WHAT TO ATTEND, WHAT TO KEEP: SKILL-CONDITIONED VISUOTACTILE REPRESENTATION WITH PROGRESS-GUIDED EVENT MEMORY

Anonymous authors
Paper under double-blind review

ABSTRACT

Robotic manipulation integrates vision, touch, and language, whose importance shifts across stages: vision guides reaching, while touch, through its evolution over time, decides grasping, alignment, and contact. Yet existing multi-modal manipulation policies typically use fixed temporal contexts and fusion strategies, despite shifts in what each modality contributes across different skills. We study how vision and touch should be combined at the level of primitive skills, asking what each skill needs from each sensor, and propose a language-conditioned representation in which the queried skill selects modality-specific encoders and horizons and attends to a sparse event memory holding the terminal states of the last K completed skills. Evaluated by skill progress estimation on three contact-rich tasks, it reduces slip-detection delay by 87% against fine-tuned SOTA progress models, twist-completion delay by 67.5% against a vision-only ablation, and progress error on a blind search task by 92% through sparse event memory. Gains concentrate exactly where completion is defined by contact or task history. More broadly, our results suggest that observation formation—not only policy architecture—is a central challenge in multi-modal representation. All data, models, and code will be open-sourced upon release.

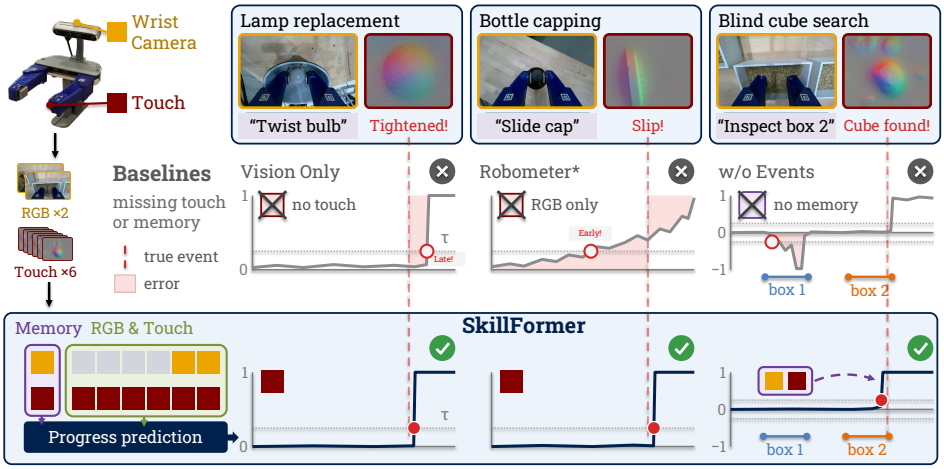


Figure 1: **SKILLFORMER learns what to attend to and what to remember.** Skill-conditioned fusion combines visual and tactile observation windows with progress-guided event memory. Tactile observations reveal tightening and slip events that are difficult to distinguish visually, while memory helps distinguish repeated inspections across boxes. By connecting event detection with retention, SKILLFORMER turns brief sensory observations into lasting context for interpreting subsequent interactions.

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.

1 INTRODUCTION

Robotic manipulation must prioritize different modalities at different stages of a task Lee et al. (2019); Chen et al. (2022). Vision provides spatial context for locating objects and guiding a reach, whereas tactile feedback helps distinguish contact states during grasping, alignment, and insertion Huang et al. (2024). The relevant observations also span different time scales: assessing bulb twisting requires interpreting how tactile readings evolve, while detecting bottle-cap slip hinges on recognizing a brief event. Similar skills can also produce distinct tactile patterns: aligning a bulb with socket threads and aligning a cap with bottle threads need not share the same contact signature. A useful representation must therefore learn which sensory patterns and temporal changes matter for each skill.

Despite these changing requirements, common observation formulations use fixed, short histories. Diffusion Policy uses short observation windows (Chi et al., 2023), while the VLA models condition on a single image and a language instruction (Yu et al., 2026). Applying the same multimodal windows and fusion structure across skills leaves their different sensory needs implicit. Short windows can also omit earlier events: when searching successive containers, the current view may not reveal which ones have already been inspected. Simply extending history is insufficient. Dense visual patch tokens increase attention cost quadratically Vaswani et al. (2017) with sequence length under standard self-attention, while consecutive frames often repeat similar information. This abundance of visual tokens risks obscuring the sparse tactile changes that distinguish contact states.

