2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

TacTape: Real-time High-accuracy Tactile Fiducial System with Structured 3D Texture for Vision-based Tactile Sensors

Meng Wang[1*], Wanlin Li[1*], Qiuxuan Chen[1,2], Yuzhe Huang[1,3], Hang Li[1], Kaspar Althoefer[4], Ziyuan Jiao[1], Yao Su[1†], and Hangxin Liu[1†]

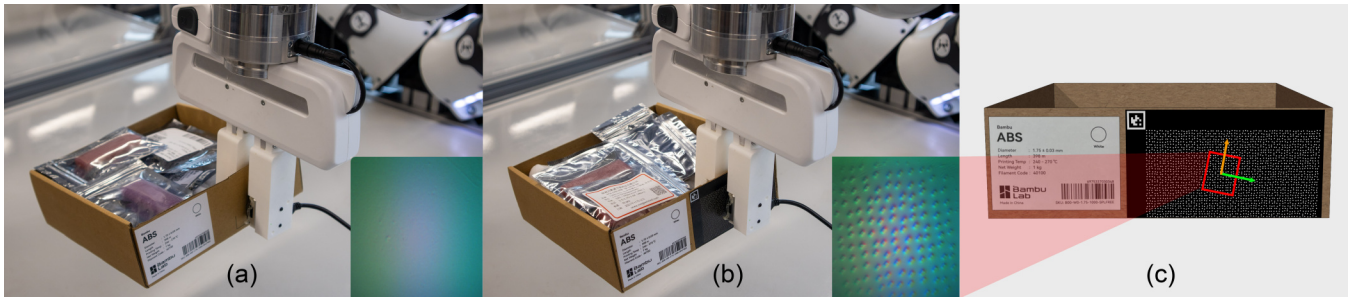


Fig. 1: **Proposed tactile fiducial system.** (a) VBTS cannot perceive information from smooth surfaces; (b) TacTape attaches structured patterns to interactive surfaces; (c) Partially observed patterns enable global localization on object.

***Abstract*— Vision-based tactile sensors enable high-resolution tactile perception by capturing image-based contact data. However, their utility in tactile localization is limited by their inherently small and local sensing area, as well as their dependence on distinct object surface features. We propose TacTape, a novel tactile fiducial system that enables accurate and efficient tactile localization by attaching textured tape to object surfaces. A lightweight algorithm allows real-time estimation of contact position and orientation from partially observed structured 3D textures. Experiments demonstrate that TacTape achieves sub-millimeter positional and sub-degree angular localization accuracy, and operates significantly faster than classic tactile mapping methods.**

I. Introduction

Unlike capacitive and resistive tactile sensors relying on discrete pressure points, **vision-based tactile sensors (VBTSs)** [1–8] capture images of the contact area, offering pixel-level spatial resolution to glean detailed tactile information about the contact surface. Leveraging these features, VBTSs have been integrated into various robotic gripper configurations and have proven effective in a range of robotic manipulation tasks, including shear/slip force regulation [9–11], texture classification [12, 13], object manipulation [14–16], and the handling of fragile and deformable objects [8, 17–19]. These advances are largely supported by the ability to localize contact using high-resolution tactile information.

Traditional tactile sensing methods [8, 16, 20, 21] rely on distinct texture features for accurate localization and identification, which makes them ineffective on smooth surfaces or surfaces with repetitive patterns, both of which are common in everyday objects. While many recent methods [22–26] aim to reduce the amount of prior information required,a **tactile fiducial** system provides an alternative contact-based localization strategy. Analogous in function to visual fiducial systems such as AprilTag [27], tactile fiducials encode positional information through structured, coded 3D textures, enabling contact-based localization when mounted on object surfaces.

However, a reliable **tactile fiducial system is absent** for current VBTSs, due to several fundamental challenges. Unlike visual systems that capture full tags from a global view, tactile sensors only perceive local contact, making it difficult to obtain full-tag observation. Thus, tactile tags must support reliable decoding from partial patterns across the entire interactive surface. Moreover, their structured 3D textures require a parameterized and fully automated design-to-fabrication pipeline.

In this paper, we propose TacTape, a tactile fiducial system that enables accurate and real-time tactile localization on objects for VBTSs. TacTape features a flexible tape that can be easily customized to apply structured textures to objects. It is fabricated using standard SMT-FPC (Surface Mount Technology Flexible Printed Circuit) through a parameterized and automated process, ensuring high reproducibility and consistency while supporting scalable manufacturing. By integrating a symbolic encoding and decoding scheme [28], a geometric method for efficient pattern recognition, and pixel resolution transformations [29], the system achieves sub-millimeter positional and sub-degree angular localization accuracy. The fabrication and localization pipeline is illustrated in Fig. 2. TacTape's real-time, high-accuracy performance and ease of deployment make it well-suited for laboratory and potential industrial setups and provide a reliable metric for evaluating and adjusting gripping actions with VBTSs.

* Meng Wang and Wanlin Li contributed equally to this work.
† Corresponding authors.
[1] State Key Laboratory of General Artificial Intelligence, Beijing Institute for General Artificial Intelligence (BIGAI). Emails: {wangmeng, liwanlin, chenqiuxuan, huangyuzhe, lihang, jiaoziyuan, suyao, liuhx}@bigai.ai.
[2] School of Artificial Intelligence, Beijing University of Posts and Telecommunications.
[3] Department of Automation, Beihang University.
[4] Centre for Advanced Robotics @ Queen Mary (ARQ), Queen Mary University of London. Email: k.althoefer@qmul.ac.uk.

