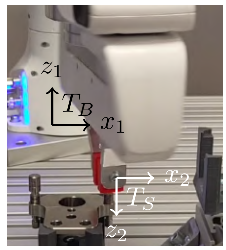
2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Tactile Execution Monitoring of Robotic Manipulation via Time-Series based Predictive Encoding

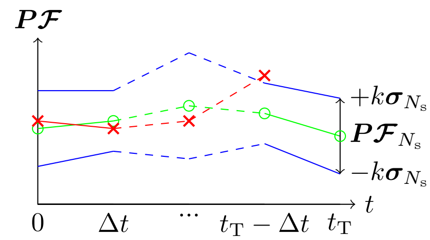
Florian Voigt, Abdeldjallil Naceri, and Sami Haddadin

***Abstract*— Tactile manipulation is a prominent and growing field where most research focuses on developing generalized manipulation policies. However, tactile execution monitoring - the ability to reliably evaluate manipulation at the skill level - is often overlooked, despite being critical for unsupervised deployment in both human-centered environments and industry, where strict safety and quality requirements apply. We propose the Tactile Predictive Encoding Model (TPEM), a time-series tactile perception framework inspired by human predictive encoding that enables real-time anomaly detection from skill-level sensory data. TPEM extends predictive coding concepts from global task modeling to precise monitoring of contact-rich manipulation beyond the capabilities of visual sensing.**

We evaluate TPEM on three representative tasks: key insertion and turning, peg-in-hole insertion, and screw insertion and tightening using an industrial assembly model. Experiments on a tactile-enabled Franka Emika robot under realistic noise conditions show robust anomaly detection with zero false positives. Comparison with baseline methods - including Support Vector Machines (SVM), Hidden Markov Models (HMM), and recurrent generative models such as LSTM-VAE — demonstrates that TPEM consistently outperforms state-of-the-art approaches in contact-rich skill-level execution monitoring.



(a)



(b)

Fig. 1: Tactile Predictive Encoding Model (TPEM): (a) The tactile manipulation policy and its predictive encoding model are represented in the skill frame T_S rather than the robot base frame T_B. (b) Using tactile data from successful trials, the light weight TPEM can distinguish between success (green) and failure (red) in real time, allowing the task to be aborted if necessary. Since both the policy and the TPEM are expressed in the skill frame, they can theoretically be used in a manipulator-agnostic manner and transferred from one manipulator system to another after being learned.

I. Introduction

Tactile execution monitoring (TEM)[1] in robotics is crucial for verifying correct task performance as well as guaranteeing safety (e.g., humanoid service robot). With the new generation of tactile robots and their application in human-robot collaboration and compliant manipulation, the need for new safety concepts has arisen. Although extensive research has examined potential injuries caused by robotic inertia and velocity [1], [2], safety measures are still often triggered only by violations of predefined wrench limits or by collision detection [3], [4]. However, explicit TEM in real-time during robotic manipulation has not been extensively researched.

To achieve safe and reliable tactile manipulation, adaptation, and reflexes comparable to humans, it is necessary to anticipate tactile and force feedback with respect to the task. This statement is inspired by the predictive encoding behavior of the human brain [5]. In this process, the brain is generating context-dependent models of the environment to predict future sensory input. Only this anticipation, even if only implicit or subconscious, allows humans to intuitively detect anomalies during manipulation. Such irregularities may be faults, such as clamping, missing contacts, or unexpected collisions. Another kind would be changes in the tasks' dynamic model, e.g. different stiffness of a button or additional friction in a lever, which might induce a lack of lubricant or upcoming failure. By incorporating multiple anticipation models of different task dynamics, the manipulator could distinguish between them. After identifying the correct instance, it could adapt its manipulation policy accordingly. Ideally, a system would be able to generate this anticipation on the fly during or immediately before task execution, based on its understanding of the environment's dynamics. While the latter is out of the scope of the current paper, we would like to make a first step towards such behavior.

While our concept of modeling predicted perception is related to the idea of predictive coding in robotics [6], our approach differs in two important aspects: (i) we operate at the *skill level* rather than the overall task level, and (ii) we address the specific challenges of *constrained tactile manipulation problems*. In contrast to existing anomaly detection methods summarized in Table I, our method explicitly leverages tactile feedback and predictive encoding within a skill frame formulation.

II. Contributions

Our main contributions are as follows:

- We introduce a skill frame specific Tactile Predictive Encoding Model (TPEM) for tactile manipulation with robotic hands and grippers. TPEM formulates tactile

[1]The authors are with Mohamed bin Zayed University of Artificial Intelligence (MBZUAI), Abu Dhabi, UAE. with The Mohamed bin Zayed University of Artificial Intelligence (MBZUAI), Abu Dhabi, UAE. This work was funded by the Lighthouse Initiative Geriatronics by the funding of the Lighthouse Initiative Geriatronics by LongLeif GaPa gGmbH (Project Y), the Kompetenzzentrum Mittelstand GmbH of Bavaria and KI.FABRIK Bayern (grant DIK0249). S. Haddadin has a potential conflict of interest as former shareholder of Franka Emika GmbH. Email:{florian.voigt, abdeldjallil.naceri, haddadin}@mbzuai.ac.ae

[1]Tactile Execution Monitoring refers to a framework that monitors sensory signals, specifically tactile feedback such as forces, torques, and position, from the robotic manipulation system in relation to the tasks being performed. This framework is designed to detect the success or failure of task execution based on these sensory inputs.

anomaly detection at the level of fine-grained manipulation skills and integrates orientation stabilization, temporal alignment, and decision-level robustness within a unified predictive coding framework. This bridges task-level predictive modeling and skill-level tactile execution in robotic assembly.
- We demonstrate the practical effectiveness of TPEM in demanding real-world industrial assembly scenarios with a tactile robot. Under strict statistical evaluation, TPEM achieved zero false positives—never labeling a failed manipulation as successful—which is critical for safety-sensitive robotic manipulation.
- We provide a systematic empirical comparison against representative state-of-the-art anomaly detection baselines, including classical one-class methods and deep generative approaches (Table I).

