to one end, the reaction at the gripper includes not only a force in the opposite direction but also a torque relative to the grasping point. Both the forces and torques are applied to the tactile sensors mounted on the gripper.

As illustrated in Fig. 2(b), when a force in a certain direction acts on the grasped object, it could be decomposed into three components along the X, Y, and Z axes of the gripper coordinate system. To maintain the stability of the grasped object, the two gripper fingers produce corresponding torques on the object. The relationship between the forces applied to the object and those measured by the gripper tactile sensors can be expressed as follows:

$$\begin{cases} F_x = F_y^R - F_y^L \\ F_y = F_z^L - F_z^R \\ F_z = F_x^L - F_x^R \end{cases} \tag{1}$$

where F_x, F_y, and F_z are force components in the frame of the gripper. F_i^L and F_i^R are corresponding force components $(i = x, y, z)$ in the frame of the left sensor and right sensor.

When calculating the torque, the reference point must be chosen first. The measured resultant torques are calculated in the left and right sensor coordinate frames; however, what we need is the resultant torque calculated in the gripper coordinate frame about the equivalent rotation point: O_e. By analyzing the torque caused by a single force on a single point, the relationship between force (f_i^x, f_i^y, f_i^z) and torque (m_i^x, m_i^y, m_i^z) is as follows:

$$\begin{bmatrix} m_i^x \\ m_i^y \\ m_i^z \end{bmatrix} = \begin{bmatrix} 0 & -f_i^z & f_i^y \\ f_i^z & 0 & -f_i^x \\ f_i^y & -f_i^x & 0 \end{bmatrix} \cdot \begin{bmatrix} O_x \\ O_y \\ O_z \end{bmatrix} \tag{2}$$

where (O_x, O_y, O_z) is the pose of the force application point. Thus, when the reference point changes from O_1 to O_2 and doesn't consider the coordinate system rotation transformation, the torque changes to $((m_i^x)_{O_2}, (m_i^y)_{O_2}, (m_i^z)_{O_2})$:

$$\begin{bmatrix} (m_i^x)_{O_2} \\ (m_i^y)_{O_2} \\ (m_i^z)_{O_2} \end{bmatrix} = \begin{bmatrix} 0 & -f_i^z & f_i^y \\ f_i^z & 0 & -f_i^x \\ f_i^y & -f_i^x & 0 \end{bmatrix} \cdot \begin{bmatrix} O_1^x + \Delta X_{O_1}^{O_2} \\ O_1^y + \Delta Y_{O_1}^{O_2} \\ O_1^z + \Delta Z_{O_1}^{O_2} \end{bmatrix} \tag{3}$$

where $\Delta X_{O_1}^{O_2}, \Delta Y_{O_1}^{O_2}, \Delta Z_{O_1}^{O_2}$ are vectors from point O_2 to O_1.

In this way, the resultant torques applied to the grasped object about the equivalent rotation point O_e could be written as follows:

$$\begin{bmatrix} (M_x)_{O_e}^{L/R} \\ (M_y)_{O_e}^{L/R} \\ (M_z)_{O_e}^{L/R} \end{bmatrix} = \begin{bmatrix} (M_x)^{L/R} \\ (M_y)^{L/R} \\ (M_z)^{L/R} \end{bmatrix} + \begin{bmatrix} 0 & -F_z^{L/R} & F_y^{L/R} \\ F_z^{L/R} & 0 & -F_x^{L/R} \\ F_y^{L/R} & -F_x^{L/R} & 0 \end{bmatrix} \cdot \begin{bmatrix} \Delta X_{O_{L/R}}^{O_e} \\ \Delta Y_{O_{L/R}}^{O_e} \\ \Delta Z_{O_{L/R}}^{O_e} \end{bmatrix} \tag{4}$$

where F_i^L, M_i^L and F_i^R, M_i^R are corresponding force and torque components $(i = x, y, z)$ in the frame of the left sensor and right sensor. The parameters need to be measured are $\Delta X_{O_{L/R}}^{O_e}, \Delta Y_{O_{L/R}}^{O_e}, \Delta Z_{O_{L/R}}^{O_e}$.