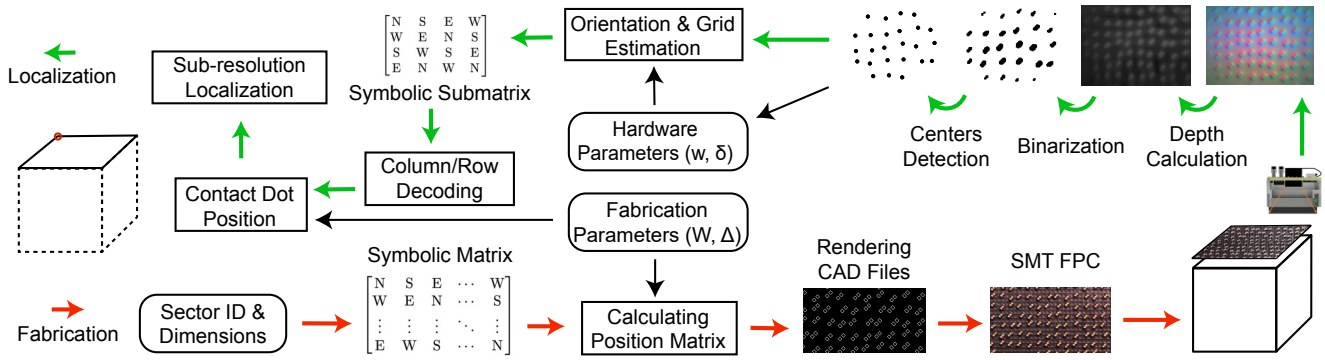


Fig. 2: **Fabrication and localization pipeline of TacTape**. Given a region-specific ID, dimensions, and sensor-specific fabrication parameters, the texture engine automatically generates an encoded symbolic matrix and renders the corresponding CAD files for SMT FPC. Mounted solid components form a structured texture that can be captured by VBTSs and transformed into structured 2D patterns through depth binarization. By estimating the orientation and restoring the grid, a symbolic submatrix can be extracted from the partially observed pattern. After decoding the column and row indices of the submatrix in the original symbolic matrix, the sensor's position within the specified area can be accurately estimated using a subsequent sub-resolution transform.

II. Encoding Principle & Fabrication

A. Encoding Principle

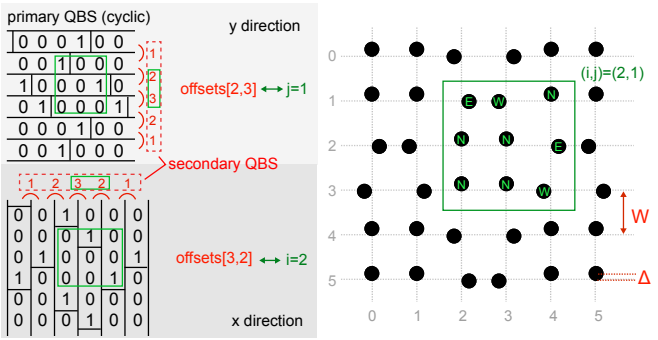


Fig. 3: **A simplified example of encoding with a cut-down QBS** $[0, 0, 0, 1]$ **of order 3.** The red numbers are offsets between a sequence and its previous column/row, encoding global column/row indices via a unique subsequence of 2 offsets in the secondary QBS. Each point encodes 2-bit information for both the x and y directions, represented by the direction of displacement. The grid space W and displacement Δ can be adjusted to generate appropriate coordinates for dot centers.

VBTSs convert structured 3D textures into distinctive 2D depth patterns, in which global positional information can be encoded and recognized. Similar to the commonly used 2D position encoding method *Anoto dots*, we utilize quasi De Bruijn sequences (QBS) with a binary alphabet $\{0, 1\}$ as the encoding scheme. As shown in Fig. 3, columns (encoding x positions) or rows (encoding y positions) contain the same cyclic sequences of different offsets. Owing to the unique properties of QBS, a sequence of **order** n enables the identification of the offset by observing n consecutive elements. Offsets between neighbor columns/rows are designed to be a secondary QBS of **order** $n - 1$, allowing a subsequence of $n - 1$ offsets to encode the absolute indices. Each grid point is required to encode 2-bit data separately for the x and y directions, represented by the displacement direction $\{\mathrm{N}, \mathrm{S}, \mathrm{W}, \mathrm{E}\}$ of a dot on the grid. Utilizing a primary QBS of length 16 and order 4, the decode requires a minimum of 4×4 dots observed, and the position is decoded into its column/row indices (i, j) at a maximum range of $(16^3, 16^3)$. In addition, the start offset of the primary QBS is adjustable and can also be decoded from observed dots, thus providing 256 sectors for identification and localization across multiple surfaces. A detailed explanation of the symbolic encoding and decoding process is given in [28], including a redundant encoding mechanism for error detection.

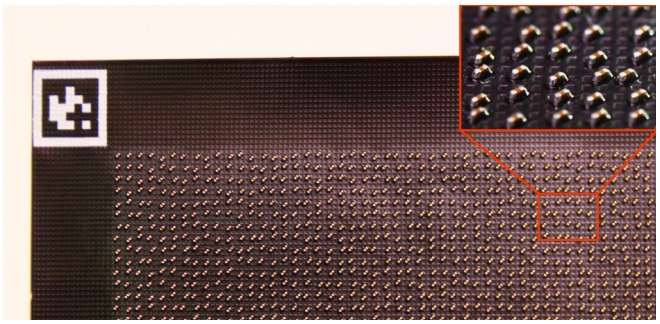


Fig. 4: **A visual-tactile fiducial tape with FPC-SMT technology**, facilitating the modification of surface texture in the environment.

B. Texture Design & Fabrication

TacTape features a flexible base with a solid texture and is fabricated using FPC-SMT technology (Fig. 4). Instead of utilizing dots, 0201 resistors are mounted to create the structured 3D texture. Given the physical dimension of the **grid space** W, a symbolic matrix can be generated via *pymicrodots* [28] to cover the entire tape. Given the **physical displacement** Δ, layered CAD files for PCB fabrication can be automatically rendered as the symbolic matrix for the placement of all components. Given the footprint, the grid space is selected as $W = 1.5mm$, the displacement is selected as $\Delta = 0.25mm$, and the components are rotated by 45° to avoid pattern overlap. In addition, an AprilTag is printed for visually alignment. The entire fabrication process is highly parameterized and requires no handcrafting.

