

Appendix

A Datasets

A.1 Dataset for TacX SSL Pretraining

Our self-supervised learning (SSL) dataset is sourced from two platforms: an Allegro hand equipped with Digit 360 sensors mounted on each fingertip, and a custom mobile picker tool with sensors integrated into its gripping mechanism. Since SSL representation learning does not require labeled data, we collect tactile data by having the Allegro hand interact rummaging freely with a tray filled with LEGO blocks and marbles. This setup enables the capture of rich, multi-contact interactions with objects that feature distinctive geometries, such as spherical shapes and sharp edges.

The mobile picker is used to gather tactile data from everyday manipulation actions, including tapping and sliding, across surfaces with varying friction and stiffness. This allows us to record both intrinsic and extrinsic contact interactions. We collected eight sequences with the Allegro hand, each lasting approximately 8.5 minutes, recording data from all four fingers. From the mobile picker, we gathered 104 sequences with an average duration of 3 minutes, logging data from both sensors on the gripper. For the subset of the dataset collected with the mobile picker, we provide annotations indicating the object in grasp, the action performed, and the surface in contact, to support downstream evaluation tasks. In total (see Figure 8), our dataset spans 18.6 hours of tactile data collected from six different Digit 360 sensors.

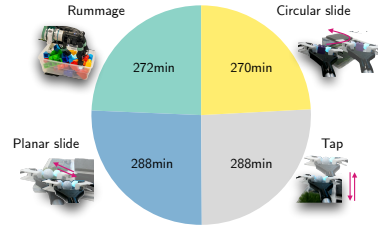


Figure 8: Distribution of recorded Digit 360 data by platform. The dataset includes 18.6 hours of data collected using two platforms: the Allegro hand (4.5 hours) and the mobile picker tool (14.1 hours).

TacX processes temporal windows of data from each tactile modality. A visualization of the input data is shown in Figure 9.

Images. We input pairs of tactile images sampled with a temporal stride of 5, concatenated along the channel dimension. These images are captured using a hyperfisheye lens in Digit 360, allowing us to view the entire dome-shaped elastomer surface. Unlike planar GelSight-like sensors, these images include reflections from the surrounding LED light sources, visible near the center of the dome. While these reflections act as useful markers, encoding meaningful information about gel deformation upon contact, they also pose challenges for standard preprocessing techniques such as background subtraction or lighting augmentations, which risk corrupting the contact signal.

Audio. Digit 360 sensor is equipped with two contact microphones that capture vibrations, sampled at 48kHz. This signal is especially informative for detecting changes in contact state, such as making and breaking contact. TacX processes 0.5sec windows of audio data from each microphone in the frequency domain. After standardization and conversion to log-mel spectrograms, the audio is treated as a single-channel image input to the model.

IMU and Pressure. We extract 0.5 second and 1 second windows of data from the 3-axis accelerometer and the static pressure sensor embedded in the Digit 360. Each window is standardized using the mean and standard deviation computed per sequence to ensure consistency across variations in sensor signal amplitude.

A.2 Datasets for Downstream Tasks

For each experiment related to estimating physical properties with TacX (see Section 4.1), we designed custom setups to collect training data tailored to each task.

For **Object-Action-Surface Classification**, we repurpose the SSL pretraining dataset collected with the manual picker. We annotate each data point with metadata specifying the object being held

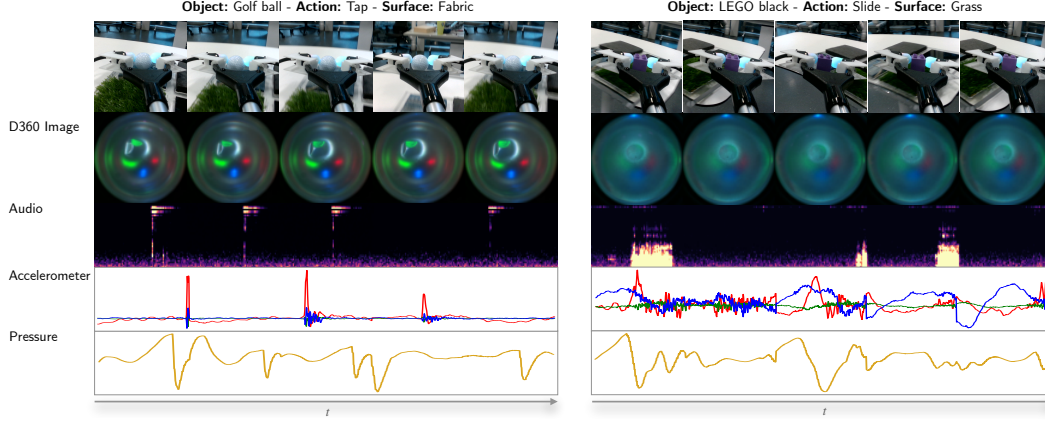


Figure 9: Visualization of each of the tactile input modalities to TacX. Samples from pretraining dataset.