Using the computed force and torque values in Eq. 1 and 4, we can apply the above equations to determine the force application region (C_x, C_y, C_z) of the grasped object:

$$\begin{bmatrix} (M_x)_{O_e}^L + (M_x)_{O_e}^R \\ (M_y)_{O_e}^L + (M_y)_{O_e}^R \\ (M_z)_{O_e}^L + (M_z)_{O_e}^R \end{bmatrix} = \begin{bmatrix} 0 & -F_z & F_y \\ F_z & 0 & -F_x \\ F_y & -F_x & 0 \end{bmatrix} \cdot \begin{bmatrix} C_x \\ C_y \\ C_z \end{bmatrix} \tag{5}$$

B. Contact SLAM Analysis

Contact SLAM aims to address the problems of relative pose localization between objects and environment reconstruction in contact-rich manipulation tasks, and then plans the subsequent end-effector motion trajectory for the robot.

According to the definition of SLAM, contact SLAM also includes both localization and mapping components:

- **Localization** focuses on determining the positions of the robot end-effector, the grasped object, the target object, and other objects of interest in the environment.
- **Mapping** involves determining the positions of the landmarks in the environment by representing and analyzing the current contact information between the grasped object and the environment. This approach does not seek to obtain an accurate contact configuration of the grasped object within the environment; instead, it only requires relative contact-region information, which is sufficient to satisfy the manipulation task requirements.

We employ factor graph optimization as our estimation framework, which is based on the principle of maximum a posteriori (MAP) estimation. A factor graph is a bipartite graph with two types of paprmeters: variables x and factors ϕ. Variable nodes are the latent states to be estimated, and factor nodes encode constraints on the variables, such as measurement likelihood functions, or physics, geometric models [18]. Under Gaussian noise model assumptions, MAP inference is equivalent to solving a nonlinear least-squares problem. That is, for Gaussian factors $\phi_i(x)$ corrupted by zero-mean, normally distributed noise, the inference equation is:

$$\hat{x} = \underset{x}{\operatorname{argmin}} \frac{1}{2} \sum_{i=1}^{n} \|F_i(x)\|^2_{\sum_i} \tag{6}$$

In contact SLAM, following factors are optimized:

- **Gripper Localization Factor** F_{gri}: The position of the gripper g_t can be derived from the position of the robot's end-effector:

$$\|F_{gri}(g_t)\|^2_{\sum_{gri}} = \left\| g_t - g_t^{pri} \right\|^2_{\sum_{gri}} \tag{7}$$

- **Grasped object Localization Factor** F_{obj}: The position of the object grasped by the gripper l_t can be obtained from the gripper position and the relative displacement of the object sensed by the tactile sensor mounted on the gripper. Based on the prior method [5], we define several coordinate frames: the gripper coordinate frame $(OXYZ)_g$, sensor coordinate frame $(OXYZ)_s$, grasped object coordinate frame $(OXYZ)_o$, and world coordinate frame $(OXYZ)_w$. Our estimation target is the position of a grasped object in the world coordinate frame, and the factor expression is:

$$\|F_{obj}(l_t, g_t)\|^2_{\sum_{obj}} = \left\| l_t - T_g^w T_s^g T_l^s I \right\|^2_{\sum_{obj}} \tag{8}$$

here, T_g^w represents the gripper pose which could be derived from the robot's end-effector pose, T_s^g is fixed during operation, and T_l^s depends on the relative displacement of the marker points sensed by the tactile sensor, which could be decomposed as: $T_t^s = T_{sd}^s T_l^{sd}$, where T_l^{sd} is unchanged. To obtain T_{sd}^s, the SVD-based optimization proces is acquired:

$$R^*, M^* = \min_{R,M} \sum_{i=1}^{N} \left\| p_i - (R \cdot p_i' + M) \right\|^2 \tag{9}$$

where the translation matrix T_{sd}^s is composed of the optimized rotation matrix R^* and translation matrix M^*.

- **Environment Localization Factor** F_{env}: Based on the contact pairs obtained through active exploration and the object position at time t, the position of obstacles e_t can be estimated. The estimated pose of environment in $(OXYZ)_w$ is:

$$\begin{aligned} &\|F_{env}(e_t, e_{t-1}, l_t)\|^2_{\sum_{env}} = \\ &\|B(l_t + B(P_t)) \cup B(l_t - B(P_t)) \cap e_{t-1} - e_t\|^2_{\sum_{env}} \end{aligned} \tag{10}$$

where $B(\cdot)$ means the boundary of the objects or obstacles' position. The parameter P_t is the distribution range of particles at the current time, which is shown in section IV-C. Initially, the e_0 is set to a range that could cover the potential position of interested objects.