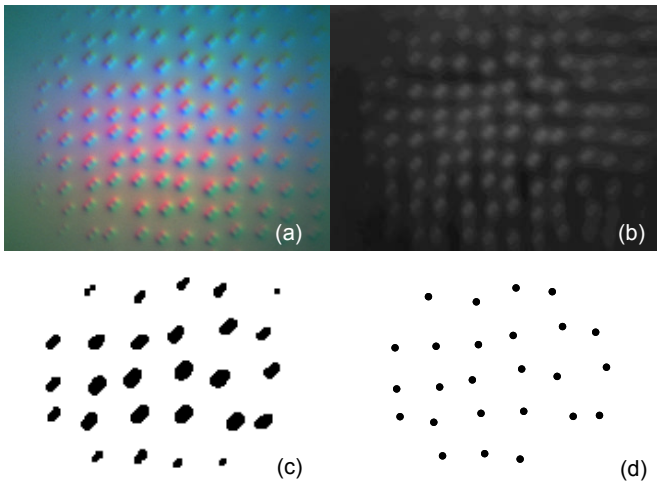


Fig. 5: **Preprocess of the image data captured by a VBTS**. (a) Sensor raw input. (b) Depth image calculated via a neural network. (c) Binary depth image of patterns. (d) Centers are calculated via geometric moment from contour detection.

III. Localization Pipeline & Sensor-level Wrap

A. Localization Pipeline

*1) **Preprocess**:* Fig. 5a shows a raw image captured by the VBTS. Unlike localization using 2D visual patterns, tactile patterns must be scaled up significantly to be clearly captured through elastomer deformation. This results in a sparse distribution, which makes traditional methods such as the constrained k-means algorithm [29] inapplicable due to the insufficient number of recognizable patterns.

The corresponding depth map, as shown in Fig. 5b is computed using a popular neural network (NN)-based method [30]. We need to remove the background depth to get a uniform pattern depth by constructing a high-pass filter:

$$\text{depth}_{\text{pattern}} = [\text{depth} - meanFilter(\text{depth})] \times K_{\text{contrast}}. \quad (1)$$

The depth of patterns is then binarized as contours shown in Fig. 5c. The black dots in Fig. 5d, representing the centers of these patterns, are calculated using geometric moments from contour detection. To estimate the center of contact $(\hat{c}_x, \hat{c}_y)$ and identify the nearest dot (c_x, c_y) , we apply the following formula:

$$(\hat{c}_x, \hat{c}_y) = \left(\frac{1}{N}\sum_{i=1}^{N} x_i,\ \frac{1}{N}\sum_{i=1}^{N} y_i \right), \quad (2)$$

$$(c_x, c_y) = \underset{(x_i, y_i)}{\arg\min} \left\{ (x_i - \hat{c}_x)^2 + (y_i - \hat{c}_y)^2 \right\}, \quad (3)$$

where (x_i, y_i) is the position of each dot, and N is the total number of dots. As the dots closest to the center are generally of the highest quality, we designate (c_x, c_y) as the **contact dot** and use it to determine decoding matrix in Sec. III-A.4 and calculate sub-resolution position in Sec. III-A.5 .

*2) **Orientation Estimation**:* First, we propose a geometric method which requires fewer dots while providing improved efficiency to estimate the orientation of the input image. This method is based on the assumption that the **grid width** w and **dot displacement** δ **in pixels** can be accurately estimated, and are constant values within the same hardware. We provide an automated estimation method in Sec. III-B.

As illustrated in Fig. 6a, a *baseline* is defined as a line that passes through a pair of dots. If a baseline aligns with the grid, all dots must adhere to an implicit spacing constraint imposed by the encoding process:

$$d_{\text{i,baseline}} = mw + n\delta, \quad m \in \{0, 1, 2, \ldots\}, n \in \{0, 1, 2\} \quad (4)$$

where $d_{\text{i, baseline}}$ represents the distance between the dot and the baseline. As such, we can efficiently search for a baseline with slope θ_{slope} satisfying this constraint, allowing for an additional angular offset $\theta_A \in \{0°, 90°, 180°, 270°\}$ from the correct orientation. θ_A can be identified later during decoding, and the actual orientation of the sensor in texture coordinates would be:

$$\theta = \theta_{\text{slope}} + \theta_A. \quad (5)$$

Since each dot is typically contained in at least one valid pair, the search complexity can be reduced from $O(N^2)$ to $O(N)$ by anchoring the search on a single dot.

*3) **Grid Estimation**:* Once the orientation is estimated, we obtain a set of orthogonal dots (x'_i, y'_i) (see Fig. 6b) by rotating the coordinates. We then need to find a valid grid point (x_g, y_g). For example, y_g can be determined by identifying equidistant distributions within y'_i (**Case A**):

$$y_g = y'_{(n)} \quad \text{for} \quad y'_{(n)} - y'_{(n-1)} = \delta, y'_{(n+1)} - y'_{(n)} = \delta \quad (6)$$

where sequence $y'_{(i)}$ is derived by sorting y'_i in ascending order and removing duplicate values. x_g can be determined similarly, or deduced by y_g and a nearby dot (**Case BC**):