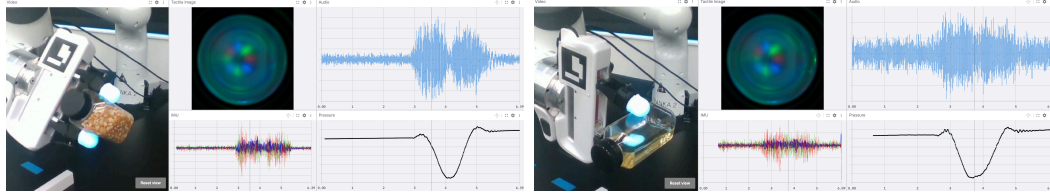


Figure 10: Visualization of the experimental setup and tactile sensory inputs for the material-quantity classification dataset. The setup involves shaking bottles filled with different materials and quantities using the Franka's gripper equipped with Digit 360 sensors.

534 (golf ball, wood block, LEGO block), the action performed (tap, linear slide, circular slide), and the
 535 external surface in contact (grass, fabric, plastic, foamwork). From the 104 available sequences, 69
 536 are used for training and the remaining 35 for testing the performance of the classifier. The dataset is
 537 balanced in number of samples per label.

538 For **Material-Quantity Estimation**, we design a 3D-printed gripper attachment to mount the
 539 Digit 360 sensors onto the Franka arm. The data collection protocol involves shaking six different
 540 8oz bottles containing various materials (lentils, rice, corn kernels, vitamin pills, water, and oil) at
 541 different fill levels (full, half, quarter). An illustration of this setup is provided in Figure 10. For

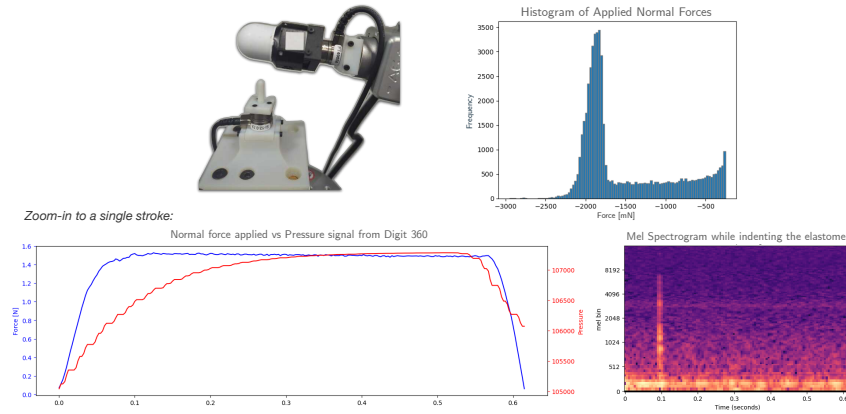


Figure 11: Top: Experimental setup and data distribution for the normal force regression experiment. **Bottom:** Zoom-in to a single indentation stroke. Note that the pressure signal from the Digit 360 sensor correlates well with the ground-truth normal force measured by the force/torque sensor beneath the hemispherical probe. The mel spectrogram also reveals the moment of contact between the probe and the elastomer.