The challenge is therefore to organize observations, not merely accumulate them. Language-based interaction offers a useful analogy: dialogue turns and tool calls provide explicit events around which memory can be organized. Manipulation instead produces continuous sensory streams, whose meaningful physical transitions must be inferred over time. Language-described primitive skills identify the interaction being queried, while progress supervision connects these descriptions to sensory evidence of completion.

We introduce SKILLFORMER, a language-conditioned visuotactile representation combining modality-specialized encoders, asymmetric short-term horizons, and sparse event-triggered memory. Fixed modality-specific windows capture recent visual and contact dynamics; event memory retains sensory windows at predicted skill completion for use beyond those horizons. A language query conditions joint attention over both sources. At skill transitions, the query and progress objective change together, encouraging phase-specific representations within a shared network. This formulation makes *observation formation*—what to attend to, and what to keep—a central design question alongside policy architecture.

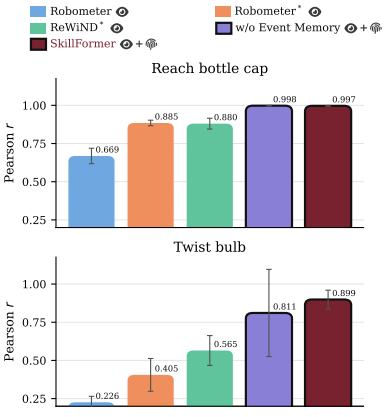


Figure 2: Pearson r measures correlation between predicted and ground-truth progress (higher is better). Greater variability on twisting *w/o event memory* suggests less consistent performance that does not establish online generalization. *: fine-tuned; eye/fingerprint: vision/touch.

Progress estimation as the lens. Skill progress provides structured, interpretable supervision for representing geometric state, contact dynamics, and interaction history without jointly optimizing an action policy. Robometer and ReWiND demonstrate the value of learned progress and reward signals for robot learning (Liang et al., 2026; Zhang et al., 2025). Motivated by these works, we use this objective to probe task-relevant information in latent representations. Queries for active and inactive skills test whether the representation distinguishes completed, ongoing, and not-yet-started skills. Across lamp replacement, bottle capping, and blind object search, we measure how skill-progress prediction changes when tactile input or event memory is removed, observation histories are shortened, or the fusion mechanism is changed. Section 3.4 specifies the event-construction protocol and its annotation access.

Our primary contributions are as follows:

A framework for visuotactile representation learning: We introduce SKILLFORMER, which uses language-described skills to organize perception and memory for contact-rich manipulation.

Skill-conditioned fusion of vision and touch: We introduce a skill-conditioned attention mechanism that integrates visual and tactile observations. Our model reduces overall skill-progress MSE by 37–80% compared with vision alone across the three manipulation tasks.

Sparse memory of past interactions: We use predicted skill completions to form a sparse memory of earlier interactions. Event memory reduces overall skill-progress MSE by 59–92% across all three tasks compared with using recent observations alone.

Three contact-rich manipulation datasets: We collect 429 demonstrations of lamp replacement, bottle capping, and blind cube search, pairing synchronized video and tactile recordings with primitive skill annotations.

2 Background

Multimodal representations for manipulation. Visuotactile learning combines scene-level visual information with local contact observations (Lee et al., 2019; Li et al., 2022). The usefulness of each modality depends on the interaction, motivating a study of its feature space, temporal context, and role in fusion Feng et al. (2025); Higuera et al. (2024). We build on CLIP visual and language features (Radford et al., 2021) and the tactile-specialized FeelAnyForce encoder (Shahidzadeh et al., 2024). A generic DINOv2 encoder (Oquab et al., 2023) provides a control for tactile specialization. The contribution studied here is how these features and their histories are organized around skills, beyond the selection of a pretrained backbone.