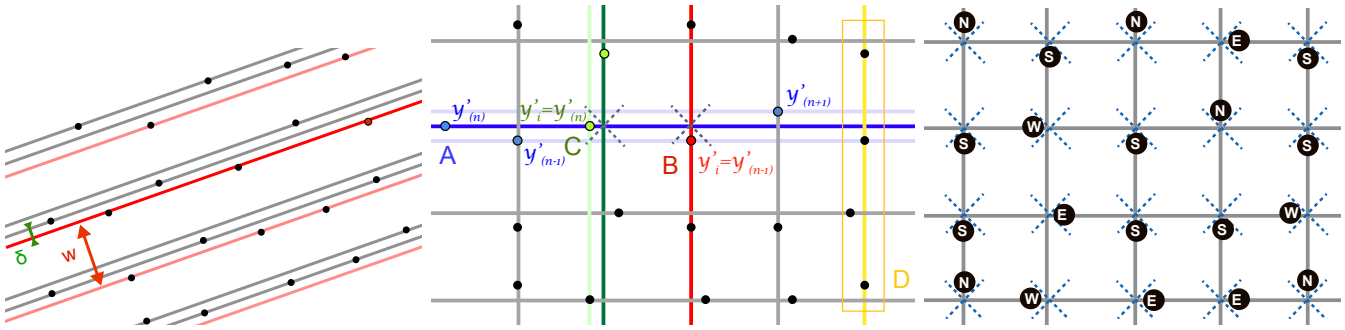
$$x_g = \begin{cases} x'_i, & y'_i = y'_{(n\pm1)} \\ \arg\min_{x \in x'_{(i)} \setminus \{x'_i\}} |x - x'_i|, & y'_i = y'_{(n)} \end{cases}. \quad (7)$$

In very rare cases where no equidistant distribution is found in either direction, valid grid points can, as a last resort, be selected based on probability (**Case D**), since a true grid line typically passes through more points:

$$P(x_g = x'_{(n)}) = \frac{M_n}{M_n + M_{n\pm1}} \quad \text{for} \quad |x'_{(n)} - x'_{(n\pm1)}| = \delta \quad (8)$$

where M_n is the count of $x'_{(n)}$ in the original series of x'_i. With any valid (x_g, y_g) identified, we can determine all the grid lines using the grid width w as offsets.

*4) **Decoding**:* After grids calculated, the tactile localization is transformed into a previously solved 2D localization problem [29]. By matching each dot with the nearest grid point and inspecting its placement in the 45° quadrant division (Fig. 6c), we can assign corresponding symbols for each of them. Although only a 4×4 symbolic matrix is needed, we extract a 5×5 matrix of symbols centered on the contact dot. This allows for four decoding attempts using a sliding 4×4 window, improving robustness against missing or incorrect symbols at the edges. The decoding is performed using *py-microdots* [28], yielding the aligned column/row indices (i, j) of the contact dot, along with the sector identifier S_{ID}.



(a) **Orientation estimation**. (b) **Grid estimation**. (c) **Symbol assignment**.

Fig. 6: **Geometric method to parse symbolic matrix from dots**. (a) Use the spacing constraint to identify an aligned line within the grid; (b) Inspect the distribution of dot coordinates and determine grid points (crossed); (c) Assign symbols using a 45° quadrant division.

As noted in Sec. III-A.2, the extracted matrix may be misoriented. Leveraging the error-detection property of the encoding, the correct orientation θ_A is determined by rotating the 4×4 matrix in all four directions and selecting the one that yields a valid decoding. Then, the contact dot's physical position in the texture coordinate frame, $\vec{C}$, is computed as:

$$\vec{C} = (iW + s_x\Delta,\ jW + s_y\Delta), \tag{9}$$

where s_x, s_y are **displacement factors** to calculate the offset from a nearest grid point to the contact dot, given the decoded symbol $\{\mathrm{N}, \mathrm{S}, \mathrm{W}, \mathrm{E}\}$:

$$(s_x, s_y) \in \{\mathrm{N}:(0,-1),\ \mathrm{S}:(0,1),\ \mathrm{W}:(-1,0),\ \mathrm{E}:(1,0)\} \tag{10}$$

*5) **Sensor Position Calculation**:* While the contact dot positions are discrete values, sensor positions in texture coordinates can be accurately calculated using a sub-resolution method (Fig. 7). By combining the position of the contact dots (c_x, c_y) in the sensor image introduced in Sec. III-A.1 with the correct orientation θ of sensor direction (calculated in Secs. III-A.2 and III-A.4), we can then calculate the translation from the contact dot to the sensor center $\vec{t}$ as:

$$\vec{t} = (\frac{1}{2}\text{width} - c_x, \frac{1}{2}\text{height} - c_y). \tag{11}$$

Utilizing a resolution factor $R = w/W$ and a rotation matrix, the accurate position of the sensor center in the texture coordinates can be calculated as:

$$\vec{S} = \vec{C} + \frac{1}{R} \cdot \vec{t} \cdot \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}. \tag{12}$$

B. Sensor-level Wrap

*1) **Sensor Initialization**:* The measurement must be initialized by parameters. W and Δ are fabrication parameters, while w and δ are determined by the sensor resolution:

$$w = W \times R, \quad \delta = \Delta \times R. \tag{13}$$

If R is unknown or subject to change, we provide a method to directly estimate w and δ from the input image:

We denote $d_{\text{i,nearest}}$ as the measured distance from the i-th dot to its nearest neighbor in the image. Theoretically, assuming an infinite grid, there are only three possible

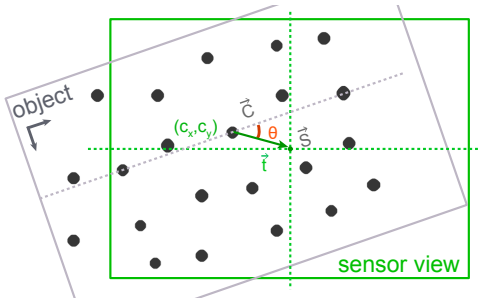
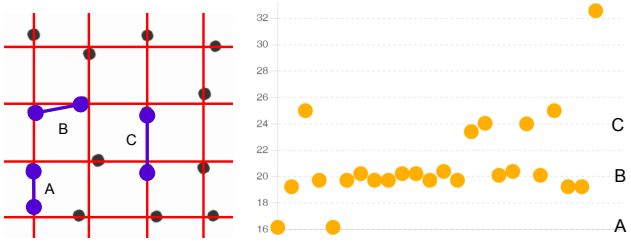


Fig. 7: **Sensor position calculation**. While dot positions are discrete values, the accurate position of the sensor can be calculated by using the orientation θ, the translation vector $\vec{t}$, and a scale factor from pixel to physical dimension.