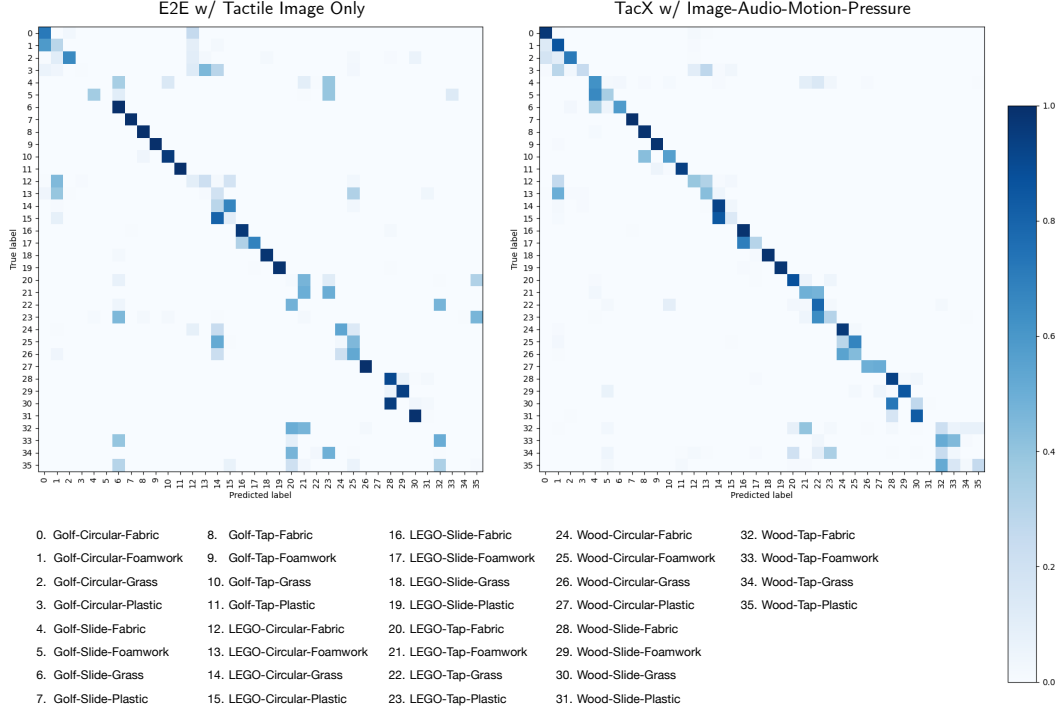


Figure 12: Confusion matrix for object-action-surface classification. We compare an end-to-end classifier trained solely on tactile images with a classifier trained on frozen TacX representations, under a 50% training data budget.

shaking the bottles to create variation in the tactile signal, the Franka’s gripper is rotated left and right in randomized motion patterns, including variation in the initial angle. We collect 20 trajectories for each material-quantity combination, using 15 sequences for training and reserving the remaining 5 for evaluating the classifier.

For **Normal Force Estimation**, we fix a hemispherical probe to a force/torque sensor and mount a Digit 360 sensor on the Meca arm, which is used to indent the elastomer surface perpendicularly, applying controlled normal forces of up to 3.5N. Figure 11 illustrates the experimental setup and the distribution of the collected data. We observe that the pressure modality correlates strongly with both the magnitude of the applied normal force and the location of the resulting deformation on the elastomer. The audio modality captures discrete events, such as the initial contact between the probe and the sensor surface.

B Benchmarking TacX for physical properties comprehension

Object-Action-Surface Classification. This task evaluates whether TacX can capture tactile cues that enable the identification of objects through both intrinsic and extrinsic contact interactions. The goal is to jointly classify the object being grasped, the action performed, and the surface in contact. The selected objects and surfaces span a range of properties, including texture, hardness, and friction.

We use representations from TacX to train a downstream classifier on the dataset described in Appendix A.2. Figure 12 shows confusion matrices on the test set for two classifiers: one trained using frozen TacX representations with all tactile modalities as input, and another trained end-to-end (E2E) using only tactile images. Note that the classification task involves 36 classes, representing all combinations of object, action, and surface. The results shown in the figure correspond to training with 50% of the labeled training set.

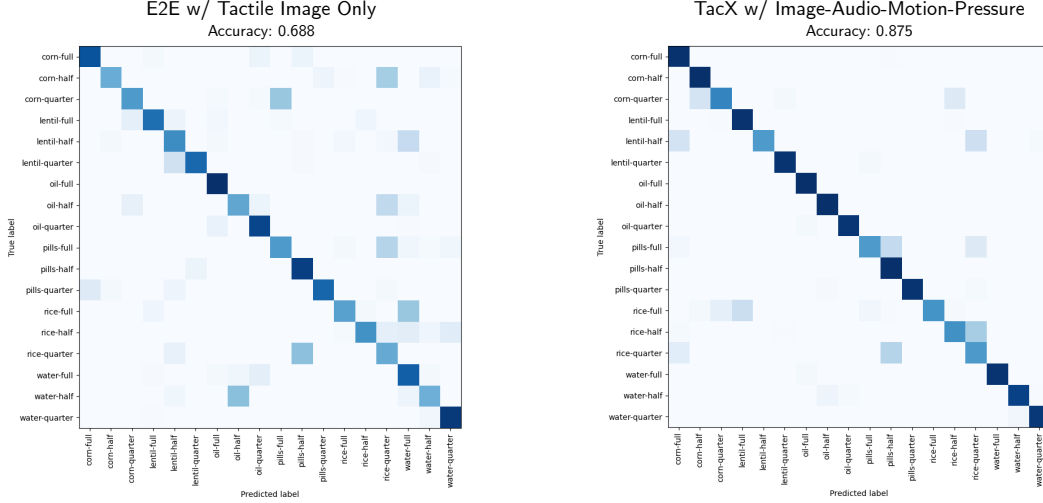


Figure 13: Confusion matrix for material-quantity estimation. We compare an end-to-end classifier trained solely on tactile images with a classifier trained on frozen TacX representations, under a 33% training data budget.

