FEEL: FINE-GRAINED EMBODIED REPRESENTATION LEARNING FOR ROBOTIC MANIPULATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Multisensory feedback enables humans to perform daily tasks. When opening a soda bottle, vision locates the cap, tactile feedback reveals slip, force feedback indicates changes in resistance, and audio signals the release of gas. Learning comparable representations for robots requires synchronized sensory observations of the same manipulation event. We therefore aim to build fine-grained embodied representations that allow robots to understand the ongoing action and judge its interaction state from multiple senses. This leads to three questions: *how can robots acquire such multisensory experience*, *how can it be organized into manipulation events*, and *how can representations be learned from it*? Accordingly, we develop our framework along three corresponding aspects: **data collection**, **dataset construction**, and **multimodal representation learning**. For data collection, we develop FEELCAP, an egocentric multisensory capture system that synchronously records RGB, depth, tactile, audio, force/torque, and IMU signals. Using FEELCAP, we construct FEELING, an egocentric multisensory dataset of real-world manipulation events. We divide each trajectory into fine-grained interaction segments. Each segment is annotated with the action and manipulated objects, together with their visual and physical attributes. We propose FEEL, trained with three objectives: **semantic-space alignment** associates sensory representations with the corresponding visual–geometry, physical, and event semantics; **intra-modal masked reconstruction** retains modality-specific sensory information; and **cross-modal prediction** learns the relationships among different sensory signals within the same manipulation event. We evaluate FEEL on the FEELING Benchmark using sensor–text retrieval and modality generation. We further assess the learned representations on real-world robot manipulation tasks. Experimental results show that FEEL effectively captures relationships across sensory modalities and improves policy performance on real-world tasks.

1 INTRODUCTION

Human perception is inherently multisensory. Different senses provide complementary information during physical interaction. Vision captures changes in object state and motion, while depth reveals spatial relationships. Once contact occurs, touch detects slip and deformation, force reflects changes in interaction resistance, and audio provides cues about impacts, friction, and state changes. Manipulation depends not only on what is visually observable, but also on what happens during physical contact. Robotic perception therefore requires integrating complementary multisensory information.

Different sensory signals encode the same manipulation differently. A common representation space is therefore needed to associate observations across modalities. Contrastive learning establishes semantic correspondences between vision and language (Radford et al., 2021), with subsequent work extending alignment to broader modalities (Girdhar et al., 2023; Zhu et al., 2024; Lei et al., 2024). Recent methods explicitly distinguish shared and modality-specific representations (Qian et al., 2026; Chen et al., 2026). Manipulation, however, poses a distinct challenge: modalities contribute differently within the same event. Learning their interactions while preserving modality-specific information requires rich, synchronized multisensory data, which remain limited. Our central question is therefore: **how can we establish cross-modal correspondences while preserving the sensory details of manipulation events?**

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Addressing this question first requires recording multiple sensory signals during manipulation. Large egocentric datasets cover diverse activities but remain largely vision-centered (Hoque et al., 2026; Punamiya et al., 2026). Handheld interfaces facilitate the collection of human demonstrations (Chi et al., 2024; Zhaxizhuoma et al., 2025). Recent systems incorporate tactile, force, and other sensory measurements (Clarke et al., 2025; Xu et al., 2025; Cheng et al., 2026; Choi et al., 2026). However, few systems synchronously record rich multimodal signals during the same manipulation.

To address this gap, we develop FEELCAP, an egocentric system that synchronously records RGB, depth, tactile, audio, force/torque, IMU, and gripper-state signals. Using FEELCAP, we collect FEELING, a dataset of synchronized multisensory manipulation trajectories. Each trajectory is segmented into action phases with separate descriptions. For each phase, we construct three descriptions of the action and involved objects: **visual–geometry** describes object appearance, size, shape, and spatial relations; **physical** describes material, hardness, roughness, and texture; and **event** retains only the action and objects.

Built on FEELING, FEEL learns multisensory representations in three spaces defined by the information captured by different modalities. The **visual–geometry** space uses RGB and depth to capture appearance and geometry. The **physical** space uses tactile, force/torque, and audio to capture contact and physical responses. The **event** space captures the action and object shared across modalities. We then perform **semantic-space alignment**, integrating each space's sensory modalities into a unified representation aligned with its text description. Alignment captures manipulation semantics, while generation focuses on fine-grained sensory variation. **Intra-modal masked reconstruction** reconstructs missing content from visible signals of the same modality, preserving fine-grained intra-modal variations. **Cross-modal generation** predicts a fully hidden modality from the others, capturing fine-grained correspondences between sensory signals during the same manipulation.

We evaluate FEEL on **cross-modal retrieval, sensory generation, and real-robot manipulation**. Under the Exact+Weak criterion, FEEL achieves the highest text-to-sensor retrieval recall across all four evaluated modalities. Downstream policies achieve higher success rates on real-robot tasks when using multimodal features extracted by FEEL.

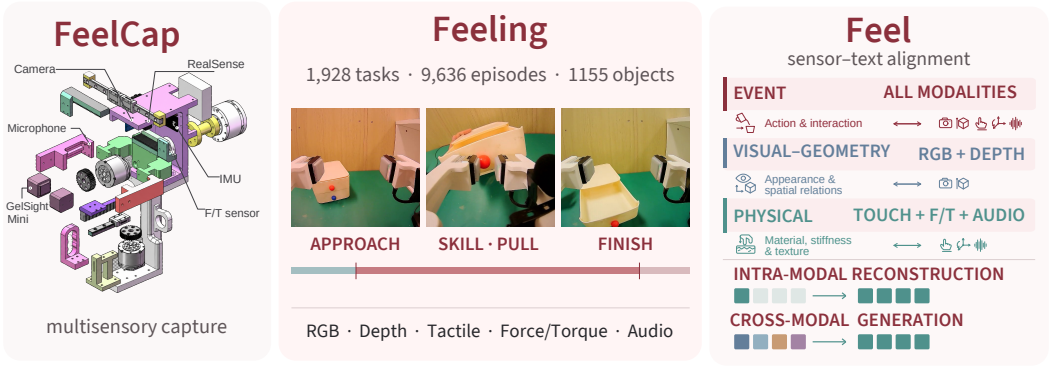


Figure 1: **Overall framework. Left:** FEELCAP enables synchronized multisensory data capture. **Middle:** FEELING provides large-scale aligned multimodal interaction data. **Right:** FEEL learns sensor–text representations through semantic-space alignment, intra-modal reconstruction, and cross-modal generation.

2 RELATED WORK

Egocentric Multisensory Data Collection. Large-scale egocentric datasets capture diverse daily activities and object manipulation from a first-person view (Grauman et al., 2022; Sener et al., 2022; Wang et al., 2023; Hoque et al., 2026; Punamiya et al., 2026), but most are still dominated by visual observations. Recent datasets add touch and audio (Yang et al., 2022; Gao et al., 2022), while handheld interfaces make large-scale collection more practical (Chi et al., 2024; Zhaxizhuoma et al., 2025; Clarke et al., 2025; Xu et al., 2025). However, richer sensing alone does not indicate which local sensory changes correspond to specific actions or object properties. FEELING therefore combines synchronized multisensory trajectories with fine-grained descriptions of short action segments and manipulated objects.

Multimodal Alignment. Contrastive learning connects heterogeneous modalities through shared semantics, first for vision and language (Radford et al., 2021) and later across broader sensory inputs (Girdhar et al., 2023; Zhu et al., 2024; Lei et al., 2024). However, aligned modalities are not necessarily equivalent, and modality gaps can remain after contrastive training (Liang et al., 2022; Ramasinghe et al., 2024; Yaras et al., 2024). Recent methods therefore separate shared and modality-specific information (Chen et al., 2026; Qian et al., 2026). This is important for manipulation, where vision, touch, force, and audio may describe the same event while capturing different aspects of it. FEEL organizes supervision into **visual–geometry**, **physical**, and **event** spaces to preserve this complementarity.

Multisensory Learning for Manipulation. Multisensory manipulation methods combine vision with touch, audio, or force to improve contact-rich perception and control (Li et al., 2022; Chen et al., 2022; Huang et al., 2024; Liu et al., 2024). Recent work further studies multisensory pretraining (Higuera et al., 2025), adaptive fusion (Feng et al., 2024; Li et al., 2025), and sensory prediction or world modeling (Liu et al., 2025; Zheng et al., 2026). Most existing methods focus on a limited set of modalities or fuse them into a single representation. FEEL instead learns from five synchronized sensory streams and combines semantic alignment, intra-modal reconstruction, and current-window cross-modal prediction to capture both shared and modality-specific information.

3 DEVICE AND DATASET

3.1 HARDWARE PLATFORM

Existing handheld data-collection systems improve the efficiency of human demonstrations, but typically capture only a limited subset of the sensory signals involved in physical interaction (Chi et al., 2024; Zhaxizhuoma et al., 2025). **FEELCAP** integrates a dual force feedback gripper with RGB, depth, tactile, force/torque, inertial, audio, and motor-state sensing. Figure 5 presents the exploded view of FeelCap, a handheld platform that integrates sensing and physical interaction into a single assembly. Its modular structure places the operator handle and gripper alongside the visual, tactile, force, inertial, and audio sensing components, with the USB hub and mounting interfaces supporting data collection both by hand and on a robot. FeelCap records the surrounding visual context, gripper motion and motor states, tactile contact, external force/torque, inertial motion, and interaction audio during manipulation. Appendix A provides further details on the hardware components, acquisition pipeline, motor control, and system evaluation.

3.2 DATASET CONSTRUCTION AND ANNOTATION

Within a manipulation episode, successive operations can involve different objects, contact conditions, and sensory responses. Learning from these changes requires data that link synchronized sensory observations to temporally localized actions and the visual and physical properties of the participating objects.

HoloAssist (Wang et al., 2023) provides egocentric recordings with fine-grained action and conversational annotations for studying human assistance during manipulation. At the object level, the dataset collected using X-Capture (Clarke et al., 2025) provides RGBD, tactile, and impact-audio observations sampled at points on objects. To study how sensory responses relate to actions and object properties throughout a manipulation episode, we construct FEELING, pairing synchronized multisensory recordings with fine-grained action descriptions and object attributes across diverse materials and manipulations.

Synchronized multisensory observations. FEELCAP records five synchronized sensory streams—egocentric RGB, depth, tactile images, audio, and force/torque—along with IMU and gripper-state signals. A shared timeline aligns these observations with annotated action segments, enabling joint analysis of visual changes, contact responses, and action descriptions.

