Simultaneous Extrinsic Contact and In-Hand Pose Estimation via Distributed Tactile Sensing

Mark Van der Merwe[1], Kei Ota[2], Dmitry Berenson[1], Nima Fazeli[1], and Devesh K. Jha[2]

***Abstract*—Prehensile autonomous manipulation, such as peg insertion, tool use, or assembly, require precise in-hand understanding of the object pose and the extrinsic contacts made during interactions. Providing accurate estimation of pose and contacts is challenging. Tactile sensors can provide local geometry at the sensor and force information about the grasp, but the locality of sensing means resolving poses and contacts from tactile alone is often an ill-posed problem, as multiple configurations can be consistent with the observations. Adding visual feedback can help resolve ambiguities, but can suffer from noise and occlusions. In this work, we propose a method that pairs local observations from sensing with the physical constraints of contact. We propose a set of factors that ensure local consistency with tactile observations as well as enforcing physical plausibility, namely, that the estimated pose and contacts must respect the kinematic and force constraints of quasi-static rigid body interactions. We formalize our problem as a factor graph, allowing for efficient estimation. In our experiments, we demonstrate that our method outperforms existing geometric and contact-informed estimation pipelines, especially when only tactile information is available. Video results can be found at tacgraph.github.io.**

***Index Terms*—In-Hand Manipulation, Force and Tactile Sensing, Perception for Grasping and Manipulation**

I. Introduction

PREHENSILE manipulation tasks require precise reasoning over grasped object poses and contacts. Even small errors in object pose can prevent proper insertion of an object [1] or yield an undesired placement [2]. Effective tool use similarly relies on precise application of contacts and forces [3]. We require systems capable of simultaneously estimating where a grasped object is and how it is in contact with the environment.

Methods for object pose estimation largely rely on visual feedback [4, 5]. Prehensile manipulation, however, often suffers visual occlusions due to the grasp and/or environment and sensor noise can corrupt estimation results. Distributed visuo-tactile sensors are a promising form of feedback for prehensile manipulation [6, 7] that can complement visual sensing [8]. These sensors utilize a compliant material that deforms on contact and a camera to observe these deformations. This provides local geometric information where the sensor contacts the object. While these geometric observations can resolve in-hand pose for small or highly featured objects [9], for many objects, the local nature of these observations still results in significant ambiguities [10] and must be paired with visual feedback for better convergence [11].

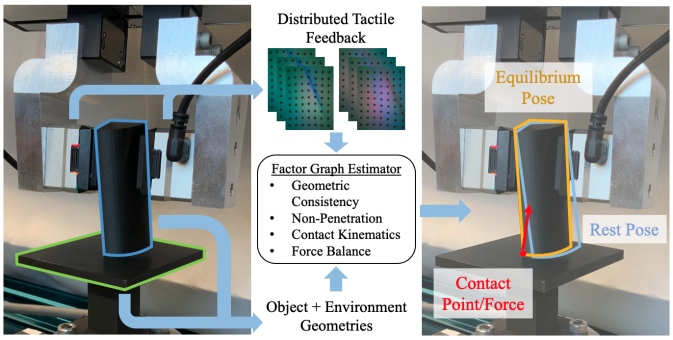


Fig. 1: We propose *TacGraph*, an estimator that exploits geometric consistency, force balance, non-penetration, and contact kinematics to jointly estimate the object pose and extrinsic contacts.

The compliance of the visuo-tactile sensor can also provide a signal of extrinsic contact, between the grasped object and the environment. How the sensor deforms when an extrinsic contact is made is indicative of the force experienced in the grasp [12], which in turn can be used as a signal to infer extrinsic contacts [13, 14]. Precisely identifying contact is still challenging, however, as multiple contact points can yield near-identical force profiles in the grasp.

In this work, we investigate how *jointly* estimating the object pose and extrinsic contacts can help resolve ambiguities and address noise. Intuitively, these two are tightly coupled, as the object pose determines which contacts are possible, and conversely, where contact is being made constrains object poses. We enforce non-penetration with the (assumed known) environment and ensure that the estimated extrinsic contact is kinematically feasible (i.e., lies on the surface of the object and environment) and yields forces consistent with tactile feedback. The result is a system that enforces consistency across multiple forms of feedback (geometry and forces) while enforcing physical realism (see Fig. 1). This reduces the space of acceptable solutions, resulting in accurate estimation.

Our contributions are as follows:

- A set of *object agnostic* models for extracting geometric and force signals for our estimator from tactile feedback.
- A factor graph method for simultaneous extrinsic contact and in-hand pose estimation, *TacGraph*, which jointly enforces geometric consistency, force balance, non-penetration, and contact kinematics.

This work was supported in part by the Office of Naval Research Grant N00014-24-1-2036 and NSF grants IIS-2113401, IIS-2231607, and IIS-2220876.

[1]Mark Van der Merwe, Dmitry Berenson, and Nima Fazeli are with Robotics Department, University of Michigan, USA markvdm@umich.edu, dmitryb@umich.edf, nfz@umich.edu

[2]Kei Ota and Devesh K. Jha are with Mitsubishi Electric Research Laboratories, USA ota@merl.com, jha@merl.com

- Demonstration of our method on a physical system and comparison to baselines.

II. Related Work

A. Extrinsic Contact Estimation

Accurate recovery of extrinsic contacts is challenging due to the broad nature of contacts possible and the indirect and partial sensing available. Kim et al. [15] learns to predict image contact masks directly from visual feedback. Other methods learn to predict line or patch geometries from point cloud and Force/Torque feedback [16, 17]. Higuera et al. [18] and Ota et al [19] predict extrinsic contact patches based on distributed visuo-tactile feedback. Lee et al.[20] predicts extrinsic contact patch along with in-hand pose. These methods rely on expensive object-specific training, unlike our method which trains only object-agnostic components.