(a) **Possible distances**. (b) **Real distribution**.

Fig. 8: **Estimation of grid width w in pixel**. Assuming an infinite grid, only three possible distances between a dot and its nearest neighbor exist. In a real input image, there is a significant aggregation on d_B, such that we can estimate d_B and calculate w.

neighbor distances (Fig. 8a):

$$d_A = w - 2\delta, \quad d_B = \sqrt{\delta^2 + (w-\delta)^2}, \quad d_C = w. \tag{14}$$

In real input images, most observed values of $d_{i,\text{nearest}}$ cluster around d_B (Fig. 8b). To estimate d_B we first compute the mean d_{mean} and standard deviation σ of all $d_{i,\text{nearest}}$ values, then refine the estimate using values within a $\pm 2\sigma$ range:

$$\hat{d}_B = \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} d_{i,\text{nearest}}, \quad \mathcal{I} = \{i \mid d_{i,\text{nearest}} \in d_{\text{mean}} \pm 2\sigma\}. \tag{15}$$

With a reference design of $\Delta = W/6$, the grid width w can then be estimated as:

$$w = 6\hat{d}_B/\sqrt{26}. \tag{16}$$

*2) **Sensor Output**:* Tactile fiducial systems require richer information output than visual fiducial systems to enable more versatile interactions between objects and robots, as described below:

- **Sector ID** S_{ID} (Sec. III-A.4) of contact surface;
- **Sensor position** $\vec{S}$ (Sec. III-A.5) and **orientation** θ (Sec. III-A.2) in the texture coordinate system;
- **Contact center position** $\vec{C}$ (Sec. III-A.4);
- **Contact status** with the number of dots N_{dot} to ensure robust decoding: *no contact* ($N_{dot} = 0$), *weak contact* ($N_{dot} < 16$), or *solid contact* ($N_{dot} \geqslant 16$).

IV. Experiments and Applications

A. Performance Experiment Setup

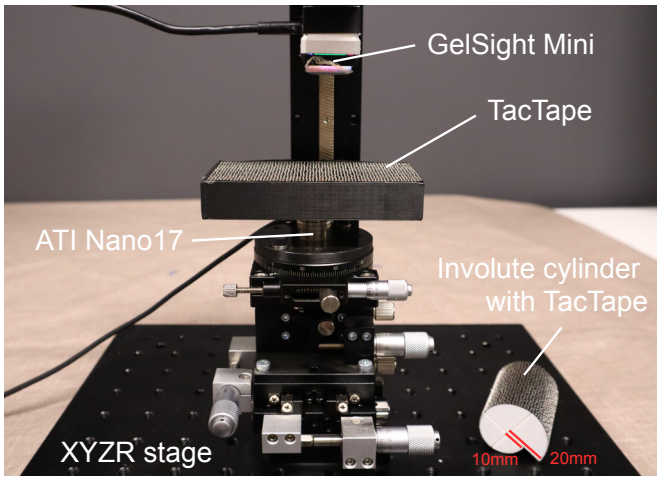


Fig. 9: **Performance Experiment setup**. A GelSight Mini [31] and a rigid block attached with the TacTape tap are set on a 4-axis platform with a force sensor. Another involute cylindrical body is used for replacement to measure the minimum curvature radius.

While the TacTape is compatible with any type of VBTSs, we use a commercially available GelSight Mini [31] for better comparison. The sensor captures raw images at a resolution of 320×240 pixels and has a sensing region of 20×15 mm. All experiments, except for curvature testing, are conducted using a TacTape measuring 100×50 mm and affixed to a rigid block. A 4-axis XYZR linear stage trimming platform is used to evaluate the sensor's performance in these experiments, as shown in Fig. 9.

Required contact force: We used a force sensor to test the contact force threshold for the TacTape. As shown in Fig. 9, a multi-axis force/torque sensor ATI Nano17 is attached beneath the test object on the trimming platform. The GelSight Mini sensor is placed on the Z stage and trimmed down to establish contact with the TacTape until effective decoding occurs.

Minimum curvature radius: To evaluate the possible curvature range, we designed and 3D-printed an involute cylindrical body, on which we affixed the TacTape to examine the threshold of curvature. The curvature radius of the involute cylindrical body in this experiment ranged from 10 mm to 20 mm uniformly allocated over 360°. The tactile sensor is vertically pressed onto different areas of the cylindrical body.

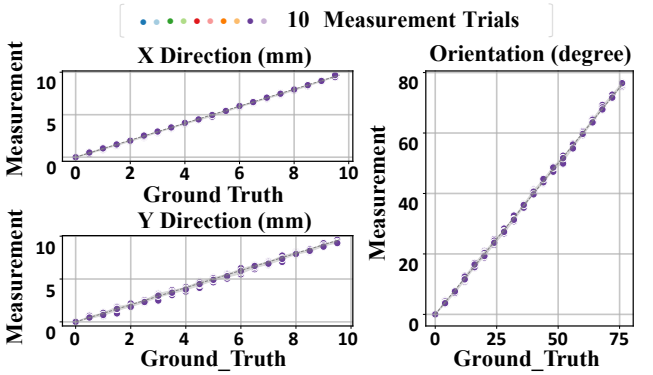


Fig. 10: **Linearity and repeatability of the TacTape**. Ten trials of accuracy and repeatability test were conducted for translations in x and y directions (from 0 to 10 mm with a step of 0.5 mm) and rotation r in z-direction (from 0 to 80° with a step of 4°).

Orientation accuracy: In the this experiment, the test object is mounted on the R stage, allowing for rotation in the Z-axis, while the tactile sensor is mounted on the Z stage, as shown in Fig. 9. We trimmed the R stage in increments of 4° and pressed the sensor to record the decoding orientation angle θ. This data collection process was repeated 10 times.