Diverse materials and manipulations. The annotated collection comprises 1,928 task groups, 9,636 episodes, and 37,391 action segments. A task group contains repeated executions of a specified manipulation, an episode is one execution, and an action segment is a temporal interval with

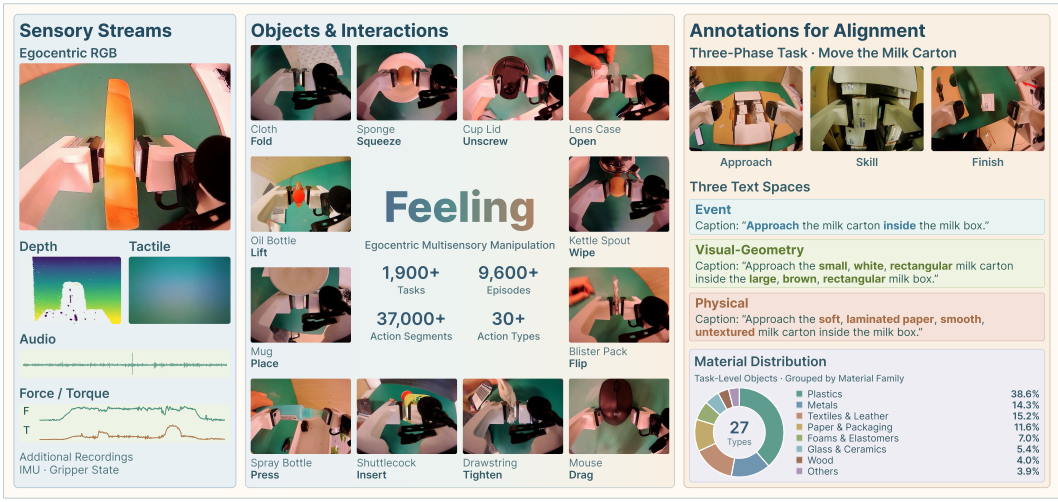


Figure 2: **Overview of FEELING: synchronized multisensory observations, diverse materials and manipulations, and fine-grained annotations. Left:** RGB, depth, tactile, audio, and force/torque streams capture complementary aspects of the same interaction, alongside IMU and gripper-state recordings. **Middle:** Representative objects and manipulations from 1,928 task groups, 9,636 episodes, and 37,391 action segments, spanning more than 30 coarse action categories. **Right:** The milk-carton example links temporal action annotations to three descriptions of the same segment: **event** for actions and participating objects, **visual–geometry** for appearance, geometry, and spatial relations, and **physical** for annotated material and qualitative physical properties. The material chart groups 27 observed material labels from object-annotation records into eight families.

its own phase label and action description. The collection spans more than 30 coarse action categories, including folding cloth, squeezing a sponge, unscrewing a lid, inserting a shuttlecock, and wiping a kettle spout. These interactions cover deformation, rotation, insertion, and surface contact involving rigid and deformable objects. The objects span rigid and deformable materials, including plastics, metals, textiles, paper-based packaging, foams, glass, ceramics, and wood. Their 27 observed material labels are summarized into eight families in Figure 2 (right).

Fine-grained temporal and object annotations. To preserve the changes within each episode, we annotate both the overall manipulation and its constituent action segments. Each segment records temporal boundaries, a phase label, an action description, and the grasped object or object part. The action descriptions distinguish operations that share the same phase label: wiping a kettle spout and subsequently squeezing the sponge form separate `skill` segments. Object annotations further specify material, hardness, surface roughness, and texture presence as qualitative properties. Repeated executions provide observations of corresponding action stages across separate recordings.

From these annotations and segment-associated RGB keyframes, we construct three complementary descriptions of each segment. The **event** (EVT) description specifies the action, participating objects, and necessary relations, providing a semantic reference across modalities. The **visual–geometry** (VG) description retains this action context while adding appearance, relative size, shape, and spatial details for RGB–depth alignment. The **physical** (PHY) description associates the action with the annotated material, hardness, roughness, and texture of the corresponding objects for alignment with tactile, force/torque, and audio observations. All three descriptions refer to the same interaction segment while emphasizing complementary aspects of its content. Appendices B.1 and B.3 provide the annotation protocol and text-construction details.

FEELING Benchmark. To evaluate representations learned from FEELING, we designate its held-out test split as the **FEELING Benchmark**. This subset contains 110 manipulation tasks, 551 episodes, and 5,969 temporal windows, with benchmark objects and their associated trajectories excluded from pre-training. We use this benchmark for cross-modal retrieval (Section 5.2) and

modality generation (Section 5.3). The latter includes both intra-modal reconstruction and cross-modal prediction.

4 METHOD

Understanding a manipulation task requires not only recognizing *what the task is*, but also understanding *how it is performed*. FEEL addresses these two aspects using five synchronized sensory streams and three types of segment description. Semantic-space alignment captures the meaning of the manipulation, while intra-modal reconstruction and cross-modal generation model fine-grained sensory variations within each modality and the relationships across modalities. Figure 3 summarizes the framework. Our training consists of two stages. In Stage 1, we initialize the visual and text branches from the EgoVLA (Yang et al., 2025) video–text pretrained weights and train video, audio, and text representations on EgoVid-5M (Wang et al., 2026). In Stage 2, we initialize from the Stage 1 checkpoint and extend the training to five sensory modalities: RGB, depth, tactile, force/torque, and audio. Further implementation details are provided in Appendix C.

4.1 THREE-SPACE MULTISENSORY REPRESENTATION

Each modality has its own encoder and two projectors. The local projector maps its tokens to VG or PHY according to the grouping above, while the event projector maps them to EVT. The following equations describe one interaction window and one space, so we omit the window and space indices.

For modality m, the encoder E_m and the corresponding projector P_m produce a sequence of 512-dimensional tokens,

$$Z_m = P_m(E_m(x_m)), \tag{1}$$

where x_m denotes the sensory observation. We mean-pool the valid tokens to obtain a modality-level representation $z_m = \mathrm{MeanPool}(Z_m)$. For each space s, let $\mathcal{M}_s$ denote its available modalities. A space-specific gate g_s predicts the contribution of each modality, and the resulting features are fused as

$$\alpha_m = \frac{\exp(g_s(z_m))}{\sum_{k\in\mathcal{M}_s}\exp(g_s(z_k))}, \qquad u_s = \sum_{m\in\mathcal{M}_s}\alpha_m z_m. \tag{2}$$

Here, k indexes the modalities in the same space. The fused representation u_s is used for sensor–text alignment.

4.2 SEMANTIC-SPACE ALIGNMENT

For each semantic space, we align the fused sensory representation with its text description. Positive and negative pairs are constructed from two aspects: the temporal relation and the semantic consistency. For positive pairs, the description of the same interaction window is treated as an *exact positive* with weight 1. Descriptions from different trajectories of the same sub-task are treated as *weak positives* with weight 0.5, because they describe the same manipulation but may differ slightly in the sensory observations.

Descriptions with different meanings in that space serve as negatives. We further select hard negatives from two aspects that are important for real-world tasks: manipulation stages and semantic similarity. **For manipulation stages**, we choose nearby sub-tasks with different actions. Adjacent stages often involve the same objects and similar scene context, but correspond to different actions. This encourages the model to distinguish different steps within the same manipulation process. **For semantic similarity**, we choose samples where at least two of action, object, and object attributes are the same, while the remaining one is different. These samples are difficult because most of their semantic information is shared. Using them helps the model distinguish manipulations that differ only in the action, the manipulated object, or its properties. For each sensory anchor i, let $\mathcal{P}_i$ denote its positive texts and $\mathcal{C}_i$ contain all valid positive and negative candidates. The sensor-to-text loss is

$$\mathcal{L}_{\mathrm{S}\to\mathrm{T}} = -\frac{1}{|\mathcal{B}|}\sum_{i\in\mathcal{B}}\sum_{j\in\mathcal{P}_i}\tilde{w}_{ij}\log\frac{\exp(\cos(u_i,t_j)/\tau)}{\sum_{k\in\mathcal{C}_i}\exp(\cos(u_i,t_k)/\tau)}, \tag{3}$$

where $\mathcal{B}$ denotes the set of valid sensory anchors in the current batch. We assign unnormalized weights $w_{ij} = 1$ to exact positives and $w_{ij} = 0.5$ to weak positives, and normalize them as $\tilde{w}_{ij} = w_{ij}/\sum_{k\in\mathcal{P}_i} w_{ik}$. Here, t_j is the text embedding of candidate j in the corresponding semantic space, and τ is the temperature parameter. The text-to-sensor loss is defined in the same way with the retrieval direction reversed. We apply the loss in each semantic space and average the two directions: $\mathcal{L}_{\mathrm{align}} = \frac{1}{2}\left(\mathcal{L}_{\mathrm{S}\to\mathrm{T}} + \mathcal{L}_{\mathrm{T}\to\mathrm{S}}\right)$.

4.3 Modality Reconstruction

Alignment captures the semantic meaning of manipulation events. To model the fine-grained sensory changes during manipulation, we further introduce two reconstruction tasks within the current interaction window: recovering masked content from the same modality and reconstructing a hidden modality from the other available modalities.

For modality m in window i, let y_i^m denote its ground-truth target signal and D_m its decoder. The same decoder is used for both reconstruction tasks. We use Q^m_{intra} and Q^m_{cross} as the decoder queries for intra-modal reconstruction and cross-modal prediction, respectively.

Intra-modal masked reconstruction. We mask approximately 75% of the valid input of modality m, leaving the remaining 25% visible. No other modalities are provided. Let $\widetilde{Z}_i^{m,\mathrm{EVT}}$ denote the event tokens obtained from the masked input. The prediction is

$$\hat{y}_i^{\mathrm{intra},m} = D_m\left(Q^m_{\mathrm{intra}}, \widetilde{Z}_i^{m,\mathrm{EVT}}\right). \tag{4}$$

Cross-modal reconstruction. We remove the target modality m from the input and reconstruct it using the other available modalities in the same window. Let $\mathcal{A}_i$ denote the available modalities in window i. Their event tokens are concatenated to form $M_i^m = \mathrm{Concat}_{k\in\mathcal{A}_i\setminus\{m\}} Z_i^{k,\mathrm{EVT}}$, where M_i^m contains only source modalities used to predict modality m. The prediction is

$$\hat{y}_i^{\mathrm{cross},m} = D_m(Q^m_{\mathrm{cross}}, M_i^m). \tag{5}$$

Reconstruction losses. For each modality, the loss is computed only on valid supervised elements. Let Ω_r^m denote these elements for modality m under reconstruction condition $r \in \{\mathrm{intra}, \mathrm{cross}\}$. The modality-specific loss is

$$\mathcal{L}_r^m = \frac{1}{|\Omega_r^m|}\sum_{(i,u)\in\Omega_r^m}\rho_m\left(\hat{y}_{i,u}^{r,m}, y_{i,u}^m\right), \tag{6}$$

where u indexes an element of the target signal and ρ_m is the reconstruction loss for modality m. For intra-modal reconstruction, $\Omega^m_{\mathrm{intra}}$ contains only masked and valid elements. For cross-modal reconstruction, $\Omega^m_{\mathrm{cross}}$ contains all valid target elements.

4.4 Joint Objective

The three objectives train the model to match sensory observations with descriptions, recover missing content within a modality, and predict a hidden modality from other sensory streams:

$$\mathcal{L} = \lambda_{\mathrm{align}}\mathcal{L}_{\mathrm{align}} + \lambda_{\mathrm{intra}}\mathcal{L}_{\mathrm{intra}} + \lambda_{\mathrm{cross}}\mathcal{L}_{\mathrm{cross}}. \tag{7}$$

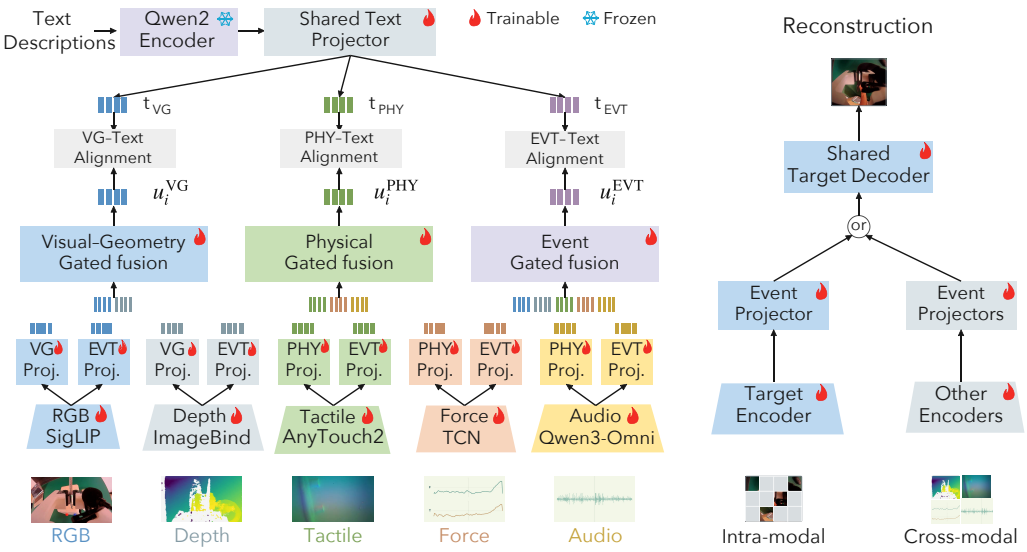


Figure 3: **Overview of FEEL. (a)** Modality-specific encoders and two projection paths produce tokens for the visual–geometry, physical, and event spaces. A gate in each space fuses available modalities for alignment with the corresponding text descriptions. **(b)** Intra-modal masked reconstruction recovers masked content from the visible part of the same modality, while cross-modal reconstruction predicts a hidden target modality from the other available modalities in the same window. Each target modality has its own decoder, shared across the two reconstruction conditions.