III. Related Work

A. Related Work

Execution monitoring and task execution monitoring have been studied for decades. An early overview is provided in [7], while high-level concepts of TEM based on abstract robot states were proposed in [8]. Despite this, advanced low-level TEM for contact-rich robotic control remains limited.

a) Classical and Statistical Approaches: Early works used autoregressive models or statistical techniques for anomaly detection. For example, [9] demonstrated anomaly detection in household manipulation using multimodal sensor data, while [6] compared sensory input against predictive models of demonstrated billiard-ball pushing. Other statistical approaches have been applied mainly to locomotion [10], [11] or collision detection [12], but they lack general applicability to tactile manipulation.

b) SVM and HMM Methods: A large body of work explored Support Vector Machines (SVMs) for post-execution success classification in assembly tasks [13]–[18]. These approaches rely on wrench signatures but are evaluated only after task completion, not in real-time. Hidden Markov Models (HMMs) have been applied to cooperative human-robot pick-and-place [19], flexible arms encountering obstacles [20], and grasping [21], but again operate at a coarse task-level rather than skill-level tactile interactions.

c) Deep Learning Approaches: Recent works employ LSTM-based models for anomaly detection in robotic tasks. OmniAnomaly [22]–[24] extends LSTM with a variational framework, but is applied to general system surveillance rather than manipulation. Similarly, SVAE approaches have been used for locomotion and lidar-based monitoring [25], not contact-rich skills. For manipulation, [26] used LSTM-VAE on multimodal sensory inputs for robot-assisted feeding. However, anomalies are detected only at the global task level rather than at fine-grained skill resolution. [27] and [28] use wrench or joint data for anomaly detection in assembly and pick-and-place, but rely on simple thresholds or detect only coarse deviations.

d) Summary: As reflected in Table I, most state-of-the-art approaches either (i) lack real-time capabilities, (ii) focus on global task-level detection rather than skill-level TEM, or (iii) are applied outside of contact-rich manipulation. Existing methods cover statistical models, classical machine learning (e.g., SVM, HMM), and deep generative approaches (e.g., LSTM-VAE, OmniAnomaly). However, none, to the best of our knowledge, fully address real-time skill-level anomaly detection in *contact-rich tactile manipulation*. This gap motivates our proposed TPEM, which introduces a skill-frame specific predictive encoding mechanism explicitly tailored to tactile anomaly detection during manipulation.

IV. Methods

A. Scope and Limitations

We assume that our manipulation strategies follow a specific motion and force profile for the given skill and can be executed and reproduced reasonably accurately by the robotic system. This holds even for kinematically redundant robots, such as the Franka Emika Robot [38], when controlling the end-effector in Cartesian task space. In structured environments, like factories, where tools and objects are placed at defined poses, standardized execution is often desired. A predictive model that encodes the dynamics of a specific manipulation skill[2] allows real-time evaluation of tactile perception, enabling detection of anomalies such as clamping, obstruction, material failures, missing lubricants, or breakages before excessive force is applied.

Multiple predictive encoding profiles can be combined to account for variations in task execution. Detected anomalies can trigger subsequent skills, stop the robot, or reverse executed motions, assuming fault-free execution until detection. Upon stopping, the robot may become compliant, allowing human inspection. While these reactions are included for completeness, they are not the main focus of this work.

In the remainder of this work, we adopt the following definitions: The skill frame is a task-aligned coordinate frame associated with the functional reference of a manipulation primitive (e.g., an insertion axis or a contact surface). The skill space refers to the set of pose and wrench variables expressed in the skill frame, as opposed to the robot's base or end-effector frames (see Fig. 1). Furthermore, note that the skill frame does not necessarily have to coincide with the robot end-effector frame.

B. The Tactile Predictive Encoding Model (TPEM)

Our approach uses only joint torque and position sensors, though it can incorporate additional sensors (e.g., IMUs). Inspired by Gaussian processes [39] and collision detection patents [40], [41], TPEM differs in that it: (i) represents the model compactly using equidistantly distributed segments connected by support points; (ii) computes mean $\boldsymbol{\mu}$ and variance $\boldsymbol{\sigma}$ piecewise; and (iii) omits computationally expensive Gaussian kernels without performance loss. In this work, we focus exclusively on tactile sensory input for contact-rich tasks, where force and contact dynamics are critical and cannot be reliably captured by visual sensing. Visual and audio modalities are left for future work.

1) Data Preparation: Execute a task N times and divide the obtained time series into N_s equidistant segments of length Δt_s. Each trajectory is extended by duplicating its final data tuple to ensure consistency with the maximum trajectory length:

$$t_\mathrm{T} = \left\lceil \frac{\max(t_{\mathrm{T}_1}, ..., t_{\mathrm{T}_N})}{\Delta t_s} \right\rceil \Delta t_s, \quad N_\mathrm{s} = \frac{t_\mathrm{T}}{\Delta t_\mathrm{s}}. \quad (1)$$

[2]"Predictive" denotes phase-conditioned estimation of the expected tactile distribution at each time index, not long-horizon forecasting.