Kim et al. [14] and Kim et al. [13] utilize visuo-tactile feedback to estimate an extrinsic contact. Without knowledge of object geometry, their method utilizes the in-hand displacements observed from tactile sensing and active exploratory interactions to derive contact line or point. However, this work does not consider object pose.

B. Prehensile Object Pose Estimation

Tac2Pose [10] learns to perform object pose estimation via an object-specific tactile model that returns pose distributions consistent with observed tactile feedback. Follow up work fuses incorporates visual feedback to further resolve ambiguity [11]. Dikhale et al. [21] fuse tactile and visual feedback to learn to predict object poses directly, akin to purely visual based tracking models [5].

Several works investigate model-based estimation with combined visual and tactile feedback. Several works jointly reconstruct and estimate pose from visual and tactile geometric feedback [22, 8]. Zhong et al. [23] extends model-based pose estimation to include free-space non-penetration constraints as well as tactile point clouds, but does not consider contact consistency of any form.

C. Joint Prehensile Pose and Extrinsic Contact Estimation

Bronars et al. [24] extends Tac2Pose [10] to include additional geometric contact constraints, however, it does not utilize the wrench consistency and relies on the object-specific models of Tac2Pose.

SCOPE [25] and Multi-Scope [26] propose a model-based approach that utilizes Force/Torque feedback on the robot and environment to simultaneously estimate object and environment pose and extrinsic contacts, utilizing the physical constraints of contact. However, assuming F/T sensing on the environment is often unrealistic and their approach does not incorporate geometric consistency.

III. Problem Statement

Our goal is to estimate the pose of a grasped object and the extrinsic contact that the grasped object makes with a known environment. We make the following assumptions:

- The object is rigid and has a known geometry $\mathcal{M}_o$.
- The environment is rigid, has a known geometry $\mathcal{M}_e$, and is static.
- The grasp is elastic, meaning it complies due to external contact, but the object returns to the same location when the extrinsic contact is removed.
- We assume that the extrinsic contact can be described as a single summary contact point and force, and does not induce a torque.

We utilize feedback from a pair of Gelsight tactile sensors [9], located at the grasp, and robot proprioception. For some experiments, we additionally provide visual feedback from an external camera. Our inputs are:

- (Optional) $\boldsymbol{P}^V \in \mathbb{R}^{N_V \times 3}$ - initial partial point cloud of object from external camera.
- $\boldsymbol{g}_t \in SE(3)$ - gripper pose at time t.
- $\boldsymbol{I}_t^L, \boldsymbol{I}_t^R \in \mathbb{R}^{H \times W \times 3}$ - Gelsight tactile images from the left and right gripper fingers at time t.
- $\mathcal{M}_o, \mathcal{M}_e$ - triangle mesh geometry of the grasped object and environment.

Our goal is to estimate the object pose as well as the contact point and contact force. Our desired outputs are:

- $\boldsymbol{o}_t \in SE(3)$ - object pose at time t.
- $\boldsymbol{c}_t \in \mathbb{R}^3$ - contact point at time t.
- $\boldsymbol{f}_t \in \mathbb{R}^3$ - contact force at $\boldsymbol{c}_t$ at time t.

IV. TacGraph: Factor Graph Based Estimation

We propose our method for simultaneously estimating in-hand object pose and extrinsic contacts from tactile feedback. First, we propose a set of object agnostic tactile models for extracting useful geometric and force signals to be used downstream. Second, we propose TacGraph, our factor-graph based estimator. An overview of our method is shown in Fig. 2.

A. Tactile Models

Distributed visuo-tactile sensors provide several forms of feedback that are valuable for contact-rich prehensile manipulation. First, we gain local geometric information from the grasp. Second, the deformation allowed by the sensor upon an external contact provides information about a) the in-hand displacement of the object and b) the force applied on the object. We propose a set of *object-agnostic* models that extracts these feedback terms from the raw tactile images.

We use the initial tactile observations (i.e., at $t = 0$), to predict a tactile point cloud $\boldsymbol{P}^T \in \mathbb{R}^{N_T \times 3}$. We train a model that takes in a tactile image (either from the left or right sensor) and yields a depth image $D \in \mathbb{R}^{H \times W}$. We threshold the depth to determine which pixels are in contact and de-project the depths to yield our final point cloud $\boldsymbol{P}^T$.

As the grasped object makes contact with its environment, the compliance of the Gelsight sensor means the object moves in the grasp. In order to accurately recover the object pose, we must then reason about this displacement. We train an in-hand displacement model which takes in both tactile images and predicts $\boldsymbol{\delta}_t \in SE(3)$, the relative displacement of the object in-hand.