- **Operation Alignment Factor** F_{ali}: During manipulation, researchers are not concerned with the absolute position of the object being manipulated; rather, they focus on whether the correct contact state has been achieved. Therefore, we define the following operation alignment factor:

$$\|F_{ali}(\theta_{ali})\|^2_{\sum_{ali}} = \|E(F_x \cap F_y \cap d) - \theta_{ali}\|^2_{\sum_{ali}} \tag{11}$$

where F_x, F_y are force components calculated by eqution 1, and d represents the distance. $E(\cdot)$ represents a function used for measuring the error between the current state and the goal state. θ_{ali} is a binary independent variable, if $\theta_{ali} = 1$, the task is considered to be finished.

C. Active Tactile Exploration Policy

In the process of blind manipulation, the exact pose of the object is unknown. To achieve the goal of estimating relative contact regions, the system must generate exploratory actions to interact with the environment, continuously estimate the contact state during the contact process, and adjust the action based on the tactile sensor feedback. Based on these, we draw inspiration from active localization methods [25] and propose the **Active Tactile Exploration Policy(ATEP)** method. The detailed workflow is shown as follows:

Preparation stage: At this stage, given that the exact geometric shapes of the objects in the environment and the grasped object are all represented as polygons, and their vertices $v_1, v_2, ..., v_N$ are collected in counterclockwise order. The contour edges can then be computed as: $e_i = v_{i+1} - v_i$, and the outward normal vector of each contour edge is given by: $n_i = (e_i^y, -e_i^x)$. Thus, we can define the contour of the grasped object and contours of other objects in the environment: $S_{e_i} = \{(n_i, v_i, v_{i+1}) | i = 1, ..., N_i\}$, $S_{l_j} = \{(n_j, v_j, v_{j+1}) | j = 1, ..., M_j\}$. And $C_{env} = \{S_{e_1}, S_{e_2}, ..., S_{e_N}\}$, $C_{object} = \{S_{l_1}, S_{l_2}, ..., S_{l_M}\}$.

After that, to represent the relative position state between the grasped object and the scene objects, the distribution of the reference point within the environment is initialized. In the manuscript, we choose the particle filtering method. The distribution of particles at time t is represented as P_t, and the weight of each particle is represented as $w_i^t = 1/len(P_t)$.

Step 1, Local optimum detection. Examine the weights and choose the local optimum as follows: $peaks = \{p_i^t | p_i^t \in$

$P_t, w_i^t > 0.5/len(P_t), i = 1, 2, ...n\}$. If the boundary measurement of peaks within δ_{thr}, the distribution of peaks is taken as the potential region of the reference point. If not, the process turns to Step 2 to select a movement direction.

Step 2, Information gain evaluation: For each potential motion direction of the object, the expected information gain is calculated, which is represented as the types of contact states and the distance variance of motion distance. The direction with the maximum information gain is then selected as the movement direction. The detailed process is in Algorithm 1.

Algorithm 1: Information Gain Evaluation

Input:

The potential distribution of movable objects: $P_t = peaks$;
The referance actions:$A = \{a_1, a_2, ...a_m\}$;
The contour of environment: $C_{env} = \{S_{e_1}, S_{e_2}, ..., S_{e_N}\}$;
The contour of grasped object: $C_{object} = \{S_{l_1}, S_{l_2}, ..., S_{l_M}\}$;

Output:

for a in $range(A)$ **do**
  for p in $range(Pose)$ **do**
    Calculate the contact state: z_pred at p with action a;
    Add z_pred to Z_a;
    Calculate the distance d to get the state z_pred;
    Add d to D_a;
  end for
end for
Choose the action which could get the most information: $\max_a\{\alpha_1 \cdot entropy(Z_a) + \alpha_2 \cdot variance(D_a)\}$.

Step 3, Exploration through motion: The grasped object moved in the selected direction until a change in the tactile signal is detected.