Position accuracy: In this experiment, the test object is mounted on the XY stage, allowing for translation across both axes, while the tactile sensor is mounted on the Z stage. We trimmed the X and Y stages individually in increments of 0.5 mm and pressed the sensor to record the decoding sensor position. The process was repeated 10 times.

B. Performance Results

A minimum force of 2 N and a minimum curvature radius of 15.7 mm are required to capture sufficient patterns for successful decoding. Given sufficient input, the algorithm achieves a decoding success rate of over 99%. Fig. 10 shows the measurement trials, and Tab. I presents the results of a linear regression analysis. Installation errors between the sensor and the testbed, which are indicated by the regression intercept and slope, are included in the root mean square error (RMSE); therefore, the residual standard error (RSE) from the regression provides a more appropriate representation of sensor accuracy. The localization algorithm requires less than 3 ms per frame, excluding image capture and depth reconstruction time of VBTS (30 ms in our implementation).

Tab. II shows an comparison with classic tactile mapping method using BRISK+RANSAC [20], of which the reported accuracy is still better than other existing methods with prior and featured texture information. Our method, by utilizing structured patterns and geometrical localization pipeline, provides much higher accuracy and efficiency. To be noted, the efficiency of tactile mapping is heavily dependent on the size of the localization area and the image resolution, whereas our method is not. The efficiency in Tab. II is under the same structured patterns input and a localization area five times larger than the sensor size.

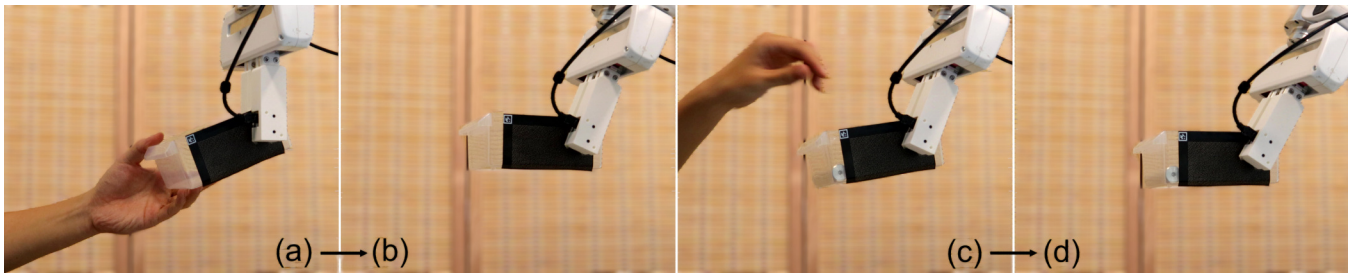


Fig. 11: **Dynamic posture maintenance in human–robot interaction.** (a)(b) The robot keeps the toolbox horizontal after a handover, regardless of the initial grasping state. (c)(d) The robot maintains the toolbox horizontally after the human drops in a heavy part, by detecting and responding to the displacement caused by the slippage.

TABLE I: **Liner regression analysis summary.**

	R^2	Slope	Intercept	RSE	RMSE
X (mm)	0.999	0.997	0.052	**0.063**	0.092
Y (mm)	0.997	0.988	0.004	**0.170**	0.184
O (deg)	0.999	1.005	-0.260	**0.598**	0.610

RMSE are computed with respect to the ground-truth values.

TABLE II: **Comparison with classic tactile mapping.**

	BRISK+RANSAC [20]	Ours
Position accuracy	0.14 mm	0.12 mm
Orientation accuracy	1.15°	0.60°
Computation Time	$\sim$ 1000 ms	3 ms
Actual Frame Rate	$<$ 1 Hz	30 Hz

C. Applications Demonstration

Structured patterns enable high-accuracy, real-time tactile localization, facilitating highly dynamic scenarios such as human-robot interaction. Fig. 11a illustrates a collaboration scenario in which a human hands a toolbox to the robot. Following the handover, the robot can immediately compensate for the initial grasping offset and align the toolbox horizontally, preparing for the next task. In Fig. 11c, a human drops a heavy component into the toolbox, causing slippage and inclination. Detecting the change of contact patterns, the robot promptly reconfigures its posture to maintain horizontal alignment of the toolbox.

The proposed TacTape system also significantly enhances the localization flexibility and applicability of VBTSs, by enabling direct modification of interactive surfaces of everyday objects. This is particularly beneficial for objects with smooth surfaces or repetitive patterns, which traditional methods struggle to handle due to the lack of distinct textural features. Fig. 12a shows a robot inserting a key. Traditional tactile mapping method [20] works most of the time but occasionally fails when gripping on partially repetitive patterns. By overlaying the surface with a piece of TacTape, the robot achieves a higher success rate on this task. Fig. 12b shows the robot extending its workspace using a smooth stick. Without distinct textural features, the robot can only apply a principal component analysis (PCA) [32] method to estimate the stick's orientation, whereas with TacTape it can accurately estimate the position of the tip.

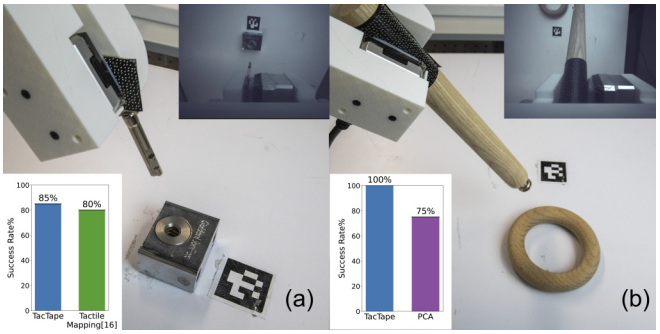


Fig. 12: **Accurate tactile localization on everyday objects**. (a) Grasping and inserting a key with partially repetitive patterns. We achieved a success rate of 17 out of 20 trials, better than the tactile mapping baseline [20]. (b) A robot extends its workspace by using a smooth stick and accurately estimates the position of the tip. We achieved a success rate of 20 out of 20 trials in retrieving the object, outperforming the baseline using PCA [32] without adequate position information.