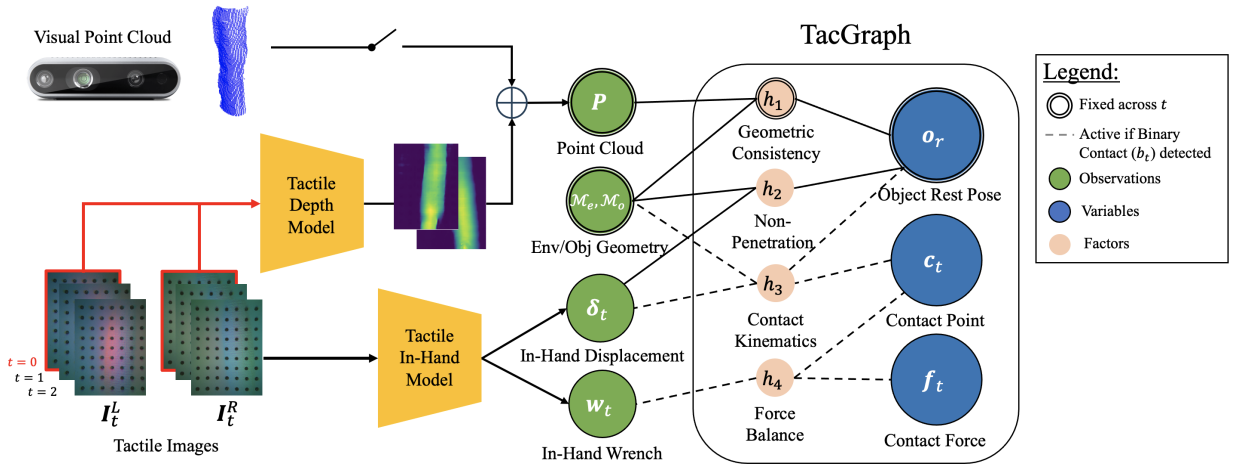


Fig. 2: Overview of our proposed methodology. First, we propose a set of tactile models which process raw distributed tactile observations into geometric and force feedback terms. Second, these terms are utilized, along with known geometries and optionally with visual feedback, in a factor-graph based estimator, *TacGraph*, which estimates the object pose and extrinsic contacts. Observations, variables, and factors that are fixed (non time-varying) are double circled. Factors only active when contact is detected are connected with dashed lines.

Finally, we wish to determine the in-hand wrench experienced at the grasp, which is helpful for contact localization and for resolving the contact force [27, 25]. We train an additional model that takes in both tactile images and predicts $\boldsymbol{w}_t \in \mathbb{R}^6$, the wrench experience at the grasp. Additionally, we can utilize our predicted wrench to estimate whether the system is in contact at time t: $b_t = ||\boldsymbol{w}_t||_\Sigma > \epsilon$.

B. TacGraph

Our goal is to incorporate our tactile feedback terms along with the physical constraints of contact to determine the *maximum a posteriori* (MAP) estimate of our variables. As stated in Sec. III, we aim to recover the object pose and extrinsic contact information at each time t.

We assume that the grasp is elastic, that is, the object moves in-hand due to sensor compliance upon the application of an external force, and returns to the same rest pose when the external force is removed. As described in Sec. IV-A, we estimate this in-hand displacement $\boldsymbol{\delta}_t$ from the tactile feedback. As such, we perform a change of variables and infer $\boldsymbol{o}_t$ by estimating a single *rest pose* $\boldsymbol{o}_r \in SE(3)$ and applying the predicted in-hand displacement at a given time t.

$$\boldsymbol{o}_t = \boldsymbol{g}_t{}^g \boldsymbol{\delta}_t{}^g \boldsymbol{o}_r \tag{1}$$

We use the prefix g to indicate that the displacement and rest pose are both expressed in the gripper pose frame, but drop the frames from here on.

A natural way to describe our MAP estimation problem is as a Factor Graph, a bi-partite graph of variables and the factors which describe the relationship between variables [28]. Assuming Gaussian noise models on the factors, the MAP estimation becomes a sum of nonlinear least-squares.

$$\boldsymbol{o}_r^*, \boldsymbol{c}_{1:T}^*, \boldsymbol{f}_{1:T}^* = \underset{\boldsymbol{o}_r, \boldsymbol{c}_{1:T}, \boldsymbol{f}_{1:T}}{\arg\min}\ H(\boldsymbol{o}_r, \boldsymbol{c}_{1:T}, \boldsymbol{f}_{1:T}) \tag{2}$$

$$\begin{aligned} H(\cdot) = &||h_1(\boldsymbol{o}_r)||^2_{\Sigma_1} + \sum_{t=1}^{T}\{||h_2(\boldsymbol{o}_r)||^2_{\Sigma_2} + \\ &\mathbf{1}[b_t](||h_3(\boldsymbol{o}_r, \boldsymbol{c}_t)||^2_{\Sigma_3} + ||h_4(\boldsymbol{c}_t, \boldsymbol{f}_t)||^2_{\Sigma_4})\} \end{aligned} \tag{3}$$

The structure of the factor graph is shown in Fig. 2 in the box. Note, b_t is an observation (see Sec. IV-A), not a variable being solved for, and is used to enable contact-specific factors only when the object is actively in contact. Each factor h_i has an accompanying covariance matrix Σ_i which is empirically selected. The resulting MAP problem can be solved efficiently using the iSAM2 solver [29].

C. Factors

1) Geometric Consistency: Our first factor ensures the estimated rest pose $\boldsymbol{o}_r$ is consistent with the observed object point cloud at time $t = 0$. This point cloud $\boldsymbol{P}$ includes the tactile point cloud $\boldsymbol{P}^T$ and, if available, a visual point cloud $\boldsymbol{P}^V$. As we assume that the grasp is elastic, it is sufficient to ensure the rest pose is geometrically consistent, without enforcing at each time step.

We define our factor error function as the signed-distance value of each point of the observed point cloud to the surface of the object:

$$h_1(\boldsymbol{o}_r; \boldsymbol{P}, \mathcal{M}_o) = \boldsymbol{S} \tag{4}$$

Here, $\boldsymbol{S}$ is a vector containing the signed distance value of each point to the surface.

$$\boldsymbol{S}_i = SDF(\boldsymbol{o}_r^{-1}\boldsymbol{P}_i^T | \mathcal{M}_o) \tag{5}$$

SDF is the signed distance value computed using the geometry $\mathcal{M}_o$. We transform each point into the object frame using the estimated $\boldsymbol{o}_r$. Both the SDF and the SDF gradients can be efficiently computed using a triangle mesh geometry [23].

2) Non-Penetration: We ensure the estimated pose at each time-step does not yield penetration with the environment geometry. We apply Eq. 1 to get the estimated object pose, considering the observed gripper pose and in-hand displacement, as well as the current estimate of the rest pose.