Step 4, Contact pair selection and updation of particles: After detecting a tactile signal change, according to the method described in SectionIV-A, we can estimate the direction of the force acting on the grasped object. From the perspective of object dynamics, the force direction and the outward normal vector of the object's contour satisfy the following relationship: $F_G^t \cap n_i > 0, F_G^t \cap n_j < 0$. Thus, the contact pairs can be represented as: $(i, j) = \{n_i = -n_j\}$, and the distribution of particels is updated as follows: $P_t = P_{t-1} \cap (B(S_{e_i}) \cup B(S_{l_j}))$.

Step 5, Particles' weights updation: After selecting the candidate contact pairs, we then backtrack the contact events during the motion trajectory. Specifically, based on the traveled distance $Distance$ and the number of motion steps T, we infer the potential contact states at intermediate time steps. If at any of these time steps, the inferred contact state is inconsistent with the observed tactile signal, the weight of the corresponding particle is reduced. This process is repeated until the weights of all particles are updated. The weight w_i^t is updated as follows:

$$w_i^{t+1} = w_i^t \cdot \mathcal{P}(z_{obs}^t, z_{pred}(p_i - (T-t)/T \cdot Distance)) \quad (12)$$

where z_{obs}^t represents the tactile obserbation at the t time, z_{pred} represents the predicted tactile signal at the pose of $p_i - (T-t)/T \cdot Distance$.

The whole **Active Tactile Exploration policy** is shown in Algorithm 2.

Algorithm 2: Active Tactile Exploration Policy

Input:

Scene construction and obtain C_{env}, C_{object};
Initialization of the reference point distribution: $P_t = \{p_1, p_2, ..., p_n\}$, $W_t = \{w_1^t, w_2^t, ..., w_n^t\}$;

Output:

if $len(P_t) < 30$ **then**
  Add more particles to the scene;
end if
Calculate the local optimal particles: $peaks = \{p_i^t | p_i^t \in P_t, w_i^t > 1/len(P_t), i = 1, 2, ...n\}$;
if $len(peaks) > 1$ **then**
  Information gain evaluation and obtain action π;
  Execute π and monitor force until the contact occurs;
  Contact pair selection: $P_{t+1} = P_t \cap (B(S_{e_i}) \cup B(S_{l_j}))$;
  Update the particles' weights;
  return to "Calculate the local optimal particles".
end if
if $len(peaks) < n_{thr}$ and $B(peaks) < \delta_{thr}$ **then**
  Find the optimal trajectory π;
  Execute π, until the alignment factor satisfies $\theta_{ali} = 1$.
end if

V. Experimental Studies

A. Experimental Setup

This manuscript defines two blind manipulation task scenarios, which are illustrated in Fig. 4.

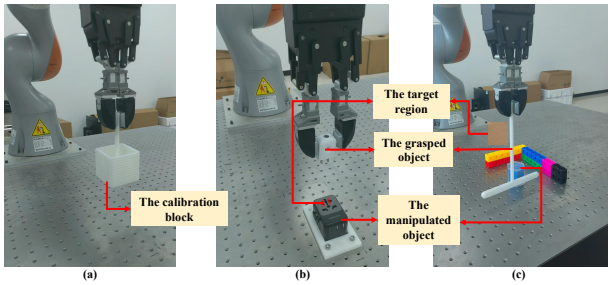

Fig. 4. The calibration task scenario and two manipulation experiment scenarios. (a) represents the calibration process of the Tac3D sensor; (b) represents the socket assembly task scenario; (c) represents the block-pushing task scenario.

The first task scenario involves a blind peg-in-hole assembly of a power socket component of two different standards. The robot is required to grasp the socket and insert it into the correct receptacle. The second task scenario is a blind pushing task in the presence of obstacles. In this scenario, the operator has prior knowledge of the approximate position of the block to be pushed, the target location, as well as the geometrical

shapes of the obstacles present in the environment. However, the exact positions of these obstacles within the scene are unknown.

B. Contact Point Localization of the Grasped Object

For tactile perception, we first perform an unknown-parameter estimation process, and then use the estimated parameters together with Eq. 5 to verify the effectiveness of the proposed contact position prediction method.