5 EXPERIMENTS

5.1 DATASETS AND BASELINES

Manipulation understanding requires recognizing the ongoing interaction and capturing the sensory responses produced during that interaction. We therefore evaluate FEEL from three complementary aspects. Cross-modal retrieval (Sec. 5.2) examines whether sensory observations can be matched with the correct manipulation semantics. Intra-modal reconstruction evaluates the preservation of modality-specific information, while cross-modal prediction evaluates the relationships learned across modalities (Sec. 5.3). We further evaluate FEEL on real-world robotic manipulation to test whether these representations are useful for downstream action prediction.

The FEELING Benchmark contains synchronized RGB, depth, tactile, force/torque, and audio observations from 110 manipulation tasks. All benchmark tasks involve objects that are excluded from pre-training, so the evaluation measures how well the learned representations generalize to unseen objects. We compare FEEL with representative pre-trained models for cross-modal retrieval and modality generation, including **EgoVLP** (Lin et al., 2022), **ImageBind** (Girdhar et al., 2023), **LanguageBind** (Zhu et al., 2024), **UniTouch** (Yang et al., 2024), **Sparsh** (Higuera et al., 2024), **AnyTouch 2** (Feng et al., 2026), and **CLAP** (Elizalde et al., 2023). For a fair comparison, we additionally train or adapt several strong baselines on the same training data before evaluating them on the benchmark. Even under this setting, FEEL achieves strong performance across the evaluated modalities and tasks.

5.2 CROSS-MODAL RETRIEVAL

Given a sensory signal from one modality as a query, we retrieve the corresponding signals from another modality based on representation similarity in Sec. 4.1. For each baseline, we evaluate the supported modality pairs under the same retrieval protocol and report **Recall@1**, **Recall@5**, and **Recall@10**. As shown in Table 1, FEEL achieves consistently strong retrieval performance across

Table 1: **Cross-modal retrieval results on the FEELING Benchmark.** We evaluate sensor-to-text (S→T) and text-to-sensor (T→S) retrieval under two relevance criteria: **Exact positive** and **Exact positive+Weak positive**. All results are reported as Recall@1/5/10 (%), and higher is better (↑). † **denotes methods trained on the same FEELING training set.** The best result within each sensory space is shown in bold.

Space	Method	Exact						Exact+Weak					
		S→T			T→S			S→T			T→S		
		R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
Vision	ImageBind	1.005	3.016	5.529	0.754	3.568	6.483	**7.338**	10.270	13.235	8.226	20.673	26.956
	EgoVLP†	0.821	2.546	4.590	0.972	3.233	5.646	5.964	9.985	14.425	19.886	30.307	38.884
	LanguageBind†	0.737	2.798	4.892	**1.340**	4.255	**7.271**	4.842	**11.560**	**15.061**	9.750	34.847	42.603
	FEEL	**1.072**	**3.049**	**5.713**	1.122	**4.423**	7.120	6.383	8.812	11.643	**22.165**	**36.974**	**44.580**
Depth	ImageBind	0.136	0.577	1.205	0.288	0.984	1.714	1.476	2.257	3.631	1.358	7.755	12.795
	LanguageBind†	0.730	2.223	3.869	0.882	2.545	4.836	6.126	**9.163**	**13.032**	6.516	16.477	24.147
	FEEL	**1.086**	**3.139**	**5.787**	**1.137**	**4.497**	**7.161**	**6.431**	8.994	11.675	**22.298**	**37.095**	**44.527**
Audio	CLAP†	0.135	0.271	0.851	0.097	**0.909**	1.315	0.387	0.696	1.663	1.353	6.381	9.783
	FEEL	**0.155**	**0.599**	**1.121**	**0.271**	0.773	**1.817**	**1.102**	**1.953**	**2.939**	**5.433**	**15.139**	**19.432**
Tactile	AnyTouch2	0.071	0.407	0.921	0.089	0.390	0.762	0.620	1.134	2.126	0.709	5.598	7.848
	Sparsh	0.018	0.053	0.070	0.018	0.088	0.439	0.018	0.105	0.193	0.632	1.703	2.440
	UniTouch	0.035	0.070	0.140	**0.123**	0.228	0.333	0.105	0.210	0.368	1.559	2.960	3.783
	FEEL	**0.158**	**0.525**	**1.086**	0.088	**0.438**	**1.173**	**0.981**	**1.628**	**2.434**	**3.432**	**9.263**	**13.010**

vision, depth, audio, and tactile modalities. In particular, under the Exact+Weak setting, FEEL obtains the best text-to-sensor retrieval results across all four modalities.

5.3 MODALITY GENERATION

We evaluate modality generation under two settings: **intra-modal reconstruction** and **cross-modal prediction**. Both settings use the same trained modality-specific decoder. For intra-modal reconstruction, 75% of the target signal is masked and reconstructed from the remaining 25% of the same modality. For cross-modal prediction, the target modality is fully hidden and predicted from the remaining modalities in the same window, with identical source modalities provided to all compared methods. We report **Mean Absolute Error (MAE)** and **Root Mean Squared Error (RMSE)**, where lower is better. As shown in Table 2, FEEL performs best on vision and audio reconstruction, while tactile reconstruction is slightly behind specialized tactile models. For cross-modal prediction, FEEL achieves the lowest error on four of the six reported target–metric pairs, while LanguageBind performs best on depth MAE and audio RMSE. For tactile intra-modal reconstruction, FEEL performs slightly below the best specialized baseline on a few metrics. One possible reason is that FEEL is trained with several objectives at the same time. It needs to preserve information within each modality, while also learning the relationships between different modalities. It is therefore not optimized only for tactile reconstruction. This may lead to a small drop on some single-task metrics. These results show that FEEL preserves modality-specific information while learning useful cross-modal correspondences.

5.4 ROBOT MANIPULATION TASK

As shown in Figure 4, we evaluate FEEL with two downstream policies, π_0 and EgoVLA, on three real-world tasks: target-weight pouring, tool insertion, and blackboard wiping. For each task, we collect 50 demonstration trajectories for policy training and conduct 10 evaluation trials for each setting. A pouring trial is considered successful when the poured amount falls in the target range of 39–41 g. Tool insertion succeeds only when the tool is fully inserted into the target slot, while blackboard wiping succeeds when the marked area is completely cleaned. We compare the original policy (Baseline) with the same policy trained using the features extracted by FEEL (**+ FEEL**).

Table 2: **Reconstruction results on the FEELING Benchmark.** Lower MAE and RMSE are better (↓). Best results for each target and metric are shown in bold.

Intra-modal Reconstruction				Cross-modal Prediction			
Target	**Method**	**MAE**	**RMSE**	**Target**	**Method**	**MAE**	**RMSE**
Vision	EgoVLP	0.18062	0.22596	Vision	ImageBind	0.17298	0.22092
	FEEL	**0.15179**	**0.19717**		LanguageBind	0.17153	0.21970
					FEEL	**0.17099**	**0.21594**
Tactile	**Sparsh**	**0.06308**	**0.09884**	Depth	ImageBind	0.12849	0.25695
	AnyTouch2	0.06841	0.10356		**LanguageBind**	**0.12715**	0.25804
	FEEL	0.06629	0.11414		FEEL	0.15603	**0.25482**
Audio	CLAP	0.1547	0.2145	Audio	ImageBind	0.16270	0.23239
	FEEL	**0.12876**	**0.19966**		LanguageBind	0.16069	**0.23059**
					FEEL	**0.15636**	0.23103



Figure 4: **Real-world manipulation tasks.**

Performance is reported as task success rate. Adding FEEL features increases the observed success rates of both policies on all three tasks (Table 3).

Table 3: **Real-world manipulation results.** Each policy is trained with 50 demonstration trajectories per task. We compare the original policy (**Baseline**) with the same policy augmented with features extracted by FEEL(**+ FEEL**). Success rate (%) is reported, and higher is better (↑).

Policy	**Setting**	**Pouring**	**Insertion**	**Wiping**	**Avg.**
π_0	Baseline	40.0	30.0	20.0	30.0
	+ FEEL	**50.0**	**50.0**	**60.0**	**53.3**
EgoVLA	Baseline	20.0	10.0	0.0	10.0
	+ FEEL	**40.0**	**30.0**	**20.0**	**30.0**

6 CONCLUSION

In this work, we study multisensory understanding for robotic manipulation from three aspects: data collection, dataset construction, and multimodal representation learning. From the data perspective, FEELCAP enables synchronized egocentric capture of RGB, depth, tactile, audio, force/torque, and IMU signals. Using FEELCAP, we construct FEELING, a multisensory manipulation dataset. From the representation learning perspective, we propose FEEL, which learns shared cross-modal structure while preserving modality-specific information through cross-modal alignment, intra-modal masked reconstruction, and cross-modal generation. We evaluate FEEL on cross-modal retrieval and sensory generation, and further examine its transferability on real-robot manipulation tasks. Experimental results show that the learned representations improve multisensory perception and interaction understanding and can transfer to downstream robotic manipulation.

AI USE STATEMENT

In this work, we used generative AI tools for language polishing and refinement of the manuscript, as well as for assisting with the generation and refinement of textual descriptions for the FEELING dataset. We did not use generative AI tools to generate experimental results, conduct experiments, or make final scientific conclusions, and the remaining required disclosure items are not applicable to this work. We manually reviewed and verified all AI-assisted content, including the generated dataset text, for correctness, consistency, and alignment with the corresponding sensory observations. We take full responsibility for the final content of this work, including all text, claims, annotations, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We make efforts to ensure the reproducibility of our work by providing detailed descriptions of our hardware design, dataset construction, and experimental setup. The main paper presents the design of the multisensory data acquisition hardware and the construction of the FEELING dataset. The source code and dataset will be released upon acceptance to facilitate further research and reproduction of our results.

REFERENCES

Carlos Campos, Richard Elvira, Juan J Gómez Rodríguez, José MM Montiel, and Juan D Tardós. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE transactions on robotics*, 37(6):1874–1890, 2021. doi: 10.1109/TRO.2021.3075644. URL https://doi.org/10.1109/TRO.2021.3075644.

Yizhou Chen, Andrea Sipos, Mark Van der Merwe, and Nima Fazeli. Visuo-tactile transformers for manipulation. *arXiv preprint arXiv:2210.00121*, 2022. URL https://proceedings.mlr.press/v205/chen23d.html.

Zeyu Chen, Jie Li, and Kai Han. Codebind: Decoupled representation learning for multimodal alignment with unified compositional codebook. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 19712–19738, 2026. doi: 10.18653/v1/2026.findings-acl.987. URL https://aclanthology.org/2026.findings-acl.987/.

Tailai Cheng, Kejia Chen, Lingyun Chen, Liding Zhang, Yue Zhang, Yao Ling, Mahdi Hamad, Zhenshan Bing, Fan Wu, Karan Sharma, et al. Tacumi: A multi-modal universal manipulation interface for contact-rich tasks. *arXiv preprint arXiv:2601.14550*, 2026.

Cheng Chi, Zhenjia Xu, Chuer Pan, Eric Cousineau, Benjamin Burchfiel, Siyuan Feng, Russ Tedrake, and Shuran Song. Universal manipulation interface: In-the-wild robot teaching without in-the-wild robots. *arXiv preprint arXiv:2402.10329*, 2024.