Given the known geometries $\mathcal{M}_e, \mathcal{M}_o$ and the object pose, we then compute a penetration check. In practice, we approximate our object geometry for this factor as a point cloud $\boldsymbol{P}^{obj} \in \mathbb{R}^{N_P \times 3}$ by sampling points on the surface of the geometry $\mathcal{M}_o$. Our factor error function then returns for each point the distance to the surface of the environment, if the point is inside of the environment geometry, thus indicating penetration.

$$h_2(\boldsymbol{o}_r|\mathcal{M}_o, \mathcal{M}_e, \boldsymbol{g}_t, \boldsymbol{\delta}_t) = \boldsymbol{S} \tag{6}$$

Once again $\boldsymbol{S}$ is a vector which contains the SDF terms when a point is in penetration.

$$\boldsymbol{S}_i = \min(0, SDF(\boldsymbol{g}_t \boldsymbol{\delta}_t \boldsymbol{o}_r \boldsymbol{P}_i^{obj} | \mathcal{M}_e)) \tag{7}$$

Each point is transformed into the world frame before evaluating the SDF against the environment geometry.

3) Contact Kinematics: When we detect that contact has been made b_t, we add additional variables and factors to the graph at that time step to estimate the contact location and force. First, we address the kinematic constraints on the contact. Namely, the contact point $\boldsymbol{c}_t$ should lie on the surface of *both* the environment and object geometries. This amounts to the following error function:

$$h_3(\boldsymbol{o}_r, \boldsymbol{c}_t; \mathcal{M}_e, \mathcal{M}_o) = \begin{bmatrix} SDF(\boldsymbol{c}_t|\mathcal{M}_e) \\ SDF((\boldsymbol{g}_t \boldsymbol{\delta}_t \boldsymbol{o}_r)^{-1} \boldsymbol{c}_t | \mathcal{M}_o) \end{bmatrix} \tag{8}$$

We again apply Eq. 1 to derive the object pose and invert to map the point to the object frame.

4) Force Balance: In Sec. IV-A, we showed how we can recover the wrench experienced at the grasp $\boldsymbol{w}_t$ from our tactile sensors. This in-hand wrench is the result of the application of an external contact force. As such, we add a factor which ensures that the in-hand wrench implied by the contact matches our $\boldsymbol{w}_t$ observation.

To estimate the in-hand wrench applied by a contact force $\boldsymbol{f}_t$ applied at $\boldsymbol{c}_t$, we apply the contact Jacobian to map the force from the contact point to the gripper frame.

$$\hat{\boldsymbol{w}}_t = J(\boldsymbol{g}_t^{-1} \boldsymbol{c}_t)\ \boldsymbol{f}_t \tag{9}$$

We can then define our factor error function as the difference between the observed and predicted in-hand wrench.

$$h_4(\boldsymbol{c}_t, \boldsymbol{f}_t; \boldsymbol{g}_t) = \hat{\boldsymbol{w}}_t - \boldsymbol{w}_t \tag{10}$$

D. Inference

Solving for the MAP estimate with our proposed TacGraph enables us to find the most likely rest pose $\boldsymbol{o}_r$ and contact points $\boldsymbol{c}_{1:T}$ and forces $\boldsymbol{f}_{1:T}$. We then apply Eq. 1 to resolve our object poses at each timestep $\boldsymbol{o}_{1:T}$.

The non-linear nature of our system means that the minimization performed by iSAM2 in Eq. 2 is subject to local minimums. As such, we initialize a set of particles to represent possible rest poses $\{\boldsymbol{o}_r^1, \boldsymbol{o}_r^2, \ldots, \boldsymbol{o}_r^K\}$. We then perform our inference procedure on each rest pose particle separately, resulting in K solutions. We utilize our factor graph cost in Eq. 3 to score each solution particle, and select our final solution accordingly:

$$\boldsymbol{o}_r^*, \boldsymbol{c}_{1:T}^*, \boldsymbol{f}_{1:T}^* = \arg\min_k H(\boldsymbol{o}_r^k, \boldsymbol{c}_{1:T}^k, \boldsymbol{f}_{1:T}^k) \tag{11}$$

We utilize Iterative Closest Point (ICP) to match our initial rest pose particles to the available point cloud $\boldsymbol{P}$. We apply a sampling heuristic based on likely grasp directions [10] to ensure that the rest pose particles cover the space of possible initializations well.

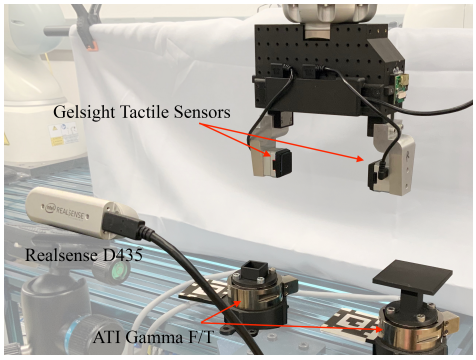


Fig. 3: Our experimental setup. On the left ATI Gamma is an example object fixture. On the right ATI Gamma is the sensorized press surface we use for data collection/experiments.

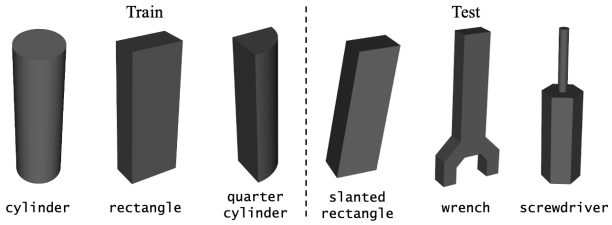


Fig. 4: Train/Test objects used in our experiments.