TABLE I: State-of-the-art comparison of anomaly detection methods for tactile/multivariate time-series data. Legend: ✓= Supported, (✓) = Partially supported/conditional, – = not supported. Computational complexity: ✓= efficient, (✓) = moderate, – = heavy.

Algorithm	**Real-time Detection**	**High Scalability**	**Anomaly Classification**	**Robustness to Noise**	**Interpret-ability**	**Computational Complexity**
ARIMA / VAR [29]	–	–	–	✓	✓	(✓)
Isolation Forest [30]	✓	✓	–	(✓)	–	✓
One-Class SVM [31]	–	–	–	–	–	–
LSTM / EncDec [32]	✓	–	✓	(✓)	–	–
LSTM-VAE / GRU-VAE [33], [34]	✓	–	✓	✓	(✓)	–
DAGMM [35]	–	–	✓	(✓)	–	–
OmniAnomaly [34]	(✓)	(✓)	(✓)	✓	(✓)	–
Anomaly Transformer / TranAD [36], [37]	(✓)	✓	✓	(✓)	–	(✓)
Ours (TPEM)	✓	✓	✓	✓	✓	(✓)

2) Robot Dynamics: The dynamics of a robot with n degrees of freedom in joint space are given by:

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{c}(\boldsymbol{q},\dot{\boldsymbol{q}}) + \boldsymbol{h}(\boldsymbol{q},\dot{\boldsymbol{q}}) + \boldsymbol{g}(\boldsymbol{q}) - \boldsymbol{\tau}_{\text{ext}}, \quad (2)$$

where $\boldsymbol{\tau}$, $\boldsymbol{\tau}_{\text{ext}} \in \mathbb{R}^n$ are the actuation torque and the external torque. $\boldsymbol{M} \in \mathbb{R}^{n\times n}$ is the inertia matrix and $\boldsymbol{c}$, $\boldsymbol{h}$, $\boldsymbol{g} \in \mathbb{R}^n$ are the Coriolis, the friction and the gravitational terms, respectively.

The external torque can be estimated as

$$\hat{\boldsymbol{\tau}}_{\text{ext}} = \hat{\boldsymbol{M}}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \hat{\boldsymbol{c}}(\boldsymbol{q},\dot{\boldsymbol{q}}) + \hat{\boldsymbol{h}}(\boldsymbol{q},\dot{\boldsymbol{q}}) + \hat{\boldsymbol{g}}(\boldsymbol{q}) - \boldsymbol{\tau}, \quad (3)$$

where $\hat{\cdot}$ denotes the modeled terms.

3) TPEM in Skill Space: External wrench in skill space is estimated via the Jacobian w.r.t. the skill space $\boldsymbol{J}_{\text{S}} \in \mathbb{R}^{n\times n}$:

$$\hat{\boldsymbol{\mathcal{F}}}_{\text{ext}} = (\boldsymbol{J}_{\text{S}}\boldsymbol{J}_{\text{S}}^T)^{-1}\boldsymbol{J}_{\text{S}}\hat{\boldsymbol{\tau}}_{\text{ext}}. \quad (4)$$

For the start and end points of each of the segments mentioned in Sec. IV-B.1 the average positions $\tilde{\boldsymbol{p}}_i$, orientations $\tilde{\boldsymbol{\phi}}_i$, and external wrenches $\tilde{\boldsymbol{\mathcal{F}}}_{\text{ext},i}$ are computed. The orientation averages are obtained via the principal eigenvector of quaternions [42].

a) Positioning of Technical Design: The components used in TPEM, including quaternion averaging [42] and window-based anomaly decision strategies [43], are established techniques. Our contribution lies in their integration within a skill frame predictive encoding formulation for tactile manipulation, enabling robust real-time monitoring in contact-rich assembly tasks.

The resulting skill level predictive encoding is

$$pe = \{(\tilde{\boldsymbol{P\mathcal{F}}}_0, \boldsymbol{\sigma}_0), .., (\tilde{\boldsymbol{P\mathcal{F}}}_{N_{\text{s}}}, \boldsymbol{\sigma}_{N_{\text{s}}})\}. \quad (5)$$

This representation is illustrated as a “hose” around the mean trajectory in Fig. 1b.

C. Using the Model for Execution Monitoring

At runtime, the expected perception $\tilde{\boldsymbol{P\mathcal{F}}}_t$ and variance $\boldsymbol{\sigma}_t$ are interpolated as follows:

$$n_{\text{Index}} = \left\lfloor \frac{t}{\Delta t_{\text{s}}} \right\rfloor, \quad (6)$$

$$\begin{aligned}\tilde{\boldsymbol{P\mathcal{F}}}_t =& \tilde{\boldsymbol{P\mathcal{F}}}_{n_{\text{Index}}+1}\left(\frac{t}{\Delta t_{\text{s}}} - n_{\text{Index}}\right) + \\ & \tilde{\boldsymbol{P\mathcal{F}}}_{n_{\text{Index}}}\left(1 - \left(\frac{t}{\Delta t_{\text{s}}} - n_{\text{Index}}\right)\right),\end{aligned} \quad (7)$$

Fig. 2: Number of detected anomalies in successful trials for different hyperparameter values for the peg insertion task.

$$\begin{aligned}\boldsymbol{\sigma}_t =& \boldsymbol{\sigma}_{n_{\text{Index}}+1}\left(\frac{t}{\Delta t_{\text{s}}} - n_{\text{Index}}\right) + \\ & \boldsymbol{\sigma}_{n_{\text{Index}}}\left(1 - \left(\frac{t}{\Delta t_{\text{s}}} - n_{\text{Index}}\right)\right).\end{aligned} \quad (8)$$

We detect outliers by combining the Z-score method which is multiplied over the last l measurements:

$$y_t = \prod_{j=0}^{l-1} H\left(\left|\frac{\boldsymbol{m}_{t-j} - \tilde{\boldsymbol{P\mathcal{F}}}_{t-j}}{\boldsymbol{\sigma}_{t-j}}\right| - k\right). \quad (9)$$

H denotes the Heaviside function and $\hat{\boldsymbol{m}}$, y_t the tactile robot measurement on skill level and the boolean result of the anomaly monitoring process, respectively. When $y = 1$, the task is classified as anomalous, triggering collision handling.