Skill conditioning and temporal representations. Language-described skills provide an interface for structured manipulation (Kroemer et al., 2021; Ahn et al., 2022; Shi et al., 2025). We use a skill query to condition a representation of the observed interaction. Short observation windows in policy learning (Chi et al., 2023) and longer sequence context (Chen et al., 2021) offer different temporal information. Our sparse memory retains skill-terminal observations and is evaluated against comparable recent and uniformly sampled histories Zeng et al. (2026); Yang et al. (2026); Shi et al. (2026).

Progress supervision and representation evaluation. VIP and LIV connect learned visual representations to reward and task structure (Ma et al., 2022; 2023); ReWiND and Robometer study learned rewards and their use in robot learning (Zhang et al., 2025; Liang et al., 2026; Chen et al., 2026). We use progress supervision to train and diagnose a multimodal representation, concentrating on contact states and historical dependencies. Comparisons with existing progress models measure how well the representation supports progress estimation. We separately evaluate the representation itself by freezing its features and training simple probes on top, testing whether it captures task-relevant information beyond what is needed by the original progress predictor.

3 Structuring Skill-conditioned Representations with Event Memory

We introduce SkillFormer, a skill-conditioned Transformer that combines recent visual and tactile observations with a sparse memory of earlier skill events (Figure 3). We first define primitive skills, the skill-conditioned representation, and skill progress targets (Section 3.1). We then describe how visual and tactile observations are encoded over modality-specific histories (Section 3.2), how skill-conditioned attention combines current observations with retained events (Section 3.3), and how observation windows are retained at detected skill terminations to form a sparse event memory (Section 3.4). Finally, we describe how skill-progress supervision trains the representation and how predicted progress is used to construct memory during training (Section 3.5).

3.1 Skill-conditioned formulation

Long-horizon manipulation comprises primitive skills with distinct objectives and sensory requirements: reaching brings the gripper to a target, guided predominantly by vision; tightening secures a part, relying on tactile feedback; and alignment establishes the required relative position and orientation, using both modalities.

Skills as the unit of long horizon tasks. Long-horizon manipulation comprises primitive skills with distinct objectives and sensory requirements: reaching brings the gripper to a target, guided predominantly by vision; tightening secures a part, relying on tactile feedback; and alignment establishes the required relative position and orientation, using both modalities. This decomposition structures representation learning around local objectives and enables skill-dependent attention to multi-modal observations. The same observed state may satisfy one skill's objective but not another's. We therefore condition the representation on the queried skill when estimating progress and completion. Associating representations with primitive objectives also provides a basis for reusing skills in different orders and task compositions. To assess whether the representation captures these skill-specific distinctions, we use skill progress as an interpretable readout of intermediate states and completion. Its definition depends on how each interaction unfolds, motivating the following progress types.

Skill progress types. A primitive skill is a goal-directed interaction with a language-described objective and an observable termination condition. We distinguish continuous and discrete skills, with conditional skills extending the discrete formulation to include failed outcomes.

Continuous skills, such as reaching, have observable advancement throughout execution. We assign dense progress targets by linearly normalizing time within the annotated skill interval:

$$p_t^k = \frac{t - t_k^{\text{start}}}{t_k^{\text{end}} - t_k^{\text{start}}}, \qquad t \in [t_k^{\text{start}}, t_k^{\text{end}}].$$

Discrete skills, such as sliding a bottle cap or tightening a bulb, have intermediate observations that do not reliably indicate progress. Completion is instead marked by a terminal event. Progress remains at 0 during execution and rises sharply to 1 over the last few frames preceding the annotated termination.

Conditional skills, such as inspection, are discrete skills that can terminate in success or failure. For these skills, we use three-valued targets: $p_t^k = 0$ before termination, $p_t^k = 1$ at successful termination, and $p_t^k = -1$ at failed termination. The terminal outcome determines the subsequent sequence of skills.