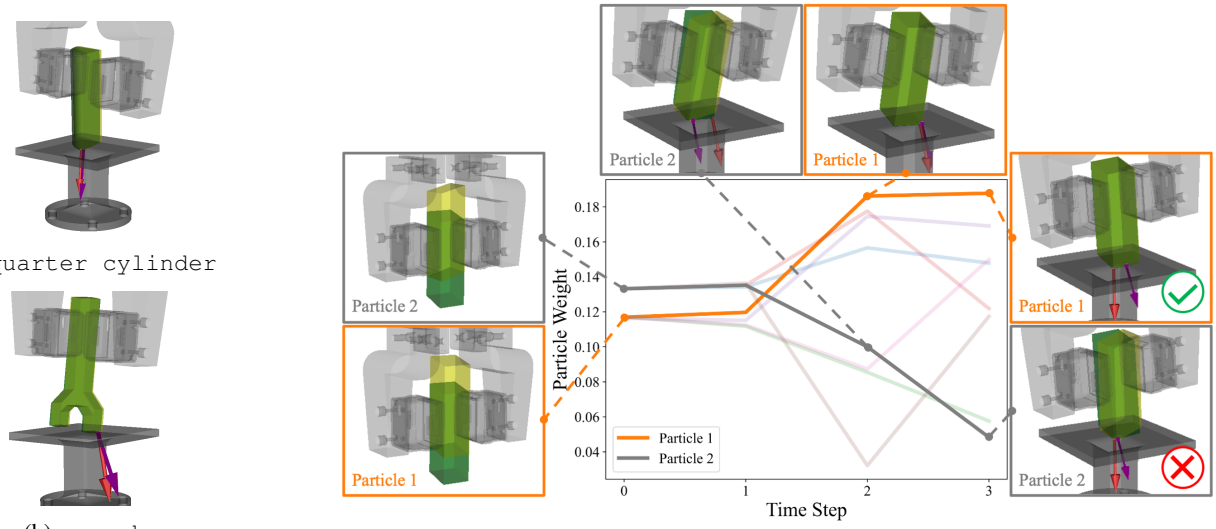
V. Implementation

A. Experimental Setup

Our experimental setup is shown in Fig. 3. We use a WSG-50 parallel jaw gripper attached to a KUKA LBR iiwa Med R820 robot. We attach a GelSight Mini sensor to each finger. We have an external Intel Realsense D435 sensor for visual feedback. We utilize Segment Anything Model for visual point cloud segmentation [30]. Two ATI Gamma Force/Torque sensors are mounted in the scene and used to attach fixtures and environment geometries. We utilize these extrinsic F/T sensors *only* for training our tactile models and evaluation; our method does not utilize these sensors' feedback. Our tactile models are implemented using Pytorch and our factor graph is implemented using GTSAM [31].

B. Tactile Model Training

We train our tactile models described in Sec. IV-A utilizing datasets collected on our physical system, using three different



(a) `quarter cylinder`

(b) `wrench`

(c) Particle Weight Progression

Fig. 5: We show comparison of predicted and ground truth object pose, and predicted and ground truth extrinsic contact. (a-b) final qualitative TacGraph estimates for two different objects. (c) progression of particle weights within TacGraph for a tactile-only inference. Two highlighted particles indicate how initial orientations can be filtered based on the contacts made to correctly select particle solutions. Figure best viewed in color.

train objects (see Fig. 4). All models are convolutional models trained via supervised learning.

1) Tactile Depth: To generate tactile images with ground truth depth labels, we rigidly attach fixtures of known geometry to our environment and apply random grasps to the fixture, adjusting the position and orientation of the grasp. Then using the known fixture geometry, we render corresponding depth images.

2) In-Hand Displacement: To generate in-hand displacement labels, we rigidly attach fixtures of known geometry to our environment. Following Kim and Rodriguez [14], we randomly grasp the fixture, then apply small delta motions to the end effector. These delta motions become the in-hand displacement labels $\boldsymbol{\delta}_t$ associated with the tactile images.

3) In-Hand Wrench: To generate in-hand wrench labels, we utilize the tabletop fixture shown in Fig. 3 and grasp objects, with randomized grasps, from a fixture before performing a random poke into the tabletop. We then use the environment mounted F/T sensor to get the wrench caused by the poke $\boldsymbol{w}_t^e$. We transform this wrench into the grasp frame, recovered from the robot proprioception, to label the wrench.

VI. Experiments

A. Baselines

- **ICP:** Iterative Closest Point enforces geometric consistency in the object pose. To estimate a contact point and force, we use our tactile models to detect when contact occurs as well as the force. When in contact, we select the point on the object closest to the environment as our contact point, assigning the predicted force from the tactile model.
- **CHSEL [23]:** CHSEL adds free-space reasoning as well as a Quality Diversity (QD) to gradient-based pose registration. CHSEL thus considers geometric consistency and non-penetration via free points. At each timestep, we sample points from the surface of the environment, transform into the grasp frame, and add as free points in the CHSEL optimization, to ensure the object does not collide. We follow the same procedure as ICP to label extrinsic contact. We run 2 CHSEL iterations per update.
- **SCOPE [26]:** SCOPE utilizes non-penetration, contact kinematics, and force balance for pose and contact estimation via a dual particle filter method. Object pose particles are scored based on non-penetration and their contacts, determined by a Contact Particle Filter (CPF) [27] associated for each object pose particle to estimate the most likely contacts. We modify SCOPE to fit our setup by removing environment pose filtering and replace the F/T sensing with our learned tactile model. Note, this method does not consider geometric consistency. We run two versions of SCOPE. SCOPE (v1) uses 8 pose and 30 contact particles, with 3 iterations every update. SCOPE (v2) uses 64 pose and 200 contact particles, with 20 iterations every update.