In the parameter estimation stage, we design the experimental setup shown in Fig. 4(a). In this setup, a two-finger gripper equipped with Tac3D sensors clamps a calibration block, and each face is uniformly divided into several grooves along both horizontal and vertical directions. During the experiment, several points are selected on each face, and a contact force is applied at each selected point. Using the method described in Eq. 1, we compute the three-dimensional contact force exerted at each point and extract the resultant moment applied at the origin of each sensor coordinate system.

During the data collection stage, the contact point coordinates (C_x, C_y, C_z) in the object coordinate system are assumed to be known. We then adopt the least-squares method to solve for the unknown parameters $\Delta X^{O_e}_{O_{L/R}}, \Delta Y^{O_e}_{O_{L/R}}, \Delta Z^{O_e}_{O_{L/R}}$. In the data validation stage, given new input force and moment data, the estimated parameters are substituted into Eq. 4 and 5 to calculate the coordinates of the contact point. The prediction results of the contact points in the training set are shown in Fig. 5.

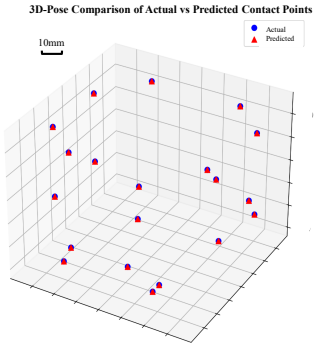

Fig. 5. The predicted contact points vs the actual contact points.

As illustrated in Fig. 5, the fitting errors remain at a very small magnitude, with most errors concentrated around zero and the maximum error not exceeding 0.5mm. These experimental results demonstrate that the proposed tactile-based contact position estimation method is both accurate and effective.

C. Results of the Socket Assembly Experiments

In this experiment, we used two different plugs: a two-pin plug and a triangular three-pin plug, to assemble into the same socket, as illustrated in Fig. 4(b). The main challenge of this task lies in the need to distinguish between different assembly regions corresponding to different plug geometries. In the absence of visual guidance, the robot must rely solely on tactile feedback and prior knowledge of the scene to identify and reason about the relative contact regions between the plug and the socket.

The experimental results are presented in Fig. 6. As can be observed, by leveraging variations in tactile signals and the proposed active exploration framework, the robot is able to achieve incremental recognition of relative contact regions. Through interaction, the robot progressively reduces uncertainty in its understanding of the scene. Ultimately, after $6 \sim 8$ exploration steps, the estimated relative contact distribution between the grasped object and the environment converges from a multi-modal distribution to a uni-modal distribution, thereby enabling the construction of a closed-loop control scheme to successfully accomplish the peg-in-hole assembly task.

After the localization process, the robot analyzed the pose errors. then moved the plug to the target region, and finished the assembly task with a spiral hole search strategy finally.

We futher test the accuracy and efficiency of the proposed contact SLAM method in the above task scenarios. We made each task 5 times, and the results are in Table I and Fig. 7.

TABLE I
THE RESULTS OF SOCKET ASSEMBLY EXPERIMENTS.

	Mean Errors(mm)	**Mean Iteration Number**
two-pin	3.775	7.13
three-pin	1.815	7.67

It could be found in the Table. I that after 6 to 8 iterations, the localization error remained 1 to 5mm, which is related to the error of resampling of particles, as well as the localization error between the gripper and the grasped plug. When re-sampling new particles, in order to cover the accurate pose of the object, the distribution of particles is random in a certain range, causing potential errors. Whatever, during the assembly process, the localization error was reduced gradually, as illustrated in Fig. 7, the Distribution of particles reduced from 30mm to within 5mm.

D. Results of the Blind Pushing Experiments

In this task, the robot must first detect the contact region between the block and the T-shaped tool. When obstacles are encountered, the robot estimates their relative positions, replans the task trajectory accordingly, and ultimately pushes the block into the designated target region. The experimental results are illustrated in Fig. 8.

As shown in Fig. 8, in this task scenario, the robot is capable of detecting the relative contact region between the T-shaped tool and the block using the method described in Section V-B, which enables continuous monitoring of the block's position throughout the pushing process. When obstacles are encountered, the robot is able to reduce the localization error of the obstacles to within 10 mm through a limited number of active explorations and contact interactions, which is sufficient to meet the requirements of trajectory re-planning.