Hojung Choi, Yifan Hou, Chuer Pan, Seongheon Hong, Austin Patel, Xiaomeng Xu, Mark R Cutkosky, and Shuran Song. In-the-wild compliant manipulation with umi-ft. *arXiv preprint arXiv:2601.09988*, 2026.

Samuel Clarke, Suzannah Wistreich, Yanjie Ze, and Jiajun Wu. X-capture: An open-source portable device for multi-sensory learning. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6436–6446. IEEE, 2025. doi: 10.1109/ICCV51701.2025.00607. URL https://doi.org/10.1109/ICCV51701.2025.00607.

Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. Clap learning audio concepts from natural language supervision. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2023.

Ruoxuan Feng, Di Hu, Wenke Ma, and Xuelong Li. Play to the score: Stage-guided dynamic multi-sensory fusion for robotic manipulation. *arXiv preprint arXiv:2408.01366*, 2024. URL https://proceedings.mlr.press/v270/feng25a.html.

Ruoxuan Feng, Yuxuan Zhou, Siyu Mei, Dongzhan Zhou, Pengwei Wang, Shaowei Cui, Bin Fang, Guocai Yao, and Di Hu. Anytouch 2: General optical tactile representation learning for dynamic tactile perception. *arXiv preprint arXiv:2602.09617*, 2026.

Ruohan Gao, Zilin Si, Yen-Yu Chang, Samuel Clarke, Jeannette Bohg, Li Fei-Fei, Wenzhen Yuan, and Jiajun Wu. Objectfolder 2.0: A multisensory object dataset for sim2real transfer. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10588–10598. IEEE, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Gao_ObjectFolder_2.0_A_Multisensory_Object_Dataset_for_Sim2Real_Transfer_CVPR_2022_paper.html.

Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind one embedding space to bind them all. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15180–15190. IEEE, 2023. doi: 10.1109/CVPR52729.2023.01457. URL https://openaccess.thecvf.com/content/CVPR2023/html/Girdhar_ImageBind_One_Embedding_Space_To_Bind_Them_All_CVPR_2023_paper.html.

Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 18995–19012, 2022.

Carolina Higuera, Akash Sharma, Chaithanya Krishna Bodduluri, Taosha Fan, Patrick Lancaster, Mrinal Kalakrishnan, Michael Kaess, Byron Boots, Mike Lambeta, Tingfan Wu, et al. Sparsh: Self-supervised touch representations for vision-based tactile sensing. *arXiv preprint arXiv:2410.24090*, 2024.

Carolina Higuera, Akash Sharma, Taosha Fan, Chaithanya Krishna Bodduluri, Byron Boots, Michael Kaess, Mike Lambeta, Tingfan Wu, Zixi Liu, Francois Robert Hogan, et al. Tactile beyond pixels: Multisensory touch representations for robot manipulation. In *Conference on Robot Learning*, pp. 105–123. PMLR, 2025.

Ryan Hoque, Peide Huang, David Yoon, Jian Zhang, et al. Egodex: Learning dexterous manipulation from large-scale egocentric video. In *International Conference on Learning Representations*, volume 2026, pp. 4218–4237, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/07fcc6e2b89439d3ee5ab60939aaa6a0-Abstract-Conference.html.

Binghao Huang, Yixuan Wang, Xinyi Yang, Yiyue Luo, and Yunzhu Li. 3d-vitac: Learning fine-grained manipulation with visuo-tactile sensing. *arXiv preprint arXiv:2410.24091*, 2024.

Weixian Lei, Yixiao Ge, Kun Yi, Jianfeng Zhang, Difei Gao, Dylan Sun, Yuying Ge, Ying Shan, and Mike Zheng Shou. Vit-lens: Towards omni-modal representations. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26637–26647. IEEE, 2024. doi: 10.1109/CVPR52733.2024.02516. URL https://doi.org/10.1109/CVPR52733.2024.02516.

Hao Li, Yizhi Zhang, Junzhe Zhu, Shaoxiong Wang, Michelle A Lee, Huazhe Xu, Edward Adelson, Li Fei-Fei, Ruohan Gao, and Jiajun Wu. See, hear, and feel: Smart sensory fusion for robotic manipulation. *arXiv preprint arXiv:2212.03858*, 2022. URL https://proceedings.mlr.press/v205/li23c.html.

Jinzhou Li, Tianhao Wu, Jiyao Zhang, Zeyuan Chen, Haotian Jin, Mingdong Wu, Yujun Shen, Yaodong Yang, and Hao Dong. Adaptive visuo-tactile fusion with predictive force attention for dexterous manipulation. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3232–3239. IEEE, 2025.

Victor Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Y Zou. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. *Advances in neural information processing systems*, 35:17612–17625, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/702f4db7543a7432431df588d57bc7c9-Abstract-Conference.html.

Kevin Qinghong Lin, Alex Jinpeng Wang, Mattia Soldan, Michael Wray, Rui Yan, Eric Zhongcong Xu, Difei Gao, Rongcheng Tu, Wenzhe Zhao, Weijie Kong, Chengfei Cai, Hongfa Wang, Dima Damen, Bernard Ghanem, Wei Liu, and Mike Zheng Shou. Egocentric video-language pretraining, 2022. URL https://arxiv.org/abs/2206.01670.

Zeyi Liu, Cheng Chi, Eric Cousineau, Naveen Kuppuswamy, Benjamin Burchfiel, and Shuran Song. Maniwav: Learning robot manipulation from in-the-wild audio-visual data. *arXiv preprint arXiv:2406.19464*, 2024. URL https://proceedings.mlr.press/v270/liu25c.html.

Zhuoyang Liu, Jiaming Liu, Jiadong Xu, Nuowei Han, Chenyang Gu, Hao Chen, Kaichen Zhou, Renrui Zhang, Kai Chin Hsieh, Kun Wu, et al. Mla: A multisensory language-action model for multimodal understanding and forecasting in robotic manipulation. *arXiv preprint arXiv:2509.26642*, 2025.

Ryan Punamiya, Simar Kareer, Zeyi Liu, Josh Citron, Ri-Zhao Qiu, Xiongyi Cai, Alexey Gavryushin, Jiaqi Chen, Davide Liconti, Lawrence Y Zhu, et al. Egoverse: An egocentric human dataset for robot learning from around the world. *arXiv preprint arXiv:2604.07607*, 2026.

Chengxuan Qian, Shuo Xing, Li Li, Yue Zhao, and Zhengzhong Tu. Decalign: Hierarchical cross-modal alignment for decoupled multimodal representation learning. In *International Conference on Learning Representations*, volume 2026, pp. 152802–152824, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/f7f5f501282771c96bb3fedcc96bedfe-Abstract-Conference.html.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Sameera Ramasinghe, Violetta Shevchenko, Gil Avraham, and Ajanthan Thalaiyasingam. Accept the modality gap: An exploration in the hyperbolic space. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 27253–27262. IEEE, 2024. doi: 10.1109/CVPR52733.2024.02574. URL https://openaccess.thecvf.com/content/CVPR2024/html/Ramasinghe_Accept_the_Modality_Gap_An_Exploration_in_the_Hyperbolic_Space_CVPR_2024_paper.html.

Fadime Sener, Dibyadip Chatterjee, Daniel Shelepov, Kun He, Dipika Singhania, Robert Wang, and Angela Yao. Assembly101: A large-scale multi-view video dataset for understanding procedural activities. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21064–21074. IEEE, 2022. doi: 10.1109/CVPR52688.2022.02042. URL https://doi.org/10.1109/CVPR52688.2022.02042.

Xiaofeng Wang, Kang Zhao, Feng Liu, Jiayu Wang, Guosheng Zhao, Xiaoyi Bao, Zheng Zhu, and Yingya Zhang. Egovid-5m: A large-scale video-action dataset for egocentric videos generation. *Advances in Neural Information Processing Systems*, 38, 2026.

Xin Wang, Taein Kwon, Mahdi Rad, Bowen Pan, Ishani Chakraborty, Sean Andrist, Dan Bohus, Ashley Feniello, Bugra Tekin, Felipe Vieira Frujeri, et al. Holoassist: an egocentric human interaction dataset for interactive ai assistants in the real world. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 20213–20224. IEEE, 2023. doi: 10.1109/ICCV51070.2023.01854. URL https://doi.org/10.1109/ICCV51070.2023.01854.

Yue Xu, Litao Wei, Pengyu An, Qingyu Zhang, and Yong-Lu Li. exumi: Extensible robot teaching system with action-aware task-agnostic tactile representation. *arXiv preprint arXiv:2509.14688*, 2025.

Fengyu Yang, Chenyang Ma, Jiacheng Zhang, Jing Zhu, Wenzhen Yuan, and Andrew Owens. Touch and go: Learning from human-collected vision and touch. *arXiv preprint arXiv:2211.12498*, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/354892587fe39b17c2b727af02abff4a-Abstract-Datasets_and_Benchmarks.html.

Fengyu Yang, Chao Feng, Ziyang Chen, Hyoungseob Park, Daniel Wang, Yiming Dou, Ziyao Zeng, Xien Chen, Rit Gangopadhyay, Andrew Owens, et al. Binding touch to everything: Learning unified multimodal tactile representations. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26330–26343. IEEE, 2024.

Ruihan Yang, Qinxi Yu, Yecheng Wu, Rui Yan, Borui Li, An-Chieh Cheng, Xueyan Zou, Yunhao Fang, Hongxu Yin, Sifei Liu, Song Han, Yao Lu, and Xiaolong Wang. Egovla: Learning vision-language-action models from egocentric human videos, 2025. URL https://arxiv.org/abs/2507.12440.

Can Yaras, Siyi Chen, Peng Wang, and Qing Qu. Explaining and mitigating the modality gap in contrastive multimodal learning. *arXiv preprint arXiv:2412.07909*, 2024. URL https://proceedings.mlr.press/v280/yaras25a.html.

Zhaxizhuom Zhaxizhuoma, Kehui Liu, Chuyue Guan, Zhongjie Jia, Ziniu Wu, Xin Liu, Tianyu Wang, Shuai Liang, Pengan Chen, Pingrui Zhang, et al. Fastumi: A scalable and hardware-independent universal manipulation interface with dataset. In *Conference on Robot Learning*, pp. 3069–3093. PMLR, 2025.

Yuhang Zheng, Songen Gu, Weize Li, Yupeng Zheng, Yujie Zang, Shuai Tian, Xiang Li, Ce Hao, Chen Gao, Si Liu, et al. Omnivta: Visuo-tactile world modeling for contact-rich robotic manipulation. *arXiv preprint arXiv:2603.19201*, 2026. URL https://arxiv.org/abs/2603.19201.

Bin Zhu, Bin Lin, Munan Ning, Yang Yan, Jiaxi Cui, WANG HongFa, Yatian Pang, Wenhao Jiang, Junwu Zhang, Zongwei Li, et al. Languagebind: Extending video-language pretraining to n-modality by language-based semantic alignment. In *International Conference on Learning Representations*, volume 2024, pp. 9588–9608, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/2862ccf01e3843c81623b246895bcc45-Abstract-Conference.html.

Appendix overview. The appendix provides additional details on three parts of our work. Appendix A describes the hardware, data acquisition, control, and evaluation of FEELCAP. Appendix B presents the construction and annotation of FEELING, including dataset statistics, temporal annotations, and semantic supervision. Appendix C provides the implementation details of FEEL, including Stage-1 pretraining and Stage-2 multisensory training.