All methods are initialized with the same procedure as TacGraph for fair comparison.

B. Pose and Contact Estimation

We test pose and contact estimation performance on the objects shown in Fig. 4. For the environment geometry, we use the geometry shown on the right ATI Gamma F/T sensor in Fig. 3. Each object is grasped from a fixture, randomizing the in-hand grasp pose. We perform a fixed set of three angled pokes into the environment. We get sensor feedback before poking and once during each poke, when contact is made. We run inference for each method under two conditions: Vision + Tactile and Tactile. We apply each method iteratively on each subsequent timestep of data.

	Methods	Train			Test		
		cylinder	rectangle	quarter cylinder	slanted rectangle	wrench	screwdriver
Vision + Tactile	ICP	3.08 (0.65)	3.68 (0.73)	3.09 (1.08)	4.86 (4.81)	2.19 (0.76)	3.33 (1.25)
	CHSEL	1.04 (0.36)	1.97 (0.39)	1.23 (0.51)	12.28 (5.11)	1.66 (0.88)	2.44 (0.83)
	SCOPE (v1)	5.96 (2.83)	7.11 (2.70)	16.50 (2.69)	15.56 (4.36)	5.86 (1.68)	7.34 (2.53)
	SCOPE (v2)	4.77 (2.50)	2.59 (1.22)	14.04 (5.55)	13.28 (4.79)	3.45 (1.57)	5.63 (3.11)
	TacGraph	0.68 (0.23)	1.48 (0.14)	1.34 (0.34)	1.05 (0.32)	0.92 (0.28)	1.62 (0.22)
Tactile	ICP	17.80 (13.86)	17.15 (10.66)	23.70 (7.58)	20.10 (6.94)	16.77 (6.03)	12.32 (7.87)
	CHSEL	30.33 (12.01)	28.55 (17.20)	28.40 (6.86)	20.71 (7.09)	13.61 (8.16)	29.42 (17.20)
	SCOPE (v1)	9.01 (6.49)	12.26 (12.38)	16.13 (10.33)	18.44 (5.08)	9.89 (7.39)	8.32 (3.85)
	SCOPE (v2)	4.16 (1.00)	4.37 (2.78)	10.74 (5.37)	12.90 (4.94)	4.75 (3.77)	11.03 (5.31)
	TacGraph	2.96 (2.54)	1.29 (2.24)	8.45 (5.70)	7.58 (6.18)	0.78 (0.40)	1.54 (1.29)

TABLE I: Quantitative results for pose estimation. Mean and std. dev. of Averaged 3D Distance (ADD) in mm reported; best for each method in Vision+Tactile and Tactile regime highlighted.

	Methods	Train			Test			Overall
		cylinder	rectangle	quarter cylinder	slanted rectangle	wrench	screwdriver	
Vision + Tactile	ICP	3.43 (0.80)	4.77 (2.10)	4.33 (2.48)	5.30 (5.68)	2.61 (1.34)	4.38 (1.17)	4.14 (2.92)
	CHSEL	1.84 (1.02)	3.15 (2.72)	2.79 (2.62)	12.59 (9.44)	2.44 (1.41)	3.15 (0.55)	4.33 (5.62)
	SCOPE (v1)	5.43 (2.33)	9.59 (8.45)	6.33 (3.09)	10.97 (4.51)	4.84 (3.53)	36.51 (14.33)	12.28 (13.28)
	SCOPE (v2)	4.64 (2.48)	8.71 (8.52)	5.33 (3.38)	9.48 (5.35)	3.75 (1.87)	27.10 (17.32)	9.83 (11.59)
	TacGraph	2.78 (1.31)	4.79 (4.63)	1.93 (1.06)	7.03 (6.11)	2.29 (1.13)	3.29 (1.20)	3.68 (3.72)
Tactile	ICP	18.20 (13.49)	17.38 (10.98)	18.65 (8.92)	20.71 (8.19)	17.06 (5.54)	13.20 (7.70)	17.53 (9.75)
	CHSEL	25.16 (14.40)	32.66 (14.68)	22.54 (9.46)	21.15 (10.17)	13.54 (8.08)	31.41 (18.30)	24.41 (14.52)
	SCOPE (v1)	8.19 (8.31)	15.49 (12.55)	9.25 (12.25)	16.36 (8.89)	11.15 (8.39)	36.74 (13.21)	16.20 (14.49)
	SCOPE (v2)	4.73 (2.25)	11.16 (10.65)	3.72 (2.59)	9.34 (5.32)	6.98 (5.35)	26.49 (13.63)	10.40 (10.93)
	TacGraph	3.29 (1.52)	2.60 (2.02)	3.82 (2.55)	7.63 (7.03)	2.34 (0.85)	2.64 (1.69)	3.72 (3.78)

TABLE II: Quantitative results for contact point estimation. Mean and std. dev. of distance to G.T. contact point in mm reported; best for each method in Vision+Tactile and Tactile regime highlighted.

Methods	Vision + Tactile	Tactile
ICP	0.68 (0.39)	0.68 (0.39)
CHSEL	0.68 (0.39)	0.68 (0.39)
SCOPE (v1)	0.61 (0.37)	0.63 (0.44)
SCOPE (v2)	0.62 (0.37)	0.61 (0.41)
TacGraph	0.61 (0.36)	0.61 (0.36)

TABLE III: Summary Quantitative results for contact force estimation. Mean and std. dev. of difference to G.T. contact force in Newtons reported; best for each method in Vision+Tactile and Tactile regime highlighted.

1) Metrics: We determine the ground truth pose of the object using the fixture (tolerance of $\sim$1mm). We use the Average 3D Distance metric [10] using 1000 points as our object pose metric.