Furthermore, for the remainder of this paper, we set $l = 20$ and $k = 4$.

a) Parameter Selection: The threshold k defines a Z-score confidence bound of the predictive model, while l enforces temporal consistency to suppress transient noise. The hyperparameter values ($k = 4$, $l = 20$) were selected via validation to balance false positive suppression and detection latency (see Fig. 2). We initially identified values that yielded zero false detections ($k = 3$, $l = 15$) and then increased them to $k = 4$ and $l = 20$ as a safety margin.

b) Comparison to State-of-the-Art: Table I summarizes the capabilities of representative anomaly detection methods for tactile or multivariate time-series data. Classical models such as ARIMA, VAR, and One-Class SVM do not support real-time skill-level monitoring and are limited in robustness to noise. Deep generative approaches like LSTM-VAE and OmniAnomaly can perform real-time detection, but they typically operate at a coarse task level and do not explicitly encode skill frame-specific expectations of sensory signals. Even transformer-based approaches, e.g., Anomaly

TABLE II: Overview of number of trials for each experimental insertion task for training and evaluation.

Object	Training	Possible States	Evaluation
Key	20	Success / Incomplete / Failure	60, 20 per state
Peg	20	Success / Failure	40, 20 per state
Screw	20	Success / Failure	40, 20 per state

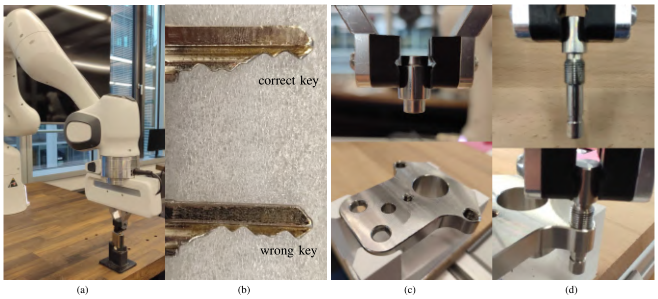


Fig. 3: (a) The tactile Franka Emika robot inserting and turning a key and (b) the used keys. (c) The steel peg and the according insertion holes. (d) A screw which is inserted and tightened.

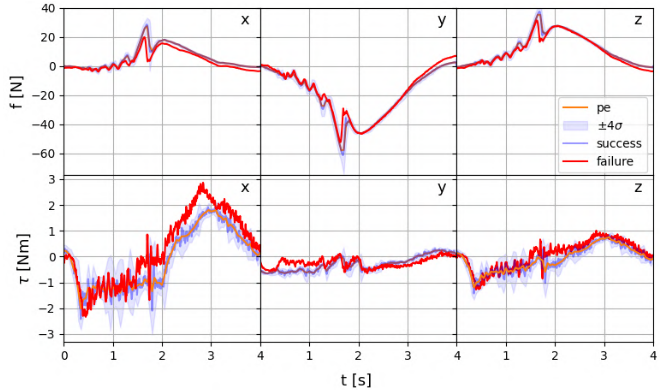


Fig. 4: The predictive encoding model of the wrench for the key insertion task. For comparison one successful and one unsuccessful trial is shown.

Transformer, are primarily designed for system-level monitoring and rely on global correlations rather than the local, time-segmented variance structure needed for contact-rich manipulation.

In contrast, TPEM constructs a predictive "hose" around mean trajectories of pose and wrench signals, segment-wise computing mean $\tilde{\boldsymbol{P}\mathcal{F}}_t$ and variance $\boldsymbol{\sigma}_t$ (Eqs. 6–8). This enables instantaneous evaluation of each measurement against the predicted distribution:

$$\tilde{\boldsymbol{P}\mathcal{F}}_t - k\boldsymbol{\sigma}_t < \boldsymbol{m}_t < \tilde{\boldsymbol{P}\mathcal{F}}_t + k\boldsymbol{\sigma}_t, \tag{10}$$

allowing robust detection of deviations even within millimeter-scale tolerances. The segment-wise representation supports real-time updates, and is robust to transient outliers via the variable l if the Heaviside function. This mathematical formulation, together with skill frame-specific variance modeling, directly addresses the limitations of existing methods shown in Table I, providing both skill-level resolution and real-time detection for contact-rich manipulation tasks.

V. Real world experiments

In the real-world experiments, we evaluated our approach on three *tactile, contact-rich manipulation tasks*:

1) inserting a key into a lock and turning it,
2) inserting a peg into a hole *(using an industrial partner's assembly model routinely employed on the factory floor)*, and
3) inserting and tightening a screw.

The objective was to assess the robot's ability to reliably detect obstructions or insertion errors while avoiding excessive force during execution. All experiments were conducted on a *tactile-enabled Franka Emika robot*, representative of industrial assembly-line use.

The trials were performed under *significant noise conditions*, arising from control dynamics, sensor measurements, and task-intrinsic contacts. We implemented a simple force calibration by recording the unloaded sensor baseline and subtracting it during manipulation, effectively compensating for sensor drift. For this study, we assumed that the same robot was used for both training and execution; therefore, calibration discrepancies across different robots were not explicitly addressed. We acknowledge that training on one robot and deploying on another with different calibration parameters may introduce deviations, which we plan to investigate in future work.