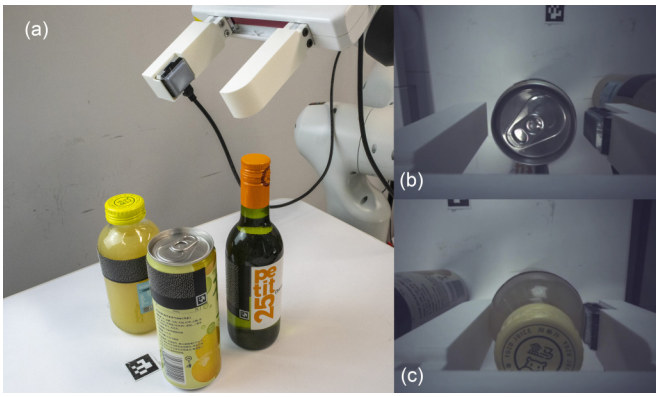


Fig. 13: **Manipulation with limited vision information**. (a) The robot is tasked with aligning the direction of labels on all the bottles. (b)(c) The robot efficiently adjusts the orientation of the bottles by rotating its wrist joint, even if the labels on the bottle are not visible from current perspective. The grasping pose is estimated through a simple combination of TacTape and diameter measurement.

In addition, localization with TacTape is entirely tactile and does not rely on external vision sensors. This facilitates manipulation in scenarios where visual information is limited due to perspective, occlusion, or lighting conditions. As shown in Fig. 13, the orientation of bottles can be adjusted through tactile sensing even when the camera cannot capture the labels from a top view, making task and motion planning

easier and more efficient. This also supports more human-like manipulation policies—for example, vision can be used to plan a rough grasping action, while tactile sensing provides high-accuracy object localization. After grasping, the vision sensor can then be allocated to other tasks, with tactile sensing alone maintaining object tracking.

V. Discussion

Performance improvement. From the algorithm applied in Sec. III-A.2 and according to standard error propagation, the angular uncertainty σ_θ scales with the centroid positional error σ_{dot} (in pixels, decided by the quality of depth reconstruction) and inversely with the pixel distance d between the selected pair of dots:

$$\sigma_\theta \approx \frac{2\,\sigma_{dot}}{d}, \quad d = D \times R. \tag{17}$$

Thus, the accuracy of orientation can be improved by selecting a pair of dots with a larger separation D, or by increasing the resolution R. Similarly, as described in Sec. III-A.5, when σ_θ is very small, the positional uncertainty $\sigma_{\vec{S}}$ is determined by:

$$\sigma_{\vec{S}} \approx \frac{\sigma_{dot}}{R}. \tag{18}$$

Consequently, a higher resolution can significantly improve the accuracy of both orientation and position. Conversely, if the accuracy exceeds the requirement, the resolution can be reduced to increase the frequency, since the depth reconstruction time decreases rapidly. Denser grid spacing W, while having little effect on accuracy, requires less contact force and a smaller area, leading to more robust decoding.

Deployment limitation. TacTape is suitable for experimental environments where surface modification is allowed. To increase flexibility and reproducibility we utilize standard FPC-SMT as a manufacturing technique. However, for geometrically non-developable surfaces such as spheres, tapes may not adhere perfectly. For such complex geometries we suggest integrating TacTape patterns during the design and fabrication process, using 3D printing or milling with CNC. Nevertheless, planar and cylindrical surfaces account for the majority of interactive areas on everyday objects, making TacTape highly applicable in practical scenarios. After repeated sliding, a small number of components may detach from the surface. Nevertheless, owing to the redundant encoding design and sub-resolution method, this leads only to a marginal decrease in decoding success rate and does not degrade localization accuracy.

Cross-modal applications. Tactile sensing is rarely used as a standalone modality in robotic interaction. By integrating an additional visual fiducial marker, TacTape establishes a shared reference frame for both vision and touch, enabling accurate, continuous, and cross-modal interaction in real-world robotic systems. Meanwhile, in tactile localization tasks that rely on vision sensors or proprioception, obtaining reliable ground truth is critical. TacTape provides an effective metric for evaluating grasp status, performing consistency calibration, and guiding sim-to-real transfer. Furthermore, when combined with force estimation from depth using conventional methods, the system enables closed-loop control of both force and position during manipulation.

Future work. Current TacTape stress reproducibility relies on standard fabrication techniques, which limits the versatility of the patterns. As a next step, we plan to explore computational algorithms for tactile pattern generation, enabling both the design and fabrication of patterns that are friendly to humans and robots. In addition, localizing the 3D postures of objects currently requires prior knowledge of their surface shapes. We aim to address this challenge by combining TacTape with visual SLAM, achieving high-accuracy tactile 3D posture estimation without any prior assumptions.

VI. Conclusion

This paper introduces TacTape, a novel tactile fiducial system featuring a flexible, textured tape that enables easy modification of object surfaces to enhance interaction with VBTSs. By employing encoded and structured 3D textures, contact position and orientation can be directly calculated even for objects with smooth surfaces. Combined with the proposed algorithm, TacTape outperforms existing tactile localization methods in both accuracy and efficiency.

TacTape is fabricated using standard FPC-SMT technology through a parameterized and automated pipeline, ensuring high reproducibility and accessibility across a wide range of scenarios. By decoupling tactile localization from surface textural priors, TacTape enables plug-and-play tactile interaction across diverse tasks, tools, and surfaces. This opens up new possibilities for general-purpose tactile augmentation in unstructured environments such as laboratory and household.

Acknowledgement: This work was supported in part by the National Natural Science Foundation of China (No. 62403064) and Shenzhen Science and Technology Program (No. ZDCY20250901094531003).

References

[1] B. Ward-Cherrier, N. Pestell, L. Cramphorn, B. Winstone, M. E. Giannaccini, J. Rossiter, and N. F. Lepora, "The tactip family: Soft optical tactile sensors with 3d-printed biomimetic morphologies," *Soft robotics*, vol. 5, no. 2, pp. 216–227, 2018.