A FEELCAP: SYSTEM IMPLEMENTATION AND EVALUATION

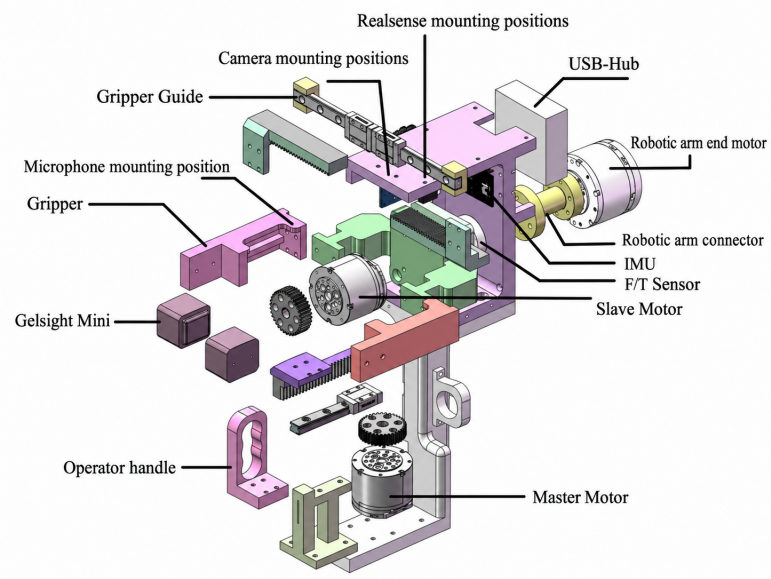


Figure 5: **Mechanical assembly of FEELCAP.** Exploded view of the gripper, operator handle, sensing components, and mounting interfaces.

To integrate multiple sensors into a compact and extensible acquisition platform, FEELCAP adopts a modular hardware architecture. The structure of FEELCAP is shown as Figure 5. Figure 6 shows the three tiers; Table 4 lists sensor roles. The handheld terminal combines the master–slave gripper and seven sensing modalities. The hub distributes power and aggregates USB 3.2/Type-C streams. A Raspberry Pi 5 (8 GB) discovers devices, aligns timestamps, and records sessions before transfer to a workstation for trajectory estimation, annotation, and export.

Table 4: Sensor components and their roles in the acquisition system.

Component	Role in the system
Motor encoders	Position feedback for the master–slave gripper subsystem
Wide-angle RGB camera	Visual context
RealSense D435i	RGB-D observations
Two GelSight Mini sensors	Tactile observations
Six-axis force/torque sensor	External wrench measurement
Independent IMU	Inertial measurements used by offline force compensation
Microphone	Audio stream

A.1 SDK, DATA FLOW, AND MOTOR CONTROL

Collection aligns streams, estimates trajectories with ORB-SLAM3 (Campos et al., 2021), compensates force/torque offline, annotates in a browser, and exports LeRobot data. Deployment sends each sensor's latest frame over Gigabit Ethernet without frame alignment, requiring downstream synchronization. Over CAN FD, the STM32 exchanges position, velocity, and current-derived torque, swaps

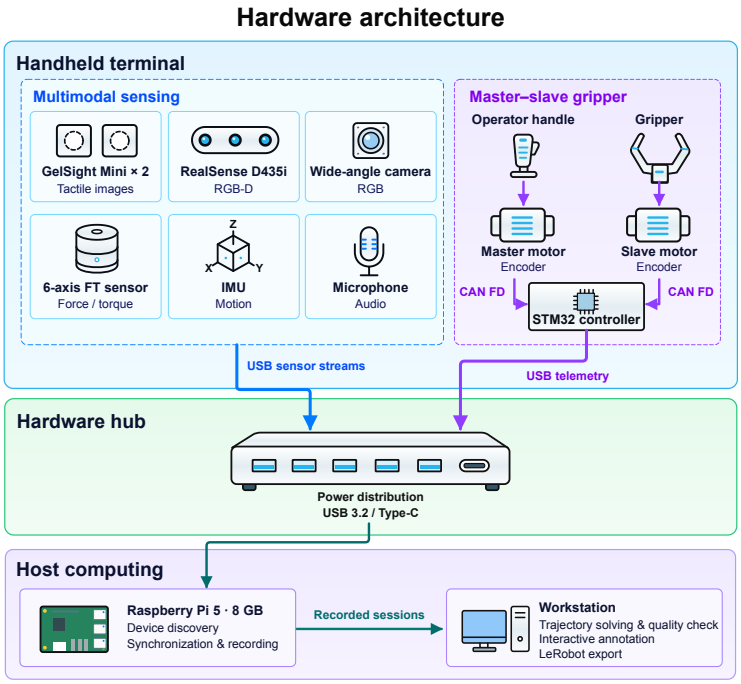


Figure 6: Three-tier path from handheld sensing through recording to workstation processing.

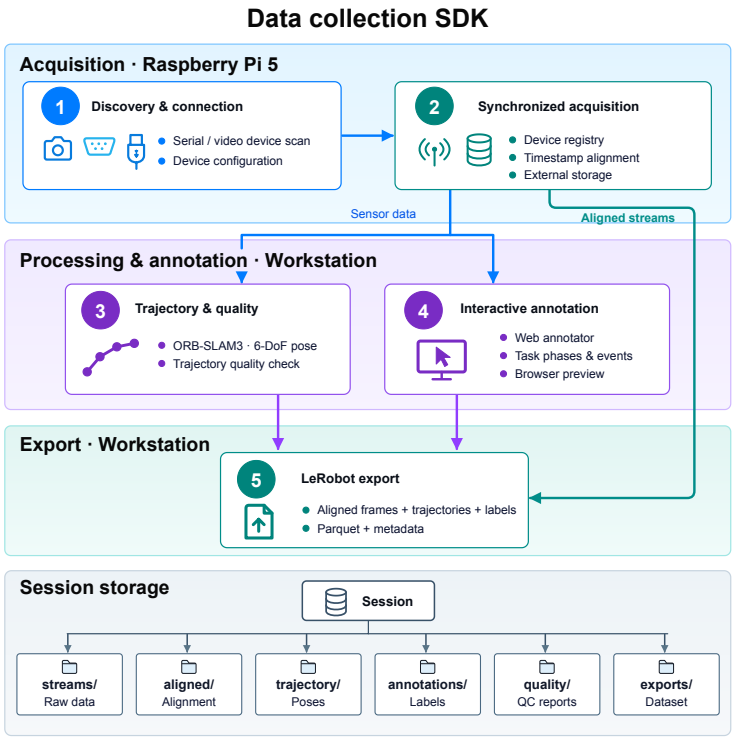


Figure 7: SDK acquisition, processing, and export.

master/slave targets at 1 kHz, and returns MIT commands; USB carries 100 Hz motor-state telemetry outside this loop. Motor torque estimates differ from six-axis measurements.

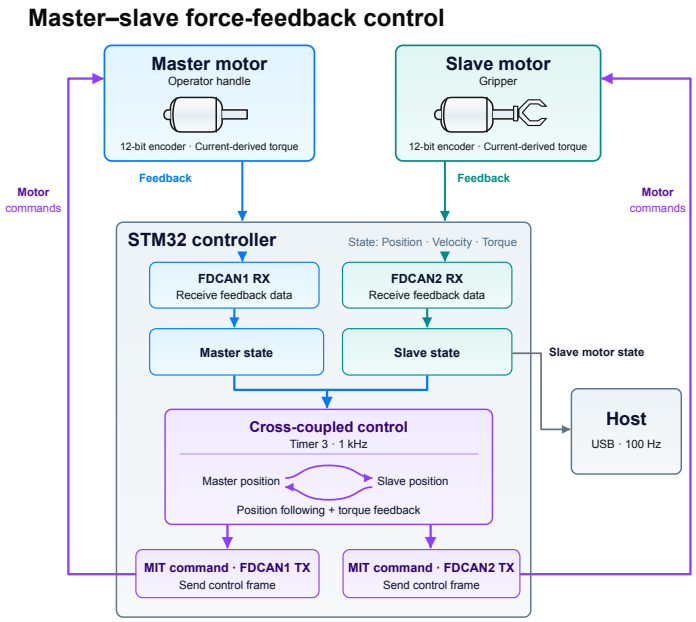


Figure 8: 1 kHz bilateral loop with separate 100 Hz USB telemetry.

A.2 Gravity Compensation and Evaluation

The offline IMU-assisted algorithm estimates posture-dependent gravitational force from a rigidly attached IMU and removes a constant session bias. Accelerations must first be expressed in the force/torque sensor frame; γ handles the remaining sign convention, not an arbitrary rotation. The mass m is the mass supported by the sensor. The initial N_{bias} samples must be static and externally unloaded, and motion-induced acceleration must be small relative to gravity.

The standalone script aligns asynchronous force/torque and IMU streams. Reported offline processing throughput exceeds 1000 samples/s; this is software throughput, not sensor bandwidth.

Algorithm 1 Offline IMU-assisted gravity compensation

Input: raw six-axis force/torque stream F_{raw}, IMU acceleration a_{imu}, mass m, gravity sign $\gamma \in \{-1, +1\}$, timestamps $T_{\text{FT}}, T_{\text{IMU}}$, bias length N_{bias}
Output: compensated stream F_{comp}
Crop both streams to their common time interval and interpolate a_{imu} at each force/torque timestamp
for each retained sample t **do**
    $F_{g,t} \leftarrow \gamma m a_{\text{imu},t}$
end for
Estimate constant force bias $B_f \leftarrow \text{mean}_{t \le N_{\text{bias}}}(F^f_{\text{raw},t} - F_{g,t})$
Estimate constant torque bias $B_\tau \leftarrow \text{mean}_{t \le N_{\text{bias}}}(\tau_{\text{raw},t})$
for each retained sample t **do**
    $F^f_{\text{comp},t} \leftarrow F^f_{\text{raw},t} - B_f - F_{g,t}$
    $\tau_{\text{comp},t} \leftarrow \tau_{\text{raw},t} - B_\tau$
end for
return F_{comp}

On a 12 s trajectory with changing posture and no external load, the mean of the three per-axis force RMS errors decreased from 2.535 to 0.332 N. The relative reduction is $100(1 - 0.332/2.535) = 86.9\%$, rounded to 87%.

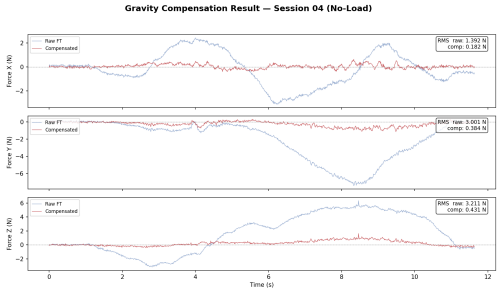


Figure 9: Force before and after compensation (X, Y, Z).

Axis	Before	After
X	1.392	0.182
Y	3.001	0.384
Z	3.211	0.431
Mean of axes	2.535	0.332

Table 5: Per-axis RMS force error before and after compensation (N).

A.3 Timestamp Alignment Evaluation

Figure 10 presents associated-frame timestamp offsets across the sensor streams. Motor, IMU, and force/torque streams run at 100 Hz; RealSense and RGB at 30 Hz; GelSight at 16–18 Hz.

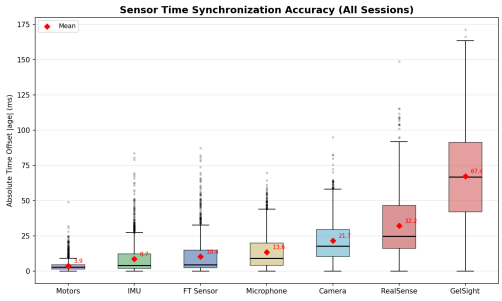


Figure 10: Sensor timestamp-offset distributions.

B Dataset

B.1 Dataset Statistics and Human Annotation

We first annotate episode descriptions, object attributes, and temporally localized actions. Each action segment links a sensory interval to a phase label, an action description, and the grasped object or object part. These human annotations provide the source records for the language and structured semantic annotations described in Appendix B.3. Figure 11 shows the annotation interface, and Appendix B.2 presents examples of the resulting temporal and object annotations.

Annotation coverage. The annotation inventory contains 9,636 distinct episodes organized into 1,928 task groups, with 37,391 temporal segments. We targeted five episodes per task group. The retained count varies because some episodes that failed quality checks were removed without replacement, whereas additional episodes collected beyond the target were retained when they passed these checks. Consequently, 1,910 task groups contain five episodes, 11 contain four, and seven contain six. These counts describe the annotated collection before experiment-specific filtering. Repeated file references to the same recording are counted once. Task groups are retained separately regardless of whether they share a title; the group count does not denote the number of distinct semantic task types.