E. Experimental Results Discussion

In the socket assembly experiment scenario, the success of perception depends on the complexity of the contact be-

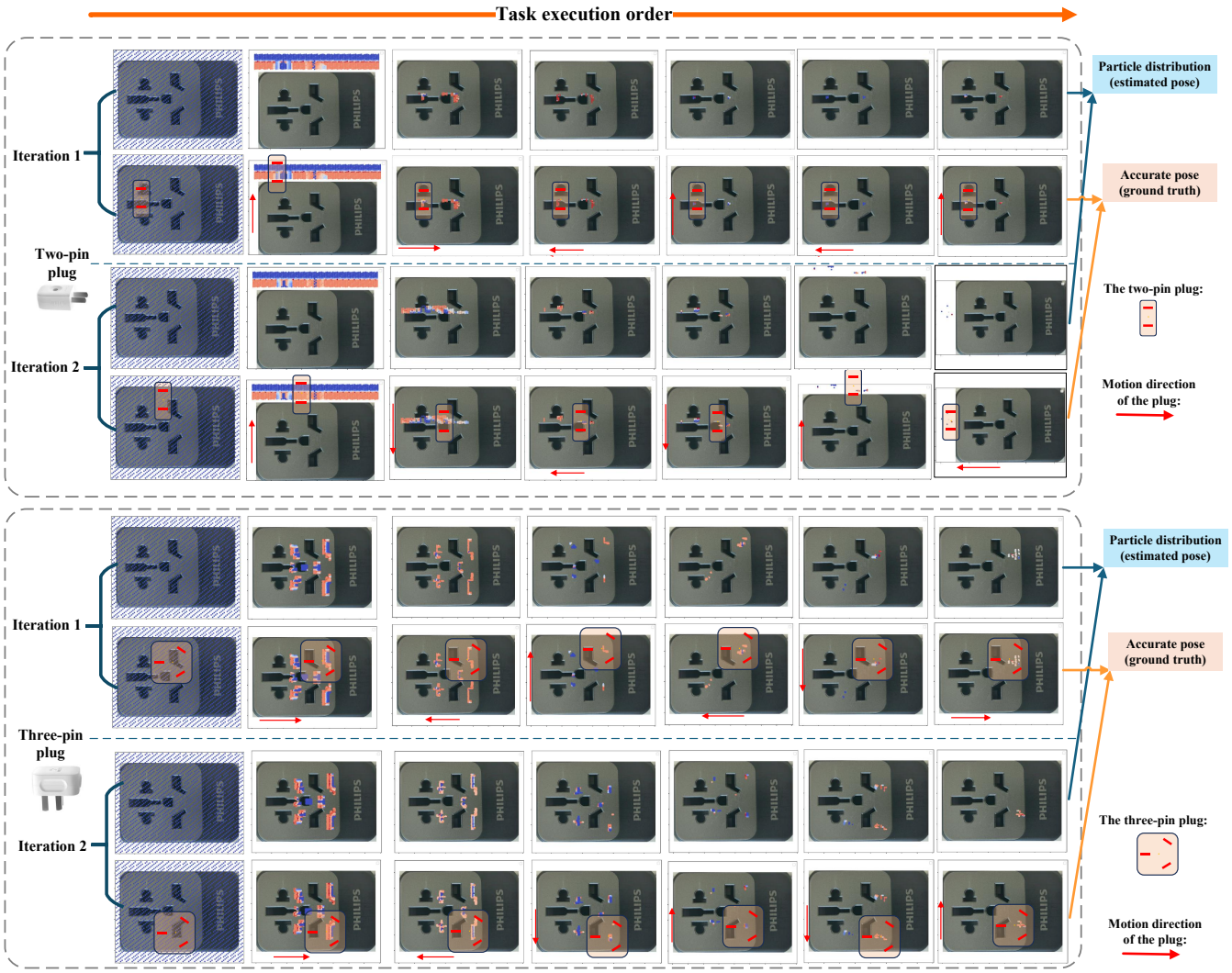

Fig. 6. The results of socket assembly experiments in different initial locations. The distribution of particles represents the potential relative pose of the plug and the socket. Red particles represent that the plug could be located at this point with more possibility, while the blue particles represent the less possibility.