Skill-conditioned state representation. We use a predefined vocabulary of primitive skills and annotate demonstrations across our three tasks as sequences of executed skill instances. We use language to identify the queried primitive skill within a predefined vocabulary, rather than to support open-vocabulary generalization. Each instance ℓ_k has annotated start and end times $(t_k^{\text{start}}, t_k^{\text{end}})$, from which we derive progress targets to supervise representation learning. At timestep t, the robot receives an RGB image and two tactile images, $o_t = (v_t, u_t^L, u_t^R)$. Conditioned on the queried skill ℓ_k, the model combines recent multi-modal observation windows O_t with event memory M_t, which retains observation windows from up to K preceding skill terminations:

$$h_t^k = F_\theta(O_t, M_t, \ell_k) \in \mathbb{R}^d. \tag{1}$$

Here $O_t = (V_t, T_t)$ contains recent RGB and dual-tactile observations, while M_t retains observation windows from earlier skill events. In the blind-search example in Figure 3, the query is *"inspect box 2"*. The green panel shows the current observation windows O_t, and the preceding blue panels illustrate the events retained in M_t. Current and retained observations follow the solid and dashed paths, respectively, through the sensory encoders. Their tokens are then processed together with the skill query to estimate progress for the inspection of box 2.

Task-conditioned models such as ReWiND (Zhang et al., 2025) and Robometer (Liang et al., 2026) estimate progress relative to a task-level instruction. Our formulation makes each primitive's objective explicit during observation fusion, allowing the representation to weight visual and tactile observations according to the queried skill. Skill-level progress supervision also provides a mechanism for identifying terminal observations to retain, connecting what the representation attends to with what it keeps for subsequent interactions.

3.2 Modality-specific features and histories

We encode visual and tactile observations separately using CLIP ViT-B/16 (Radford et al., 2021) and FAF (Shahidzadeh et al., 2024), respectively. For each frame, we extract a single CLS token from the corresponding encoder and project it to $d = 512$ dimensions. The tactile projection includes layer

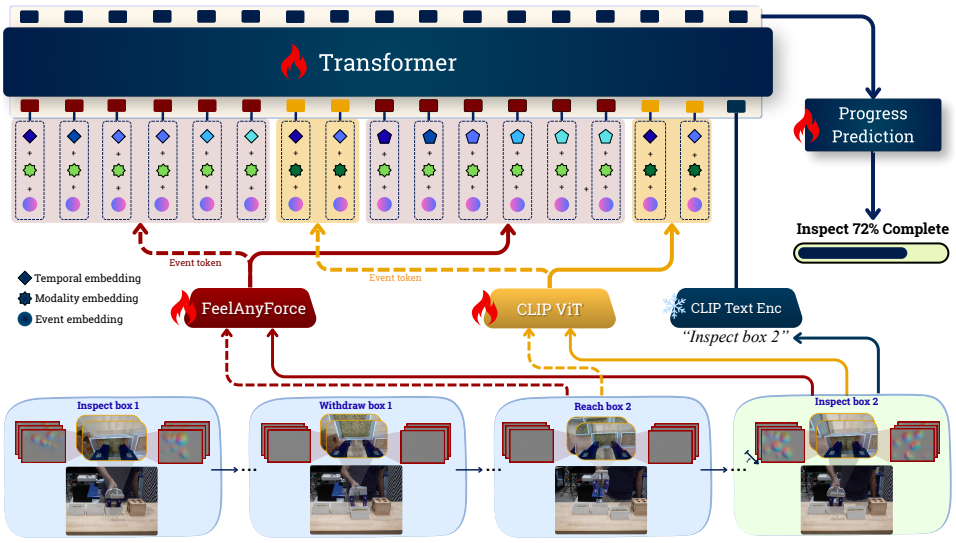


Figure 3: SKILLFORMER on *blind cube search*, queried with *"inspect box 2"*. **What to attend:** the two tactile streams are encoded frame-by-frame by FeelAnyForce and the RGB stream by CLIP ViT (both fine-tuned), each over its own horizon; the skill description is encoded by the frozen CLIP text tower. Every token is the sum of its feature and a temporal, modality, and event embedding, so the transformer knows when a token was observed, from which sensor, and whether it belongs to the current window or to memory. **What to keep:** when a skill completes (here *inspect box 1*, *withdraw box 1*, *Reach box 2*) their encoded observation window is stored as event tokens and attended alongside the current observation. The learned output tokens are mapped to predict queried skill progress.

normalization and is shared between the two tactile sensors. These tokens are passed individually to the fusion network, allowing it to attend to each sensor's features. We use observation windows of two RGB frames and six frames from each tactile sensor, reflecting the different temporal characteristics of vision and tactile sensing Feng et al. (2025).