Interaction and material diversity. Example interactions shown in Figure 2 include folding cloth, squeezing a sponge, unscrewing a lid, inserting a shuttlecock, and wiping a kettle spout. The objects span rigid and deformable materials, including plastics, metals, textiles, paper-based packaging, foams, glass, ceramics, and wood. Repeated executions support comparisons at corresponding action stages. Appendix B.2 provides additional task examples and their temporal progression.

For the material statistics in Figure 2, an object entry denotes an object-annotation record associated with a task group. The summary includes 1,155 such records with 27 observed material labels, grouped into eight material families. This count refers to annotation records and does not establish the number of distinct physical object instances.

Annotators and workflow. Annotations were produced by seven annotators: five recruited undergraduate students who received brief training, and two professional annotators with prior annotation experience. Each episode was initially annotated by one person without pre-populated labels. Annotators followed a common set of phase-segmentation guidelines. The annotation work was followed by a separate review stage, with tasks divided among three reviewers. Task-based annotation assignments were compensated under a batch-level quality review and revision policy.

Annotation interface. Our web-based annotation tool combines a multimodal viewer and shared timeline with an annotation sidebar. Figure 11 shows the full interface followed by detailed views of the sidebar. Playback controls and frame-by-frame navigation support recording inspection; the sidebar supports episode descriptions and quality flags, object attributes, and temporal segments with editable boundaries, phase labels, and descriptions.

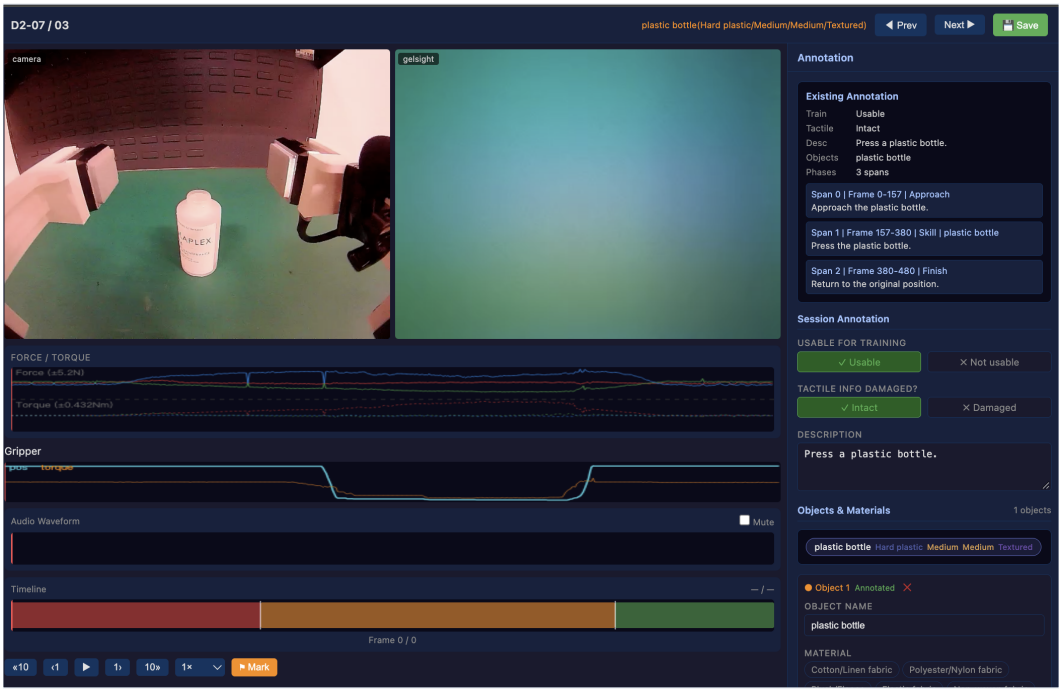
Episode descriptions and object attributes. The episode-level annotation schema includes a description of the overall manipulation and a list of the objects specified by the task, including auxiliary props. Each object is characterized by its name, material, hardness, surface roughness, and texture presence. The interface provides material categories covering everyday materials such as plastics, fabrics, metals, wood, glass, and ceramics, and allows custom entries. Hardness is represented by *soft*, *medium*, or *hard*, corresponding to qualitatively different degrees of deformability. Roughness is represented by *smooth*, *medium*, or *rough*; texture presence indicates visible surface patterns or relief. Annotators assigned these qualitative attributes using the recorded information, without direct physical access to the objects, and consulted the data-collection staff when uncertain. Material, hardness, roughness, and texture labels are coarse semantic attributes rather than instrument-measured physical properties; in particular, hardness is a qualitative judgment from the recordings, not a calibrated measurement.

Temporal action annotations. Each segment has fields for its start and end frame indices, a phase label, an action description, and the grasped object or object part. The guidelines define six phases: *approach* for movement toward an object before contact; *grasp* for closing the gripper to secure an object; *skill* for the core manipulation; *place* for releasing or putting down an object; *finish* for retreating after task completion; and *useless* for manipulation errors such as failed grasps or accidental drops. Grasping and placing receive separate labels when they constitute standalone task actions. Grasping or releasing a subcomponent as part of an operation, such as opening a lid or operating a zipper, is included in the corresponding *skill* segment. Phase labels describe broad stages of execution, while action descriptions identify the specific operations within those stages. Fine-grained annotation therefore relies on both temporal segmentation and operation-specific descriptions. Distinct operations receive separate segments; for example, wiping a kettle spout and squeezing a sponge over a basin are represented by consecutive *skill* segments.

Temporal boundaries and repeated executions. Boundaries are placed approximately at action transitions; the protocol does not require frame-exact localization. The guidelines specify a sequence of transition markers in frame coordinates, starting at zero and ending at the episode's total frame count. For adjacent segments, the stored end coordinate of the earlier segment equals the start coordinate of the later segment, identifying a shared transition location. The records provide segment-level annotations, without an exclusive per-frame phase assignment at transition frames. Repeated executions of the same task are expected to use consistent segment structures, phase labels, and descriptions, with additional *useless* segments permitted when an execution contains an error. Annotations are stored in structured JSON records, with episode metadata and temporal segments stored separately.

Episode usability. The guidelines mark an episode as unsuitable for training if the performed manipulation clearly differs from the specified task, or if severe occlusion or blur prevents the action sequence from being interpreted. Transient manipulation errors do not by themselves invalidate an

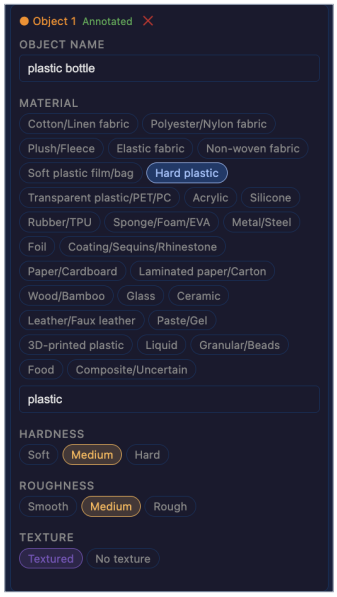
(a) Full annotation interface



(b) Episode annotations



(c) Object attributes



(d) Temporal segments

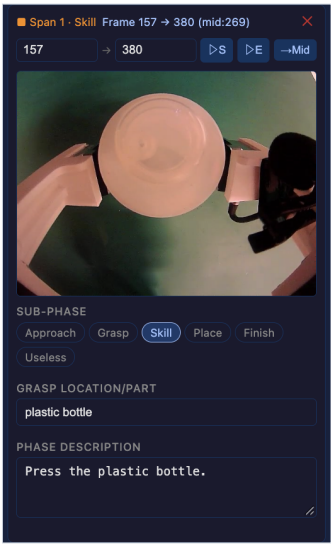


Figure 11: Annotation tool overview and detail views. (a) The full interface, including multimodal recordings, a shared timeline, and the annotation sidebar. (b) Episode-level descriptions and quality flags. (c) Object attributes. (d) Temporal action annotations. Panels (b–d) show enlarged sidebar regions at different scroll positions. Interface labels and example annotations are shown in English for presentation.

episode: the affected intervals are marked as *useless*, while an otherwise usable recording is retained. The interface also provides a separate flag for damaged tactile information.

Review and correction. The review protocol covered every task. Two episodes were randomly selected from each task group for inspection by its assigned reviewer. Review covered phase labels, temporal boundaries, episode and segment descriptions, object attributes, and episode usability. If an issue was identified in any of these fields, the review was extended to all episodes of that task group. Recordings with acquisition problems were recollected or removed, and removed episodes were not always replaced. Annotation errors identified during review were corrected. This procedure combines routine sampling across all tasks with exhaustive review of tasks where problems are detected. Quality control was based on reviewer inspection and correction of singly annotated episodes; repeated independent annotations for agreement estimation were not collected.

B.2 Dataset Examples and Temporal Progression

We present selected recordings to illustrate task content, object annotations, and temporal structure (Table 6). These qualitative examples do not estimate interaction frequencies in the dataset. All use episode 01: S1 tasks are from `umi_data0622`, and the others from `umi_data0807`. Task descriptions and properties are taken from the original episode annotations; phases and boundaries are taken from the segment annotations.

Table 6: **Examples of material diversity and fine-grained action annotations.** Object properties reproduce selected qualitative labels from the original records. Temporal sequences retain every non-deleted segment: A = *approach*, G = *grasp*, S = *skill*, P = *place*, and F = *finish*. Repeated letters denote separate segments. Durations refer to the annotated episode timeline.

Task ID	Task description	Selected object properties	Phases; duration
S1-13	Pick up a bamboo-handled fan and wave it.	Fan: wood; hard; smooth.	A G S G S P 32.23 s
S1-16	Open the lid of a contact-lens case.	Case: plastic; medium hardness; smooth.	A G S F 15.87 s
S1-26	Move an upright replica toast model.	Replica: plastic; hard; rough.	A G S F 18.73 s
S1-30	Press the pump of a spray bottle.	Bottle: plastic; medium hardness and roughness.	A S F 24.43 s
S1-34	Unscrew a thermal-cup lid and remove it.	Cup: plastic; medium hardness; smooth.	A S S F 36.70 s
S3-008	Fold a cloth into a coaster, then place a mug on it.	Cloth: synthetic fabric; soft. Mug: ceramic; hard.	A S A S F 30.23 s
S3-012	Insert a shuttlecock head-first into a drawstring bag and tighten it.	Bag and drawstring: fabric; soft; rough.	A S A S F 30.53 s
S4-044	Wipe a kettle spout with a wet sponge, then squeeze it over a basin.	Sponge: sponge foam; soft. Kettle: metal; hard.	A S S P F 36.07 s

Temporal Examples

Figures 13–16 pair full RGB frames with the original phase timelines. Times are relative to the first stored alignment timestamp, and progress is elapsed time divided by the final annotated transition time. Percentages indicate recording progress, not predicted task completion. Numbered markers locate the screenshots; bar lengths reflect duration, and boundaries indicate approximate action transitions. Frames are retrieved using the stored alignment-to-camera mapping. The four timelines use a 30 Hz alignment grid; when the final boundary follows the last recorded index, its time is the next grid step.