The TacX-based classifier shows stronger diagonal alignment, indicating more accurate predictions across the 36 joint object-action-surface classes. In contrast, the E2E model suffers from greater confusion among similar classes, particularly those with overlapping surface or action components (e.g., misclassifying "Tap-Foamwork" as "Tap-Fabric" or "Slide-Plastic"). These results highlight the benefit of multimodal tactile representations: incorporating audio, motion (IMU), and pressure modalities helps disambiguate fine-grained contact dynamics that are challenging to capture with images alone.

Material-Quantity Estimation. This task further evaluates TacX’s ability to comprehend physical properties. Specifically, we focus on distinguishing materials based on their granularity and viscosity (e.g., solids and liquids), as well as estimating mass through coarse volume classification. We train a classifier to predict one of 18 joint classes, each representing a unique combination of material type and quantity level. The classifier is trained either using frozen TacX representations or end-to-end (E2E) from tactile images alone.

Figure 13 shows the confusion matrices for the material-quantity classification task when trained with 33% of labeled data, comparing an end-to-end (E2E) classifier trained solely on tactile images with a classifier trained on frozen TacX representations. The E2E model achieves 68.8% accuracy, while the TacX-based classifier reaches 87.5%, highlighting the benefit of multimodal tactile representations. In the E2E setting, we observe frequent confusion between different fill levels of the same material and between visually similar liquids such as oil and water. In contrast, the TacX-based classifier exhibits strong diagonal alignment, suggesting accurate identification across the 18 material-quantity classes.

C TacX and Policy Learning

C.1 Real-World TacX and Policy Deployment

For real-world deployment of TacX, we use ROS2. We maintain circular buffers of 5 seconds for each tactile modality per Digit 360 fingertip. For synchronization, we use the timestamp of the image modality as the reference, selecting the closest-in-time sample from the other modalities (audio, motion from accelerometer, and pressure). Once inputs are processed, TacX can run inference at 50Hz on a GPU RTX 4090. However, the construction of log-mel spectrograms remains the main computational bottleneck for real-time processing. When processing all four Digit 360 sensors on

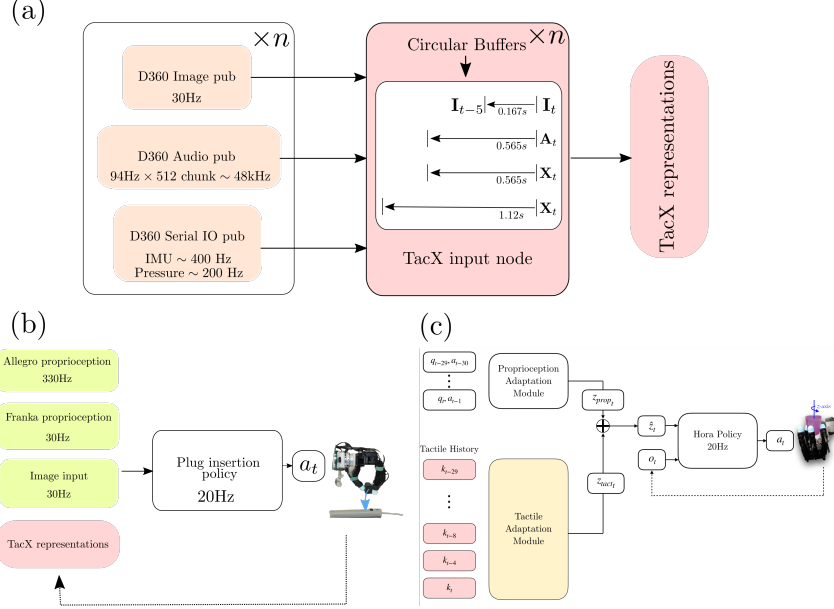


Figure 14: Real-world policy deployment architecture: We use ROS2 middleware for policy deployment, and PyTorch for deep learning modules. In addition to the proprioceptive states of the robot and optional third-person vision modality, downstream policies take as input TacX representations from upto 4 fingertips of the Allegro hand. (a) illustrates how the inputs are constructed for TacX, (b) illustrates policy deployment for the plug insertion policy, and (c) illustrates the policy deployment for the in-hand rotation (Hora) policy.

the Allegro hand, end-to-end inference with TacX runs at approximately 20Hz. Figure 14 shows the deployment pipeline for policy experiments.