For each of the three tasks, we conducted **20 trials** to train the predictive encoding models. For evaluation and validation, we executed **20 additional trials** per experimental state, where the states were defined as *Success*, *Failure*, and *Incomplete* (the latter applying to the key insertion task). Table II shows an overview of the number of trials for each task.

A. Key Insertion

The model was trained on data obtained for the successful insertion of the correct key. We evaluated if the model was able to discriminate between different keys, i.e. detect keys incompatible with the lock, and incomplete insertion before turning. The robot and lock are shown in Fig. 3a and the correct key and an example of a wrong key in Fig. 3b. Figure 4 shows the predictive encoding model for the successful insertion of the correct key. We examined and validated our model within the following different setups (state): The correct key with correct task execution, the correct key with incomplete insertion, and a similar but different key, which can be inserted into the lock easily. At the beginning of the task, the key is grasped by the robot and positioned right in front of the lock. In this validation phase, We conducted 20 trials for each actual state, 60 in total. These three different cases were reliably detected in our experiments without any false negatives or false positives. Figure 6 shows the corresponding confusion matrix.

B. Peg Insertion

In this application, we examined a state-of-the-art of industrial assembly task. A small stainless steel peg is grasped by the 2 finger gripper of the robot and is inserted into a hole. The material of the hole is also made of steel and has a diameter of 8 mm with a clearance of 0.1 mm, see Fig. 3c. For this, the peg is placed with a small offset to

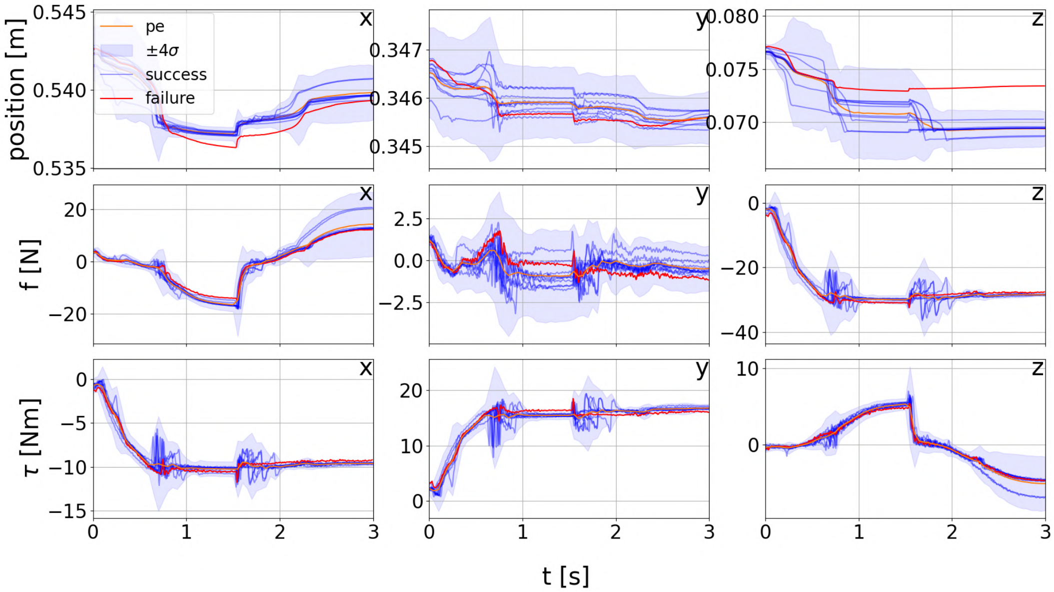


Fig. 5: Predictive encoding model of the peg insertion task shown for Cartesian position, force and torque. Multiple successful trials are shown to demonstrate compliance with the model. One example of a failed attempt is also shown. In the model, two task-specific ripple clusters at around 0.75 and 1.7 s can be seen. Obviously, in this task especially the z-position is crucial for success.