For the extrinsic contact, we utilize the F/T sensor mounted under the environment. We compute the ground truth contact by sampling a set of candidate contact points on the environment surface $\boldsymbol{C} \in \mathbb{R}^{L\times 3}$. We then identify the ground truth by solving the following minimization:

$$\boldsymbol{c}_t^* = \arg\min_l ||\boldsymbol{\tau}_t^{FT} - \boldsymbol{C}_l \times \boldsymbol{f}_t^{FT}||_2 \tag{12}$$

$\boldsymbol{f}_t^{FT}$ and $\boldsymbol{\tau}_t^{FT}$ are the force and torque at the sensor. The ground truth force is then $\boldsymbol{f}_t^* = \boldsymbol{f}_t^{FT}$. We compute the euclidean errors on the contact point and force estimate.

2) Results: We report our pose estimation results in Table I. When vision is available, ICP can perform well, but its performance is subject to sensor noise. CHSEL can rectify some noise via its non-penetration handling. Neither method works well on tactile-only, as they cannot exploit contact to improve. SCOPE outperforms ICP and CHSEL on tactile only, as it can exploit the contact information, but lack of geometric feedback and noise in the filtering reduce precision. We find that TacGraph consistently outperforms the baselines across nearly all cases, exploiting contact to inform the object pose.

Table II and Table III show the contact estimation performance. When Vision and Tactile are available, we find that geometric based methods perform comparably to ours. However, when vision is removed the contact estimate quickly drifts due to the pose error. In contrast, our method retains similar overall contact estimate performance. Since all methods utilize the learned tactile force model, force estimates differ only slightly - we can see that TacGraph can improve force estimates, due to improved contact location and torque feedback at the grasp.

Fig. 5 shows qualitative predictions by TacGraph. In Fig. 5c we show the progression of K particles by their weight (i.e., $e^{-H(\cdot)}$). We see that TacGraph uses subsequent contacts to correctly identify the correct local solution, rejecting initial solutions that may be consistent with only the local geometric

Object	Tactile-Only				
	ICP	CHSEL	SCOPE (v1)	SCOPE (v2)	TacGraph
`cylinder`	4 / 10	4 / 10	4 / 10	1 / 10	7 / 10
`rectangle`	2 / 10	6 / 10	0 / 10	1 / 10	6 / 10
`quarter cylinder`	0 / 10	1 / 10	0 / 10	0 / 10	1 / 10
`wrench`	5 / 10	3 / 10	3 / 10	1 / 10	9 / 10
Overall	11 / 40	14 / 40	7 / 40	3 / 40	23 / 40

TABLE IV: Tactile-Only Peg Insertion Results (Success / Attempt)

information.

C. Peg Insertion

We evaluate how our proposed methodology aids in a prehensile task requiring precise in-hand pose estimation. We perform a peg insertion task with several objects from Fig. 4. Each fixture for insertion is designed per the object geometry with 3 mm clearance. We perform the same inference procedure outlined in Sec. VI-B, fixing the grasps across run for fair comparison. We then take the final rest pose estimate $\boldsymbol{o}_r^*$ and perform an open-loop insertion. We use a F/T sensor mounted under the insertion fixture to stop and release if a force greater than 5 N is registered. If no force threshold is met, the object is released 5 mm above the floor of the insertion fixture. We run our experiment in the challenging tactile-only setting.

Results: The peg insertion success rates are reported across 10 presses per object for four of our objects for a total of 40 trials in Tab. IV. Our method is able to, from tactile alone, achieve precise open-loop peg insertion, outperforming the baselines across the objects. The `quarter cylinder` object we found had a noisy depth prediction from our model, as the gelsight is not as sensitive in the normal direction of the sensor, which caused a low success rate. We found ICP and CHSEL could align the overall orientation of the object well in some cases, and hence can succeed despite inaccurate poses, while SCOPE struggled to achieve accurate orientations leading to many failures.

VII. Discussion

In this work, we proposed TacGraph, a factor-graph based method that can utilize the physical constraints of contact, along with sensory feedback, to resolve object pose and contact simultaneously. While our results demonstrate the value of uniting physical constraints and sensory feedback, the work has several limitations.

TacGraph is still inherently a local method, which means we are sensitive to initialization. A potential extension to explore is enforcing diversity across particles [23]. In the experiments, our contact interactions here were manually selected - in future, we would like to utilize our estimator online during contact-rich tasks, as well as exploring action selection to explicitly drive down uncertainty in poses [32].

Our method has several limiting assumptions: that we know object and environment geometries, that no relative slip occurs, and that contact can be described by a single point. Exploring how we can utilize reconstructed object models could relax our geometry assumption [10, 8]. Utilizing our proposed TacGraph estimator in a control loop could allow corrective actions to avoid high force/torque interactions, thereby avoiding or limiting slip. Additionally, if one could detect a slip event [33], the estimator could be re-initialized. Finally, extending to handle multiple contact points or extended contact geometries would help extend this work to a more rich set of interactions.

References

[1] K.-T. Yu and A. Rodriguez, "Realtime state estimation with tactile and visual sensing for inserting a suction-held object," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 1628–1635.

[2] L. Li, G. Yang, L. Shao, and D. Hsu, "Stable object placement under geometric uncertainty via differentiable contact dynamics," *arXiv preprint arXiv:2409.17725*, 2024.

[3] R. Holladay, T. Lozano-Pérez, and A. Rodriguez, "Force-and-motion constrained planning for tool use," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 7409–7416.

[4] T. Schmidt, R. A. Newcombe, and D. Fox, "DART: Dense Articulated Real-Time Tracking." in *Robotics: Science and systems*, vol. 2, no. 1. Berkeley, CA, 2014, pp. 1–9.