C.2 Plug-Insertion via Imitation Learning

Training details. The robot, equipped with an Allegro hand and sensorized with Digit 360 fingertips, is tasked with inserting a pre-grasped plug into the first socket of an extension power strip. In our experimental setup, the socket position remains fixed, while the starting position of the robot arm is randomized within a 3D cuboid of $(5, 5, 2)$ cm around the nominal starting pose.

The model inputs include an embedding of the wrist camera image and TacX representations for thumb, index, and middle finger sensors, processed through an attentive pooling layer [41]. We train a ResNet18 [52] in an end-to-end (E2E) fashion to learn the wrist image embeddings. We append learnable action tokens, which are processed by the transformer and subsequently decoded into a sequence of actions. In our setup, the model predicts a sequence of absolute robot end-effector poses i.e., $\mathbf{a} \triangleq (\mathbf{T}_t, \mathbf{T}_{t+1}, \dots, \mathbf{T}_{t+H})$ with a prediction horizon of $H = 8$. An illustration of the plug insertion architecture is shown in Figure 15.

C.3 In-Hand rotation with sim-to-real tactile adaption

Training details. Hora [50] is a two-stage policy that rotates objects along the z -axis. The first stage trains the policy using privileged information, which includes the object’s state or pose, local shape, mass, friction, and other physical properties that can be perceived by the fingertips. The second stage trains an adaptation module to approximate the latent space of the privileged information from the discrepancy between observed proprioception history and commanded actions, which implicitly informs about contact.

Although the approximation of the privileged vector from proprioceptions transfers to the real setup, it operates with incomplete information about the object’s state. With multisensory touch sensing at the fingertip level, privileged information such as changes in object pose, slip, and friction are

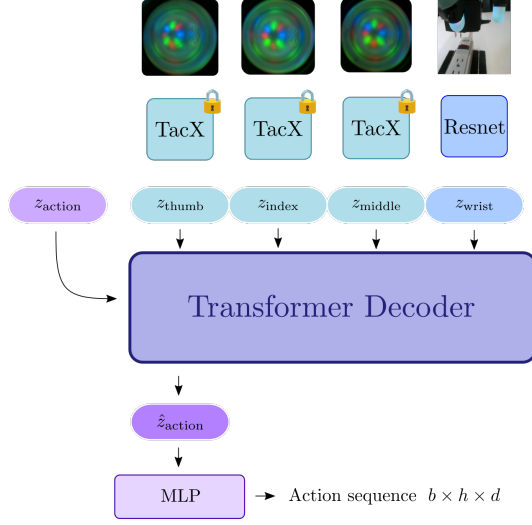


Figure 15: Architecture overview of the plug insertion policy. A transformer decoder is trained to generate action sequences based on TacX representations from three fingertips, a wrist camera image capturing the current robot state, and a learnable latent action code. All embeddings are concatenated and processed by a lightweight MLP to decode the next end-effector action.

616 now accessible in real-world scenarios, albeit not directly. We can leverage TacX representations to
 617 fine-tune the real-world approximation of the privileged information embedding. The goal is to do
 618 *tactile adaptation* on top of the baseline policy to enhance stability during object rotation.

619 We pass to the tactile adaptation module frozen TacX representations for each of the four fingers
 620 in the Allegro-hand, with a temporal stride of 0.19s, equating to 8 touch representations per finger
 621 over a 1.5s window, which matches the proprioception state history consumed by the baseline Hora.
 622 Features for each finger are pooled using attentive pooling to create a global representation, which is
 623 then concatenated along the temporal dimension, resulting in a $(t \times n) \times 768$ input embeddings. The
 624 tactile adaptation model to be trained is a shallow MLP followed by the zero-convolution layer.

625 Our dataset consists of successful rollouts of the Hora policy, where the object keeps rotating without
 626 touching the palm for at least 30 seconds. The data is serialized into the lerobot dataset format [53],
 627 sampled at a control frequency of 20Hz. For training the tactile adaptation module, our objective is
 628 to minimize the L2 loss between the real-world hand joint angles and the target action given by the
 629 frozen Hora policy under the tactile-informed privileged embedding.