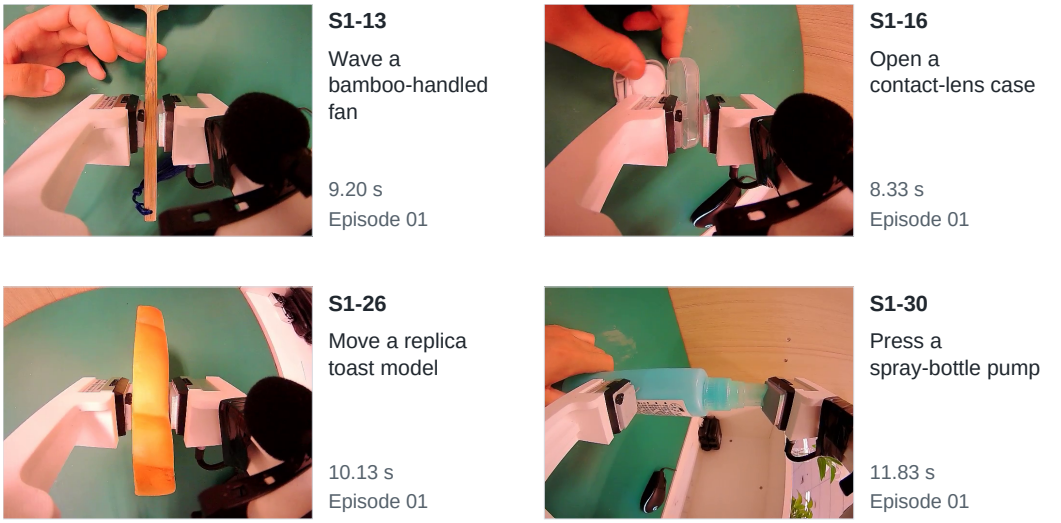


Figure 12: **Visual appearance and annotated physical properties.** Full egocentric RGB frames from four tasks in Table 6. Each frame is sampled within a *skill* segment. The toast-shaped object is annotated as a hard plastic replica, illustrating why visual appearance and annotated physical properties provide different information.

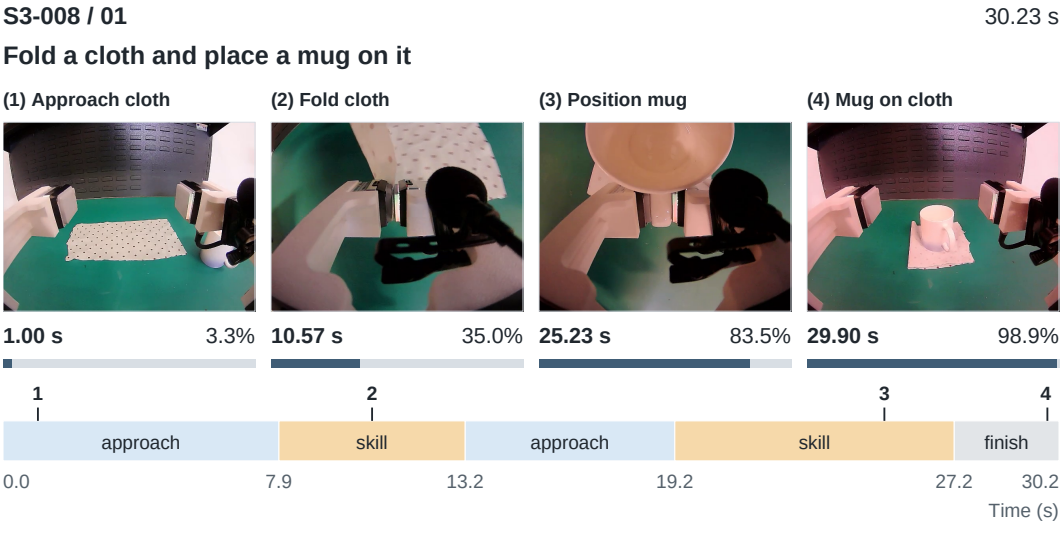


Figure 13: **Folding a cloth and placing a mug (S3-008/01).** The sequence alternates between approaching and manipulating two objects. Folding and mug placement occupy separate *skill* segments, with a second *approach* interval between them. The final screenshot shows the mug resting on the cloth.

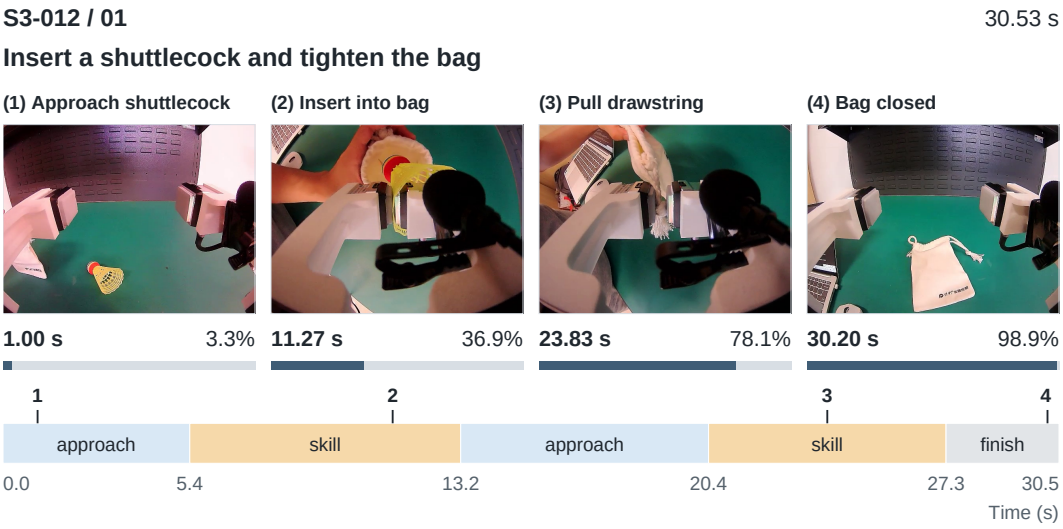


Figure 14: **Inserting an object and closing a bag (S3-012/01).** After inserting the shuttlecock, the manipulator approaches the drawstring before tightening it. This example distinguishes the two object-part interactions within a single task.

Consecutive Operations Within the Same Phase

The examples below contain consecutive *skill* segments with different action descriptions. Their phase labels alone would hide the transition between operations. The timelines therefore retain the original segment boundaries, while the screenshots and action summaries indicate what changes within the manipulation.

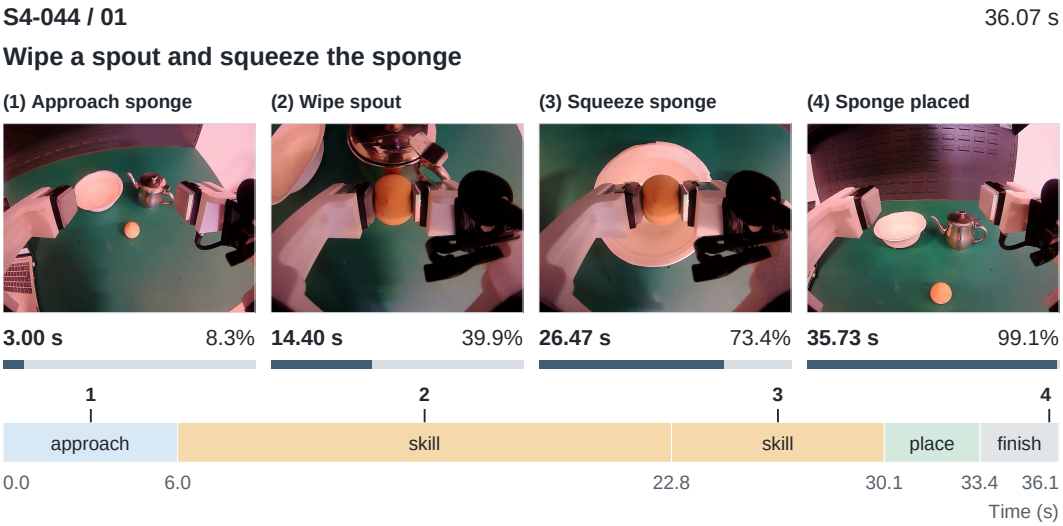


Figure 15: **Wiping and squeezing (S4-044/01).** The same sponge is used first to wipe the kettle spout and then to squeeze above the basin. Both operations are labeled *skill*; the transition at 22.8 s separates their descriptions. A distinct *place* segment precedes the final retreat. The last screenshot is sampled during *finish*, after the sponge has been put down.

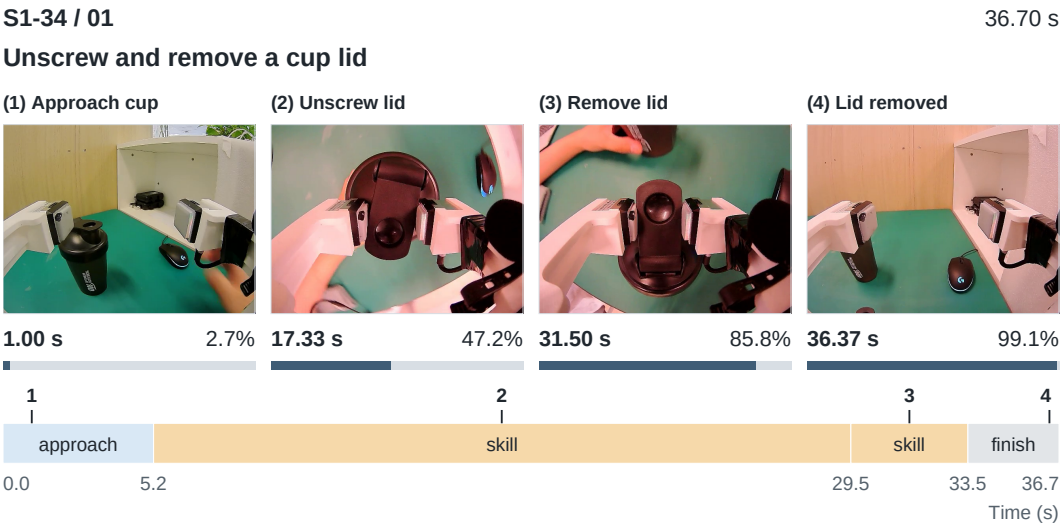


Figure 16: **Unscrewing and removing a lid (S1-34/01).** The longer *skill* interval describes loosening the lid; the following *skill* interval describes removing it. Keeping this boundary preserves the distinction between rotational manipulation and lid removal, even though both share a phase label.

B.3 Caption Generation and Structured Semantic Annotation

Starting from the human task descriptions, object annotations, and temporal phase annotations, we construct language supervision for the visual–geometry, physical, and event spaces. The preprocessing pipeline separates natural-language caption generation from structured semantic extraction. It first produces four caption fields—`visual`, `geometry`, `physical`, and `shared`—and then extracts operations, cross-trajectory correspondences, and entity-grounded evidence from the fixed captions. The resulting text and structures are subsequently assembled into three space-specific supervision records. Throughout this section, a phase-level caption refers to one annotated action-segment occurrence. Different segments may share the same phase label while describing different operations.

Caption generation and grounding. We use Gemini to jointly generate English task-level and phase-level captions for the repeated trajectories of a task; the caption-generation configuration specifies Gemini 3 Flash Preview with a temperature of 0.3. The input contains the original descriptions, object metadata, selected English object names, and one middle RGB keyframe from the available extracted frames of each non-finish phase. Each image is explicitly associated with its trajectory and phase occurrence. Phase annotations anchor the action semantics; the corresponding image supplies directly visible appearance and spatial details. The `finish` phase receives no image, and its captions are derived solely from its original description. Selected object names are kept as bare category noun phrases, with descriptive attributes represented separately. A conservative intrinsic visual profile is shared across repeated trajectories, while grasp locations, orientations, and temporary configurations remain specific to the corresponding observations. Task captions preserve the actions in the task description, and phase captions preserve all actions in their own annotations, including compound actions.

Content of the four caption fields. The `visual` caption describes supported appearance, relative size, intrinsic shape, and visible execution details. The `geometry` caption preserves the action and relevant spatial relations while excluding color, transparency, and physical properties; any relative-size or intrinsic-shape terms must match those in the paired visual caption. Geometry is qualitative supervision for the RGB–depth space, generated from the supplied RGB keyframe and annotations, without inferring metric distances or three-dimensional coordinates. The `physical` caption attaches the annotated material, hardness, roughness, and texture to the corresponding interacting objects. These properties originate from object metadata and are not inferred from image appearance or transferred to other objects or subparts. The `shared` caption provides an impera-

Table 7: **Four caption fields and their mapping to three supervision spaces.** All rows describe the same action segment. The `visual` and `geometry` fields jointly form the visual–geometry target; `physical` forms the physical target after reconciliation with object metadata; and `shared` forms the event target.