[5] J. Tremblay, T. To, B. Sundaralingam, Y. Xiang, D. Fox, and S. Birchfield, "Deep Object Pose Estimation for Semantic Robotic Grasping of Household Objects," in *Proceedings of The 2nd Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, A. Billard, A. Dragan, J. Peters, and J. Morimoto, Eds., vol. 87. PMLR, 29–31 Oct 2018, pp. 306–316.

[6] A. Alspach, K. Hashimoto, N. Kuppuswamy, and R. Tedrake, "Soft-bubble: A highly compliant dense geometry tactile sensor for robot manipulation," in *2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)*, 2019, pp. 597–604.

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

[8] S. Suresh, H. Qi, T. Wu, T. Fan, L. Pineda, M. Lambeta, J. Malik, M. Kalakrishnan, R. Calandra, M. Kaess *et al.*, "Neuralfeels with neural fields: Visuotactile perception for in-hand manipulation," *Science Robotics*, vol. 9, no. 96, p. eadl0628, 2024.

[9] R. Li, R. Platt, W. Yuan, A. ten Pas, N. Roscup, M. A. Srinivasan, and E. Adelson, "Localization and manipulation of small parts using gelsight tactile sensing," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014, pp. 3988–3993.

[10] 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.

[11] M. Bauza, A. Bronars, Y. Hou, I. Taylor, N. Chavan-Dafle, and A. Rodriguez, "Simple, a visuotactile method learned in simulation to precisely pick, localize, regrasp, and place objects," *Science Robotics*, vol. 9, no. 91, p. eadi8808, 2024.

[12] H. T. Suh, N. Kuppuswamy, T. Pang, P. Mitiguy, A. Alspach, and R. Tedrake, "Seed: Series elastic end effectors in 6d for visuotactile tool use," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 4684–4691.

[13] S. Kim, D. K. Jha, D. Romeres, P. Patre, and A. Rodriguez, "Simultaneous Tactile Estimation and Control of Extrinsic Contact," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 12 563–12 569.

[14] 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)*, 2022, pp. 10 241–10 247.

[15] L. Kim, Y. Li, M. Posa, and D. Jayaraman, "Im2Contact: Vision-Based Contact Localization Without Touch or Force Sensing," in *Proceedings of The 7th Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, J. Tan, M. Toussaint, and K. Darvish, Eds., vol. 229. PMLR, 06–09 Nov 2023, pp. 1533–1546.

[16] M. J. V. der Merwe, Y. Wi, D. Berenson, and N. Fazeli, "Integrated Object Deformation and Contact Patch Estimation from Visuo-Tactile Feedback," in *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023.

[17] M. Van der Merwe, D. Berenson, and N. Fazeli, "Learning the Dynamics of Compliant Tool-Environment Interaction for Visuo-Tactile Contact Servoing," in *Proceedings of The 6th Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, K. Liu, D. Kulic, and J. Ichnowski, Eds., vol. 205. PMLR, 14–18 Dec 2023, pp. 2052–2061.

[18] C. Higuera, S. Dong, B. Boots, and M. Mukadam, "Neural contact fields: Tracking extrinsic contact with tactile sensing," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 12 576–12 582.

[19] K. Ota, D. K. Jha, K. M. Jatavallabhula, A. Kanezaki, and J. B. Tenenbaum, "Tactile estimation of extrinsic contact patch for stable placement," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 13 876–13 882.

[20] J. Lee and N. Fazeli, "Vitascope: Visuo-tactile implicit representation for in-hand pose and extrinsic contact estimation," in *Proceedings of Robotics: Science and Systems*, Los Angeles, USA, June 2025.

[21] S. Dikhale, K. Patel, D. Dhingra, I. Naramura, A. Hayashi, S. Iba, and N. Jamali, "VisuoTactile 6D Pose Estimation of an In-Hand Object Using Vision and Tactile Sensor Data," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2148–2155, 2022.

[22] P. K. Murali, B. Porr, and M. Kaboli, "Shared visuo-tactile interactive perception for robust object pose estimation," *The International Journal of Robotics Research*, vol. 44, no. 7, pp. 1186–1216, 2025.

[23] S. Zhong, D. Berenson, and N. Fazeli, "CHSEL: Producing Diverse Plausible Pose Estimates from Contact and Free Space Data," in *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023.

[24] A. Bronars, S. Kim, P. Patre, and A. Rodriguez, "Texterity: Tactile extrinsic dexterity," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 7976–7983.

[25] A. Sipos and N. Fazeli, "Simultaneous contact location and object pose estimation using proprioception and tactile feedback," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 3233–3240.

[26] ——, "MultiSCOPE: Disambiguating In-Hand Object Poses with Proprioception and Tactile Feedback," in *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023.

[27] L. Manuelli and R. Tedrake, "Localizing external contact using proprioceptive sensors: The contact particle filter," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 5062–5069.

[28] F. Dellaert and M. Kaess, "Factor graphs for robot perception," *Foundations and Trends® in Robotics*, vol. 6, no. 1-2, pp. 1–139, 2017.

[29] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. J. Leonard, and F. Dellaert, "iSAM2: Incremental smoothing and mapping using the Bayes tree," *The International Journal of Robotics Research*, vol. 31, no. 2, pp. 216–235, 2012.

[30] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo *et al.*, "Segment anything," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 4015–4026.

[31] F. Dellaert and G. Contributors, "borglab/gtsam," May 2022. [Online]. Available: https://github.com/borglab/gtsam)

[32] S. Zhong, N. Fazeli, and D. Berenson, "Rumi: Rummaging using mutual information," *IEEE Transactions on Robotics*, vol. 41, pp. 5431–5450, 2025.

[33] F. Veiga, H. van Hoof, J. Peters, and T. Hermans, "Stabilizing novel objects by learning to predict tactile slip," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 5065–5072.