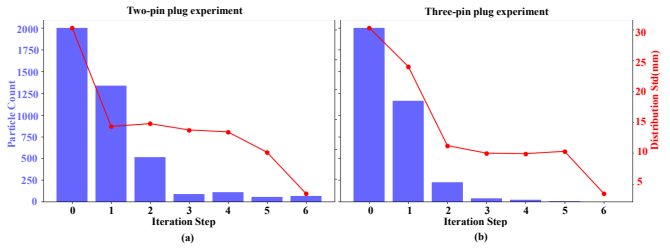

Fig. 7. The Uncertainties of the scenarios during the socket assembly task. (a) represents the two-pin plug experiment, while (b) represents the three-pin plug experiment. Distribution std means the standard deviation of the particles' distribution. Particle count means the number of remaining particles at each iteration step.

tween the plug and the socket. For a two-pin plug, there are two planes in contact with the socket plane. Although the geometric shape of a three-pin plug is more complex, the part in contact with the socket plane is only its longest prong. Therefore, the contact situation of the two-pin plug is more complex, which also affects the number of exploration attempts. The number of exploration attempts for the two-pin plug is higher than that for the three-pin plug.

In the blind block-pushing experiment, it is crucial to distinguish between the movable block and the immovable obstacles. When an obstacle is encountered, the analysis of the particle distribution must take into account the size of the movable block. Compared with the case where the grasped object is in direct contact with the obstacle, the resulting particle distribution exhibits a translation shift.

VI. Conclusion

This manuscript proposed a novel framework named contact SLAM, which could be utilized for fine blind manipulation tasks. The framework integrates tactile perception and physical reasoning to enable scene understanding and motion planning. Given prior knowledge of the objects' geometrical shapes and dimensions, it allows the robot to precisely localize the relative pose between the grasped object and the manipulated object solely through tactile sensing, thereby accomplishing manipulation tasks without relying on visual information. The authors

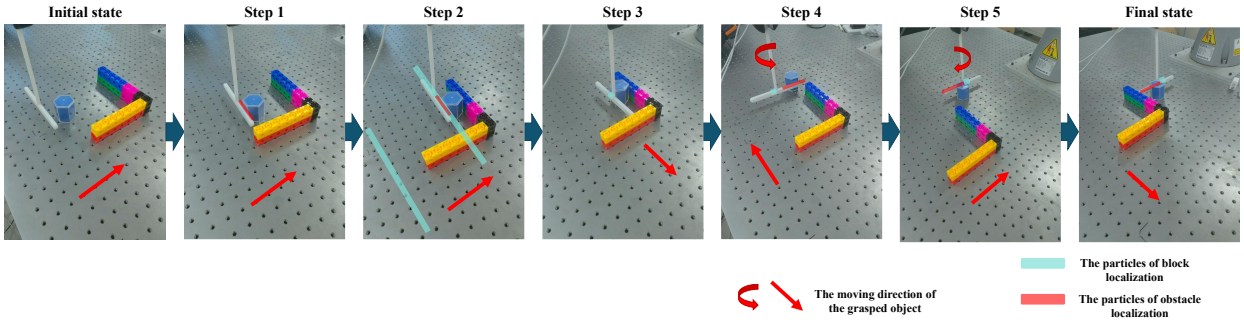

Fig. 8. The detailed results of the block-pushing task. The particle distribution represents the potential position ranges of the block and the obstacles.

validated the effectiveness of the proposed framework through experiments on peg-in-hole assembly and block-pushing tasks. The present study primarily focuses on geometric constraints. In future work, the authors plan to extend this approach to incorporate dynamic constraints in fine manipulation, aiming to further enhance the robot's capability in executing such tasks.

REFERENCES

[1] F. Chaumette and S. Hutchinson, "Visual servo control. i. basic approaches," *IEEE Robotics & Automation Magazine*, vol. 13, no. 4, pp. 82–90, 2006.

[2] H. Park, J. Park, D.-H. Lee *et al.*, "Compliance-based robotic peg-in-hole assembly strategy without force feedback," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 8, pp. 6299–6309, 2017.

[3] M. C. Koval, N. S. Pollard, and S. S. Srinivasa, "Pre-and post-contact policy decomposition for planar contact manipulation under uncertainty," *The International Journal of Robotics Research*, vol. 35, no. 1-3, pp. 244–264, 2016.

[4] A. S. Morgan, Q. Bateux, M. Hao *et al.*, "Towards generalized robot assembly through compliance-enabled contact formations," *arXiv preprint arXiv:2303.05565*, 2023.