[2] 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.

[3] C. Lin, Z. Lin, S. Wang, and H. Xu, "Dtact: A vision-based tactile sensor that measures high-resolution 3d geometry directly from darkness," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2023.

[4] C. Zhang, S. Cui, S. Wang, J. Hu, Y. Cai, R. Wang, and Y. Wang, "Gelstereo 2.0: An improved gelstereo sensor with multimedium refractive stereo calibration," *IEEE Transactions on Industrial Electronics*, vol. 71, no. 7, pp. 7452–7462, 2023.

[5] M. Wang, W. Li, H. Liang, B. Li, K. Althoefer, Y. Su, and H. Liu, "Large-scale deployment of vision-based tactile sensors on multi-fingered grippers," in *IEEE/RAS International Conference on Intelligent Robots and Systems (IROS)*, 2024.

[6] W. Li, Z. Zhao, L. Cui, W. Zhang, H. Liu, L.-A. Li, and Y. Zhu, "Minitac: An ultra-compact 8 mm vision-based tactile sensor for enhanced palpation in robot-assisted minimally invasive surgery," *IEEE Robotics and Automation Letters*, vol. 9, no. 12, pp. 11170–11177, 2024.

[7] W. Li, P. Lin, M. Wang, C. Xiao, K. Althoefer, Y. Su, Z. Jiao, and H. Liu, "R-tac0: A rounded high-frequency transferable monochrome vision-based tactile sensor for shape reconstruction," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 10400–10407, IEEE, 2025.

[8] P. Lin, Y. Huang, W. Li, J. Ma, C. Xiao, and Z. Jiao, "Pp-tac: Paper picking using tactile feedback in dexterous robotic hands," in *Robotics Science and Systems (RSS)*, 2025.

[9] W. Yuan, R. Li, M. A. Srinivasan, and E. H. Adelson, "Measurement of shear and slip with a gelsight tactile sensor," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015.

[10] I. H. Taylor, S. Dong, and A. Rodriguez, "Gelslim 3.0: High-resolution measurement of shape, force and slip in a compact tactile-sensing finger," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2022.

[11] Z. Zhao, Y. Li, W. Li, Z. Qi, L. Ruan, Y. Zhu, and K. Althoefer, "Tac-man: Tactile-informed prior-free manipulation of articulated objects," *IEEE Transactions on Robotics*, vol. 41, pp. 538–557, 2024.

[12] W. Yuan, Y. Mo, S. Wang, and E. H. Adelson, "Active clothing material perception using tactile sensing and deep learning," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2018.

[13] R. Li and E. H. Adelson, "Sensing and recognizing surface textures using a gelsight sensor," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[14] Z.-H. Yin, B. Huang, Y. Qin, Q. Chen, and X. Wang, "Rotating without seeing: Towards in-hand dexterity through touch," in *Proceedings of Robotics: Science and Systems (RSS)*, 2023.

[15] Y. Fuchioka and M. Hamaya, "An electromagnetism-inspired method for estimating in-grasp torque from visuotactile sensors," *arXiv preprint arXiv:2404.15626*, 2024.

[16] Z. Zhao, W. Li, Y. Li, T. Liu, B. Li, M. Wang, K. Du, H. Liu, Y. Zhu, Q. Wang, *et al.*, "Embedding high-resolution touch across robotic hands enables adaptive human-like grasping," *Nature Machine Intelligence*, pp. 1–12, 2025.

[17] W. Li, M. Wang, J. Li, Y. Su, D. K. Jha, X. Qian, K. Althoefer, and H. Liu, "L∈3} f-touch: A wireless gelsight with decoupled tactile and three-axis force sensing," *IEEE Robotics and Automation Letters*, vol. 8, no. 8, pp. 5148–5155, 2023.

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

[19] F. Wu, Z. Jiao, W. Li, Z. Zhang, H. Li, J. Wu, B. Jia, and S. Dong, "A vr-based robotic teleoperation system with haptic feedback and adaptive collision avoidance," *IEEE Transactions on Consumer Electronics*, 2025.

[20] 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 *IEEE/RAS International Conference on Intelligent Robots and Systems (IROS)*, 2014.

[21] M. Bauza, O. Canal, and A. Rodriguez, "Tactile mapping and localization from high-resolution tactile imprints," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2019.

[22] P. Sodhi, M. Kaess, M. Mukadanr, and S. Anderson, "Patchgraph: In-hand tactile tracking with learned surface normals," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2022.

[23] S. Luo, W. Mou, K. Althoefer, and H. Liu, "Localizing the object contact through matching tactile features with visual map," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015.

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

[25] 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.

[26] J. Zhao, M. Bauza, and E. H. Adelson, "Fingerslam: Closed-loop unknown object localization and reconstruction from visuo-tactile feedback," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2023.

[27] E. Olson, "Apriltag: A robust and flexible visual fiducial system," in *2011 IEEE international conference on robotics and automation*, pp. 3400–3407, IEEE, 2011.

[28] C. Heindl, "py-microdots: Position encoding in the euclidean plane based on the anoto codec," in *Science and Information Conference*, Springer, 2023.

[29] L. Hostettler, A. Özgür, S. Lemaignan, P. Dillenbourg, and F. Mondada, "Real-time high-accuracy 2d localization with structured patterns," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2016.

[30] S. Wang, Y. She, B. Romero, and E. Adelson, "Gelsight wedge: Measuring high-resolution 3d contact geometry with a compact robot finger," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021.

[31] "Gelsight mini tactile sensor." https://www.gelsight.com/gelsightmini/.

[32] S. Athar, G. Patel, Z. Xu, Q. Qiu, and Y. She, "Vistac toward a unified multimodal sensing finger for robotic manipulation," *IEEE Sensors Journal*, vol. 23, no. 20, pp. 25440–25450, 2023.