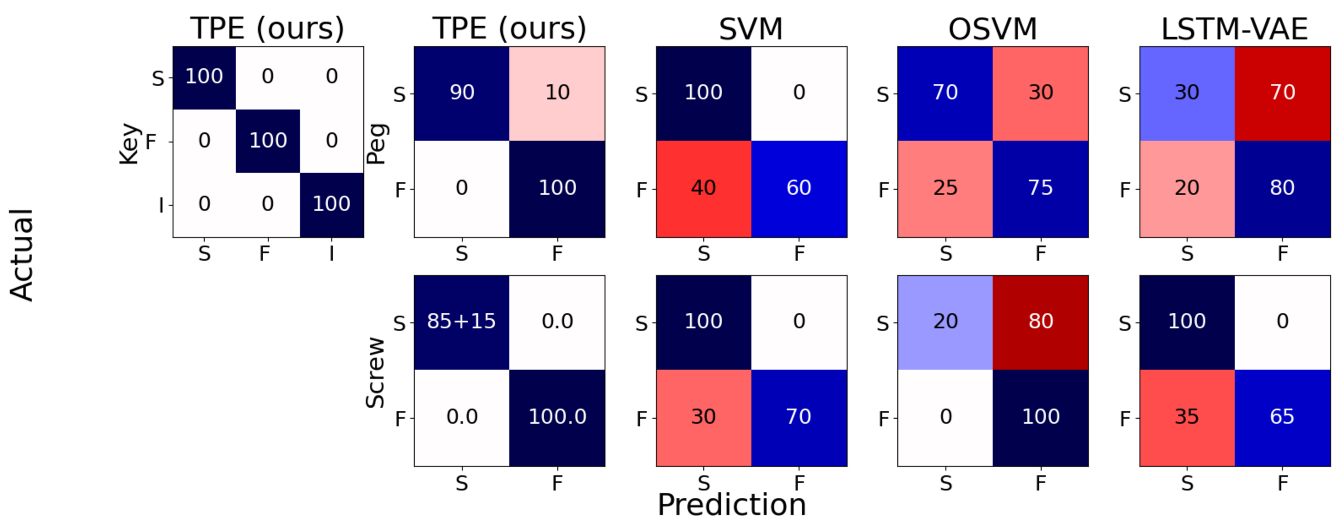


Fig. 6: Confusion matrix of the examined tasks for our approach TPE and baseline methods. The states are **S**uccess, **F**ailure and **I**ncomplete insertion. TPE: 20 trials were executed for each actual outcome. **Key**) All cases are distinguished reliably. **Peg**) 10 % false negatives and 0 % false positives were observed. **Screw**) In 15 % of the successful cases, the thread was not detected on the first rotation (85 %), but on the second one (15 %). Baseline is given for the peg in hole and the screw thread detection task for standard SVM, One-Class SVM classification and LSTM-VAE.

the hole in the x-y plane and moved to a point opposite of the hole and back, while pushing the peg down. The predictive encoding model of the task is shown in Fig. 5. We executed 20 successful and 20 unsuccessful trials (40 trials in total). For the unsuccessful ones, we included multiple different cases: 1) No peg was grasped, 2) the hole was obstructed with small paper or rubber pieces, 3) the hole was already blocked by a peg and 4) lubricants like water or oil were present. These lubricants changed the friction around the hole, which changed the robot's perception during insertion. While the presence of such liquids does not make the insertion fail per se it is often undesired or vice versa. Figure 6 shows that we could identify failure reliably and had 2 false negative classifications. While these false negatives are undesired, the far more serious case of false positives was not observed.

C. Screw Insertion

In this representative industrial assembly task, the screw is made of stainless steel and has a diameter of 7 mm and a thread pitch of < 1 mm. It is grasped using a 2 finger gripper and inserted into the hole. Next, we rotate the screw counterclockwise while pushing down to find the beginning of the thread. After finding it, we subsequently try to tighten the screw. Screw rotation w.r.t. the gripper is random, making it a challenging problem to locate the thread due to the dynamic interplay of rotational and translational movements. When the thread is detected, slight vibrations in the system can be observed. In this manipulation task, we executed 40 trials in total (20 trials in each state). The confusion matrix Fig. 6 shows that we were able to detect the thread on the first revolution in 85 % of all successful cases and the remaining 15 % on the second revolution. A plot of the model is provided in the supplementary materials.

D. Comparison with Baseline Methods

To evaluate the effectiveness of our TPEM, we compare it against representative state-of-the-art methods from classical machine learning and deep generative approaches. These baselines are widely used for robotic anomaly detection, yet they typically operate at a global task level or are less sensitive to fine-grained tactile and contact signals. By performing a controlled comparison on real-world industrial assembly tasks, we assess TPEM's ability to detect anomalies in real-time, skill-level, contact-rich manipulation scenarios.

In the real-world experiments, we validated TPEM on three contact-rich industrial manipulation tasks: (i) inserting a key into a lock and turning it, (ii) inserting a peg into a hole, and (iii) inserting and tightening a screw. These tasks were executed with a tactile-enabled Franka Emika robot, a platform widely used in industrial assembly settings. Our goal was to evaluate the robot's ability to detect obstructions or insertion errors while avoiding excessive force requirements directly relevant to assembly-line operations performed by our industrial partners in their daily routines.

We compare TPEM against classical machine learning methods (e.g., standard SVM, One-Class SVM (OSVM), HMM) and deep generative models (e.g., LSTM-VAE, OmniAnomaly). Other advanced methods, such as Anomaly Transformers, are not included as baselines because they are primarily designed for system-level time-series monitoring and have not been demonstrated for real-time, skill-level, contact-rich tactile manipulation tasks. This ensures that our comparisons focus on methods that are both relevant and feasible to adapt to the specific constraints of tactile skill monitoring, thereby highlighting the novelty and effectiveness of TPEM.

We evaluate these approaches on peg-in-hole and screw assembly tasks. Standard SVMs were trained on both successful and unsuccessful trials, while OSVMs were trained solely on successful trials from a designated training set. Each model was evaluated on a separate test set consisting of 20 successful and 20 unsuccessful trials per task. For training and anomaly detection, pose and wrench sensory data collected every 50 ms were used as input for the SVMs and LSTM-based models, with a window length of 10 samples chosen as an effective heuristic for our data[3].

Figure 6 presents the confusion matrices. Standard SVMs showed a tendency for false positives at a rate of 30–40 %. OSVMs primarily produced false negatives for the screw task and both false negatives and false positives for the peg-in-hole task. Overall, the accuracy of these models was not satisfactory, likely due to the highly constrained and contact-rich nature of the tasks, where even small deviations of 1 mm or less significantly affect the outcome.

The LSTM-VAE showed false positives (70 %) for the peg task, mainly due to insufficient capture of contact amplitudes, and false negatives (20 %) caused by discrepancies between predicted signals and the ground truth mean. In the thread detection task, LSTM-VAE was sensitive to true positives but exhibited notable false positives (35 %).

VI. Discussion and Conclusion

In this work, we introduced the Tactile Predictive Encoding Model (TPEM), a model-free execution monitoring approach for skill-level tactile manipulation. Unlike prior methods operating at the task level [13], [15], [16], [26], [27] or in locomotion and surveillance domains [10], [22], [25], TPEM leverages predictive encoding of tactile feedback within a task-frame formulation, addressing a gap summarized in Table I.