Caption field	Illustrative text
`visual-geometry`	Approach the small, white, rectangular milk carton inside the large, brown, rectangular milk box.
`physical`	Approach the soft, smooth, untextured milk carton made of laminated paper inside the milk box.
`shared`	Approach the milk carton inside the milk box.

tive description of the action, objects, and necessary relations, without visual or physical modifiers, and supplies the event-space text. Rule-based validation checks the output schema, trajectory and phase counts, action coverage, selected names, and space-specific attribute constraints. Outputs with detected violations are retained for review and excluded from the validated caption outputs.

Assembly of the three spaces. Let c^v, c^g, c^p, and c^h denote the four phase captions, where h denotes the `shared` field. The three text targets are

$$\begin{aligned} t^{\mathrm{vg}} &= \mathrm{concat}(\text{“Visual: ”}, c^v, \text{“ Geometry: ”}, c^g), \\ t^{\mathrm{phys}} = \widetilde{c}^p, \qquad t^{\mathrm{evt}} &= c^h, \end{aligned} \tag{8}$$

where $\widetilde{c}^p$ is the physical caption after reconciliation with verified object metadata. For visual–geometry semantics, we check agreement of the visual and geometry action sequences, object signatures, and entity correspondences before admitting their combined evidence for semantic relations. The original source captions and spans are retained for attribution. For physical semantics, caption entities are linked to original session objects through unambiguous bindings or reviewed bindings tied to the source annotations and caption. The material, hardness, roughness, and texture values are then taken from the bound JSON records, and the corresponding physical modifiers are rendered into the caption. Character spans are updated after this rewriting, and attribute records retain their source file and JSON pointer. These metadata attributes enrich only the physical space.

Illustrative example. Using the milk-carton approach segment in Figure 2, Table 7 illustrates the four source caption fields. Equation 8 specifies how these fields form the visual–geometry, physical, and event supervision targets.

The visual and geometry captions retain the same action, object relation, relative sizes, and shapes; color appears only in the visual caption. The physical caption associates material, hardness, roughness, and texture with the milk carton. The event target uses the shared caption, the visual–geometry target concatenates the visual and geometry captions with their respective prefixes, and the physical target uses the physical caption after reconciliation with the source object metadata, as defined in Equation 8.

The example also illustrates entity-grounded extraction. The operation is approaching the milk carton, while the milk box serves as the container in the relation “inside.” The attributes “small” and “white” belong to the carton, whereas “large” and “brown” belong to the box. Likewise, the carton’s physical attributes are not assigned to the box. Retaining these entity associations makes the ownership of each attribute explicit.

Ordered operations and cross-trajectory slots. Structured extraction uses a separate text-only API request and prompt. Existing phase captions are grouped by task and caption field, and the model receives only anonymous trajectory letters, text indices, and the fixed captions. The extractor is configured to use Gemini 2.5 Flash with a temperature of 0.0. Its prompt requires operations to be grounded in their own captions, actions to follow a fixed vocabulary, and compound-operation order to be preserved. For each caption, it extracts an ordered list of operations containing an action, an object, and optional attribute fragments. Across repeated trajectories, these operations are aligned to monotonically ordered canonical slots. The JSON output contains `canonical_slots` and

per-caption `assignments`, including operation lists, slot indices, a match type, and uncertainty information. An `exact` assignment associates a caption with one matching slot; `split` represents a compound caption spanning multiple slots; and `merged` represents multiple captions associated with one slot. The schema also records `missing`, `extra`, and `uncertain` correspondences. Validation checks the textual grounding of objects and attributes and the action–object agreement and monotonic order of exact assignments. Caption-specific attributes remain separate evidence. Slots thus accommodate segmentation differences across executions without assuming that a phase label uniquely identifies an operation.

Entity-grounded semantic records. During artifact construction, operation terms are matched back to the source captions and normalized using explicit synonym mappings. Entity records retain an operation index, a role such as target, container, reference, or tool, and a character span in the source caption. Atomic attributes are attached to their owning entities, and spatial relations explicitly identify the subject and reference entities. This ownership distinguishes, for example, the size of a manipulated object from the size of its container. An attribute-state record uses an 11-type schema: color, size, shape, orientation, spatial relation, material, hardness, roughness, texture, temperature, and transparency. Entries record `known`, `unknown`, or `not_applicable` as specified by the extraction and space-filtering rules; inclusion in this schema does not imply that every type is observed. Missing evidence is retained explicitly. Unresolved operation spans, ambiguous entities, unowned attributes, and conflicting values are recorded as validity failures for semantic-relation supervision.

Validity and traceability. Training text selection requires all four phase-caption fields and does not substitute task-level descriptions for missing phase text. Caption trust and eligibility for semantic-relation supervision are tracked separately. In particular, visual–geometry mismatches or unresolved physical-object bindings disable the affected semantic relations; a split assignment can be admitted only after downstream operation, entity, and attribute checks succeed. Each supervision record retains its sample, episode, and phase-instance identity, together with invalidity reasons where applicable. Content hashes link the captions, structured records, source annotations, and preprocessing dependencies, allowing the offline annotations and cached text features to be checked for consistency.

C Implementation Details

C.1 Stage-1 Egocentric Pretraining

Stage 1 is pretrained on EgoVid-5M (Wang et al., 2026), using egocentric video, aligned audio, and event descriptions. Each video sample contains six consecutive frames together with its aligned audio.

The visual branch uses the EgoVLA SigLIP vision encoder and is initialized from the EgoVLA human-video-pretrained checkpoint, while the audio branch uses the Qwen3-Omni audio encoder. Text features are precomputed using the EgoVLA Qwen2 language model and remain frozen during training. The visual and audio encoders are fine-tuned, together with newly initialized projection modules that map the three modalities to a 512-dimensional retrieval space.

Stage 1 jointly optimizes bidirectional video–text, audio–video, and audio–text contrastive objectives. Positive and hard-negative pairs are constructed according to the action and object information in EgoVid. The resulting checkpoint is selected according to the mean validation retrieval performance across the three modality pairs.

The trained vision and audio encoders are subsequently used to initialize the corresponding sensory branches in Stage 2.

C.2 Stage-2 Multisensory Training

Sensory encoders. FEEL uses separate encoders for the five sensory modalities. The vision and audio branches are initialized from Stage 1. Depth uses a pretrained ImageBind-Huge depth encoder, tactile uses AnyTouch2, and the force/torque branch is trained from scratch using causal temporal convolutions. The sensory backbones remain trainable during Stage-2 training.

The three text descriptions are encoded once using the EgoVLA Qwen2 language model. These text features remain frozen, while the shared text projector is trained jointly with FEEL.

Table 8: **Sensory encoders used in FEEL.** Vision and audio inherit the Stage-1 weights, while the remaining modalities use their corresponding pretrained encoders or are trained from scratch.

Modality	Encoder	Initialization
RGB	EgoVLA SigLIP	Stage 1
Depth	ImageBind-Huge	Pretrained
Tactile	AnyTouch2	Pretrained
Force/Torque	Causal temporal encoder	From scratch
Audio	Qwen3-Omni	Stage 1

Token projection and fusion. Each sensory encoder produces an unpooled token sequence. Modality-specific adapters and projectors map these features to 512-dimensional tokens.

Each modality has two projection paths. The local projectors map RGB and depth to the visual–geometry space, and tactile, force/torque, and audio to the physical space. The event projectors map all five modalities to the event space.

For alignment, valid tokens are mean-pooled within each modality. A separate gate is used in each semantic space to combine the available modalities. The visual–geometry gate combines RGB and depth, the physical gate combines tactile, force/torque, and audio, and the event gate combines all available sensory modalities.

The unpooled event tokens are retained for reconstruction.

Reconstruction decoders. Each sensory modality has its own two-layer Transformer decoder with hidden width 512, eight attention heads, and feed-forward width 2048. The same target-specific decoder is shared between intra-modal and cross-modal reconstruction.

Both reconstruction tasks operate within the current interaction window; no future sensory signal is used as a target.

Intra-modal reconstruction. For intra-modal reconstruction, approximately 75% of the valid target input is masked before encoding. Only the remaining 25% of the same modality is visible, and all other sensory modalities are removed.

For RGB, depth, and tactile, masking is performed over spatial 16×16 patches. For audio, complete time columns of the log-Mel spectrogram are masked. For force/torque, complete time steps are masked. The model reconstructs only the masked and valid target elements.

Cross-modal reconstruction. For cross-modal reconstruction, the target modality is removed entirely. The event tokens of the other available modalities in the same interaction window are concatenated and used to predict the target signal. No observation or token from the target modality is provided.

This setting therefore requires the model to use information from the other sensory streams to reconstruct the missing modality.

Reconstruction targets and losses. RGB, depth, and tactile are reconstructed in their image-like representations. Audio is represented as a log-Mel spectrogram with 128 Mel bins, and force/torque is represented as a six-axis temporal sequence.

We use modality-specific reconstruction losses: squared error for RGB and tactile, normalized Smooth L1 loss for depth and force/torque, and absolute error for audio. Missing observations, padded values, and other invalid target elements are excluded from the loss.

Training setup. The Stage-2 split contains 148,844 training windows, 5,625 validation windows, and 5,969 test windows. Training uses AdamW with weight decay 0.01. Pretrained backbone parameters use a learning rate of 10^{-5}, while newly initialized modules use 10^{-4}.

The effective batch size is 2,048. Training uses BF16 mixed precision, gradient clipping with global norm 1.0, and a linear warm-up followed by cosine learning-rate decay. We apply modality dropout with probability 0.3 during training.

Table 9: **Retrieval results of FEEL-Stage1 and FEEL-Stage2.** We report Recall@1/5/10 (%) under Exact+Strong (ES) and Exact+Strong+Weak (ESW). Higher is better. Stage1 uses the historical evaluation protocol; results are presented for reference rather than a controlled comparison.

Space	Method	Exact						Exact+Weak					
		S→T			T→S			S→T			T→S		
		R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
Vision	FEEL-Stage1	0.335	0.720	1.173	0.235	0.804	1.592	1.391	2.077	2.748	1.860	4.624	7.489
	FEEL-Stage2	1.072	3.049	5.713	1.122	4.423	7.120	6.383	8.812	11.643	22.165	36.974	44.580
Audio	FEEL-Stage1	0.039	0.058	0.135	0.019	0.251	0.329	0.077	0.174	0.348	0.522	1.740	2.514
	FEEL-Stage2	0.155	0.599	1.121	0.271	0.773	1.817	1.102	1.953	2.939	5.433	15.139	19.432

Effect of Stage-2 multisensory training. We further study the effect of Stage-2 training by comparing the Stage-1 checkpoint with the full Stage-2 model. As shown in Table 9, Stage-2 training consistently improves cross-modal retrieval for both vision and audio. Table 10 shows similar improvements in reconstruction and cross-modal prediction, especially for depth. These results show that Stage-2 multisensory training further improves both semantic alignment and sensory modeling.

Table 10: **Generation results of FEEL-Stage1 and FEEL-Stage2.** Lower MAE and RMSE are better (↓). A dash denotes an unavailable result. Cross-stage results use different evaluation protocols.

Task / Target	Method	MAE ↓	RMSE ↓
Patch / Vision	FEEL-Stage1	0.15236	0.19832
	FEEL-Stage2	0.15179	0.19717
Horizontal / Vision	FEEL-Stage1	0.17593	0.22160
	FEEL-Stage2	0.17099	0.21594
Horizontal / Depth	FEEL-Stage1	0.20425	0.31241
	FEEL-Stage2	0.15603	0.25482