[5] D. Ma, S. Dong, and A. Rodriguez, "Extrinsic contact sensing with relative-motion tracking from distributed tactile measurements," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 11 262–11 268.

[6] S. Kim, D. K. Jha, D. Romeres *et al.*, "Simultaneous tactile estimation and control of extrinsic contact," *arXiv preprint arXiv:2303.03385*, 2023.

[7] T. Migimatsu, W. Lian, J. Bohg *et al.*, "Symbolic state estimation with predicates for contact-rich manipulation tasks," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 1702–1709.

[8] J. Lee and N. Fazeli, "Vitascope: Visuo-tactile implicit representation for in-hand pose and extrinsic contact estimation," *arXiv preprint arXiv:2506.12239*, 2025.

[9] J. Xu, H. Lin, S. Song *et al.*, "Tandem3d: Active tactile exploration for 3d object recognition," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 10 401–10 407.

[10] Y. Zhang, Z. Kan, Y. Yang *et al.*, "Effective estimation of contact force and torque for vision-based tactile sensors with helmholtz–hodge decomposition," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4094–4101, 2019.

[11] K.-T. Yu and A. Rodriguez, "Realtime state estimation with tactile and visual sensing. application to planar manipulation," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 7778–7785.

[12] M. Bauza, A. Bronars, and A. Rodriguez, "Tac2pose: Tactile object pose estimation from the first touch," *The International Journal of Robotics Research*, vol. 42, no. 13, pp. 1185–1209, 2023.

[13] Y. Zhao, X. Jing, K. Qian *et al.*, "Skill generalization of tubular object manipulation with tactile sensing and sim2real learning," *Robotics and Autonomous Systems*, vol. 160, p. 104321, 2023.

[14] M. Bauza, A. Bronars, Y. Hou *et al.*, "Simple, a visuotactile method learned in simulation to precisely pick, localize, regrasp, and place objects," *Science Robotics*, vol. 9, no. 91, p. eadi8808, 2024.

[15] A. Dutta, E. Burdet, and M. Kaboli, "Push to know!-visuo-tactile based active object parameter inference with dual differentiable filtering," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 3137–3144.

[16] S. Suresh, M. Bauza, K.-T. Yu *et al.*, "Tactile slam: Real-time inference of shape and pose from planar pushing," in *2021 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2021, pp. 11 322–11 328.

[17] J. Zhao, M. Bauza, and E. H. Adelson, "Fingerslam: Closed-loop unknown object localization and reconstruction from visuo-tactile feedback," *arXiv preprint arXiv:2303.07997*, 2023.

[18] P. Sodhi, M. Kaess, M. Mukadam *et al.*, "Learning tactile models for factor graph-based estimation," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 13 686–13 692.

[19] S. Kim and A. Rodriguez, "Active extrinsic contact sensing: Application to general peg-in-hole insertion," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 10 241–10 247.

[20] C. Zhang, P. Hao, X. Cao *et al.*, "Vtla: Vision-tactile-language-action model with preference learning for insertion manipulation," *arXiv preprint arXiv:2505.09577*, 2025.

[21] W. Li, H. Zhao, Z. Yu *et al.*, "Pin-wm: Learning physics-informed world models for non-prehensile manipulation," *arXiv preprint arXiv:2504.16693*, 2025.

[22] M. Noseworthy, B. Tang, B. Wen *et al.*, "Forge: Force-guided exploration for robust contact-rich manipulation under uncertainty," *IEEE Robotics and Automation Letters*, 2025.

[23] J. Bi, K. Y. Ma, C. Hao *et al.*, "Vla-touch: Enhancing vision-language-action models with dual-level tactile feedback," *arXiv preprint arXiv:2507.17294*, 2025.

[24] H. Xue, J. Ren, W. Chen *et al.*, "Reactive diffusion policy: Slow-fast visual-tactile policy learning for contact-rich manipulation," *arXiv preprint arXiv:2503.02881*, 2025.

[25] S. Agarwal, A. Tamjidi, and S. Chakravorty, "Motion planning in non-gaussian belief spaces (m3p): The case of a kidnapped robot," *arXiv preprint arXiv:1506.01780*, 2015.