Experiments were conducted under realistic conditions with substantial noise from control, sensing, and contact dynamics. Across three manipulation tasks (20 training and 20 validation trials per condition), TPEM achieved robust anomaly detection with zero false positives—a critical safety requirement. Compared to classical SVM-based approaches [13]–[18], one-class methods [20], recurrent generative models such as LSTM-VAE [26], and deep anomaly detectors including OmniAnomaly [22], TPEM consistently outperformed in contact-rich scenarios requiring fine-grained skill monitoring.

Although TPEM focuses on tactile monitoring, positional signals are not treated as tactile input but provide phase alignment for interpreting contact dynamics. In tasks such as peg insertion, force deviations must be evaluated relative to motion progression (approach, insertion, seating). Successful insertions exhibit structured ripple patterns in the coupled z–force signal, while anomalies disrupt this pattern. Modeling such cross-correlations explicitly may further enhance detection and represents future work.

Conceptually, TPEM extends predictive coding [6] to skill-specific tactile interaction, shifting monitoring from post-hoc detection toward real-time validation. Validation with industrial partners highlights its practical relevance for assembly-line operations.

Current evaluation assumes identical training and deployment platforms; calibration differences across robots may affect generalization and will be addressed in future work. We further plan to incorporate multimodal sensing (tactile, audio, vision) at the skill level to enable transferable perception across manipulation primitives.

In summary, TPEM provides an interpretable and scalable framework for anomaly detection in contact-rich robotic manipulation, offering a principled foundation for resilient real-world autonomy.

[3]Plots and visual comparison to our approach are given in the supplementary materials.

References

[1] M. Vasic and A. Billard, "Safety issues in human-robot interactions," in *2013 IEEE International Conference on Robotics and Automation*, 2013, pp. 197–204.

[2] S. Haddadin, S. Haddadin, A. Khoury, T. Rokahr, S. Parusel, R. Burgkart, A. Bicchi, and A. Albu-Schäffer, "On making robots understand safety: Embedding injury knowledge into control," *The International Journal of Robotics Research*, vol. 31, no. 13, pp. 1578–1602, 2012.

[3] S. Hjorth and D. Chrysostomou, "Human–robot collaboration in industrial environments: A literature review on non-destructive disassembly," *Robotics and Computer-Integrated Manufacturing*, vol. 73, p. 102208, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0736584521000910

[4] S. Haddadin, A. De Luca, and A. Albu-Schäffer, "Robot collisions: A survey on detection, isolation, and identification," *IEEE Transactions on Robotics*, vol. 33, no. 6, pp. 1292–1312, 2017.

[5] R. Reichardt, B. Polner, and P. Simor, "Novelty manipulations, memory performance, and predictive coding: the role of unexpectedness," *Frontiers in Human Neuroscience*, vol. 14, p. 152, 04 2020.

[6] P. Pastor, M. Kalakrishnan, S. Chitta, E. Theodorou, and S. Schaal, "Skill learning and task outcome prediction for manipulation, year=2011, volume=, number=, pages=3828-3834, doi=10.1109/ICRA.2011.5980200," in *2011 IEEE International Conference on Robotics and Automation*. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/5980200

[7] O. Pettersson, "Execution monitoring in robotics: A survey," *Robotics and Autonomous Systems*, vol. 53, no. 2, pp. 73–88, 2005. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S092188900500134X

[8] R. Reiter and M. Soutchanski, "Execution monitoring of high-level robot programs." 02 2002. [Online]. Available: https://www.diag.uniroma1.it//degiacom/papers/1998/DeRS98kr.pdf

[9] D. Park, Z. Erickson, T. Bhattacharjee, and C. C. Kemp, "Multimodal execution monitoring for anomaly detection during robot manipulation," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 407–414. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7487160

[10] E. Gat, M. G. Slack, D. P. Miller, and R. J. Firby, "Path planning and execution monitoring for a planetary rover," in *Proceedings., IEEE International Conference on Robotics and Automation*, 1990, pp. 20–25 vol.1. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/125939

[11] S. Calinon and A. Billard, "A probabilistic programming by demonstration framework handling constraints in joint space and task space," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008, pp. 367–372.

[12] F. Dimeas, L. Avendano-Valencia, E. Nasiopoulou, and N. Aspragathos, "Robot collision detection based on fuzzy identification and time series modelling," 09 2013.

[13] A. Rodriguez, D. Bourne, M. Mason, G. Rossano, and J. Wang, "Failure detection in assembly: Force signature analysis," 09 2010, pp. 210 – 215.

[14] S. Golz, C. Osendorfer, and S. Haddadin, "Using tactile sensation for learning contact knowledge: Discriminate collision from physical interaction," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 3788–3794.

[15] E. Moreira, L. F. Rocha, A. M. Pinto, A. P. Moreira, and G. Veiga, "Assessment of robotic picking operations using a 6 axis force/torque sensor," *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 768–775, 2016.

[16] J. Rojas, S. Luo, D. Zhu, Y. Du, H. Lin, Z. Huang, W. Kuang, and K. Harada, "Online robot introspection via wrench-based action grammars," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 5429–5436.

[17] S. Doltsinis, M. Krestenitis, and Z. Doulgeri, "A machine learning framework for real-time identification of successful snap-fit assemblies," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 1, pp. 513–523, 2020.

[18] A. Stolt, M. Linderoth, A. Robertsson, and R. Johansson, "Detection of contact force transients in robotic assembly," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 962–968.

[19] H. Wu, Y. Guan, and J. Rojas, "A latent state-based multimodal execution monitor with anomaly detection and classification for robot introspection," *Applied Sciences*, vol. 9, no. 6, 2019. [Online]. Available: https://www.mdpi.com/2076-3417/9/6/1072

[20] E. Di Lello, M. Klotzbücher, T. De Laet, and H. Bruyninckx, "Bayesian time-series models for continuous fault detection and recognition in industrial robotic tasks," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 5827–5833.

[21] A. Rodriguez, M. T. Mason, S. S. Srinivasa, M. Bernstein, and A. Zirbel, "Abort and retry in grasping," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011, pp. 1804–1810.

[22] Y. Su, Y. Zhao, C. Niu, R. Liu, W. Sun, and D. Pei, "Robust anomaly detection for multivariate time series through stochastic recurrent neural network," 07 2019, pp. 2828–2837.

[23] Q. He, Y. Zheng, C. Zhang, and H. Wang, "Mtad-tf: Multivariate time series anomaly detection using the combination of temporal pattern and feature pattern," *Complexity*, vol. 2020, pp. 1–9, 10 2020.

[24] J. Kang, C.-S. Kim, J. Kang, and J. Gwak, "Anomaly detection of the brake operating unit on metro vehicles using a one-class lstm autoencoder," *Applied Sciences*, vol. 11, p. 9290, 10 2021.

[25] T. Ji, S. T. Vuppala, G. Chowdhary, and K. Driggs-Campbell, "Multi-modal anomaly detection for unstructured and uncertain environments," in *Proceedings of the 2020 Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, J. Kober, F. Ramos, and C. Tomlin, Eds., vol. 155. PMLR, 16–18 Nov 2021, pp. 1443–1455. [Online]. Available: https://proceedings.mlr.press/v155/ji21a.html

[26] D. Park, Y. Hoshi, and C. Kemp, "A multimodal anomaly detector for robot-assisted feeding using an lstm-based variational autoencoder," *IEEE Robotics and Automation Letters*, vol. PP, 11 2017.

[27] J. Watson, A. Miller, and N. Correll, "Autonomous industrial assembly using force, torque, and RGB-d sensing," *Advanced Robotics*, pp. 1–14, feb 2020. [Online]. Available: https://doi.org/10.48550/arXiv.2002.02580

[28] *KrakenBox: Deep Learning-Based Error Detector for Industrial Cyber-Physical Systems*, ser. ASME International Mechanical Engineering Congress and Exposition, vol. Volume 13: Safety Engineering, Risk, and Reliability Analysis; Research Posters, 11 2021. [Online]. Available: https://doi.org/10.1115/IMECE2021-70258

[29] G. E. P. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time Series Analysis: Forecasting and Control*, 5th ed. John Wiley & Sons, 2015.

[30] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *2008 Eighth IEEE International Conference on Data Mining*. IEEE, 2008, pp. 413–422.

[31] B. Schölkopf, J. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural Computation*, vol. 13, no. 7, pp. 1443–1471, 2001.

[32] P. Malhotra, L. Vig, G. Shroff, and P. Agarwal, "Lstm-based encoder-decoder for multi-sensor anomaly detection," *arXiv preprint arXiv:1607.00148*, 2016.

[33] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2014.

[34] Y. Su, Y. Zhao, C. Niu, R. Liu, W. Sun, J. Pei, and J. Pei, "Robust anomaly detection for multivariate time series through stochastic recurrent neural network," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 2828–2837.

[35] B. Zong, Q. Song, M. Min, W. Cheng, C. Lumezanu, D. Cho, and H. Chen, "Deep autoencoding gaussian mixture model for unsupervised anomaly detection," in *International Conference on Learning Representations (ICLR)*, 2018.

[36] J. Xu, H. Wu, J. Wang, and G. Long, "Anomaly transformer: Time series anomaly detection with association discrepancy," in *International Conference on Learning Representations (ICLR)*, 2021.

[37] S. Tuli, G. Casale, and N. R. Jennings, "Tranad: Deep transformer networks for anomaly detection in multivariate time series data," in *Proceedings of the VLDB Endowment*, vol. 15, no. 6, 2022, pp. 1201–1214.

[38] S. Haddadin, S. Parusel, L. Johannsmeier, S. Golz, S. Gabl, F. Walch, M. Sabaghian, C. Jähne, L. Hausperger, and S. Haddadin, "The franka emika robot: A reference platform for robotics research and education," *IEEE Robotics & Automation Magazine*, vol. 29, no. 2, pp. 46–64, 2022.

[39] E. Schulz, M. Speekenbrink, and A. Krause, "A tutorial on gaussian process regression: Modelling, exploring, and exploiting functions," *Journal of Mathematical Psychology*, vol. 85, pp. 1–16, 2018.

[40] S. Parusel and S. Golz, "Area-dependent collision detection for a robot manipulator," Germany Patent DE102 018 112 360B3, 9 19, 2019. [Online]. Available: https://patents.google.com/patent/DE102018112360B3/en?oq=DE102018112360

[41] S. P. and Saskia Golz, "Direction-dependent collision detection for a robot manipulator," Germany Patent DE102 018 112 370A1, 9 16, 2021. [Online]. Available: https://patents.google.com/patent/DE102018112370A1/en?oq=DE102018112370A1

[42] L. Markley, Y. Cheng, J. Crassidis, and Y. Oshman, "Averaging quaternions," *Journal of Guidance, Control, and Dynamics*, vol. 30, pp. 1193–1196, 07 2007.

[43] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Computing Surveys*, vol. 41, no. 3, pp. 15:1–15:58, 2009.