For modality m, a window contains H_m observations sampled with stride d_m. Its temporal span is $(H_m - 1)d_m/\nu_m$ seconds at sensor rate ν_m. Each observation is encoded as

$$z^m_{t,j} = P_m f_m(o^m_{t-(H_m-1-j)d_m}) + \eta^{\text{time}}_j + \eta^{\text{modality}}_m + \eta^{\text{segment}}_0, \tag{2}$$

where $j = 0, \ldots, H_m - 1$ indexes observations from oldest to newest, f_m is the sensory encoder, and P_m is the feature projection. The learned embeddings encode temporal position, sensor identity. Retained tokens additionally receive event-slot embeddings indicating chronological order. Horizons and encoders are shared across skill queries; the fusion transformer learns how to utilize their tokens for each skill.

3.3 SKILL-CONDITIONED ATTENTION

The transformer in Figure 3 jointly processes the skill query, current observation tokens, and retained event tokens. Self-attention allows interactions across sensors and observation windows, while a learned state token aggregates their information into a representation for the queried skill.

We encode the skill description with the frozen CLIP text encoder and project it to a query token q_k. Given current observation tokens Z_t, encoded event memory $E_\theta(M_t)$, and a learned state token z_{out}, the representation is

$$h^k_t = \text{LN}\big(\text{Transformer}_\theta\big([q_k; Z_t; E_\theta(M_t); z_{\text{out}}]\big)_{\text{out}}\big)\,. \tag{3}$$

Attention is bidirectional within the input sequence, which includes recent observation tokens and tokens from earlier event windows retained in M_t. The state token can therefore combine recent observations with past interactions relevant to the queried skill. No observations after time t are used.

3.4 Long- and Short-Term Memory via Sparse Events

The recent observation window O_t provides short-term context, while event memory M_t retains earlier observations that may remain useful for subsequent skills. When a skill event is detected at frame t, we add the sensor window ending at that frame, O_t^{event}, to memory:

$$M_{t+1} = \text{LastK}\left(M_t \parallel O_t^{\text{event}}\right). \tag{4}$$

Here $\parallel$ appends the new window, and LastK retains at most K of the most recent event windows in chronological order. Memory remains unchanged when no event is detected.

Event detection at time t uses the current observations O_t and existing memory M_t. The newly stored window becomes available at $t+1$, so it cannot affect its own detection. We use $K = 1$ for lamp and bottle and $K = 3$ for blind search. In Figure 3, the three retained events correspond to *inspect box 1*, *withdraw box 1*, and *reach box 2*. Memory preserves the observation windows ending at these events, rather than the full sequence of intervening observations.

We detect events using the progress predictor described in Section 3.5. For the skill instance a_t queried during event construction, a single frame satisfying $|\hat{p}_t^{a_t}| \geq \tau_{a_t}$ is sufficient to add the observation window ending at that frame to memory. We use a threshold of 0.95 for continuous skills and a lower threshold of 0.25 for discrete and conditional skills, whose targets remain zero until near termination and then change sharply. Taking the absolute value allows both successful and failed conditional outcomes to trigger an event.

3.5 Training the Skill Progress Predictor

The progress-prediction block in Figure 3 applies a head g_ψ to the skill-conditioned representation h_t^k, producing $\hat{p}_t^k = g_\psi(h_t^k)$. We train the model by minimizing the squared error between predicted progress and the annotated targets:

$$\mathcal{L}(\theta, \psi) = \mathbb{E}_{(t,k)}[(g_\psi(h_t^k) - p_t^k)^2]. \tag{5}$$

With probability α, we query the active skill; otherwise, we sample uniformly from the inactive skill descriptions. Pairing the same observations with different skill queries encourages the model to estimate **progress for the queried skill**, rather than always for the skill currently being performed. This encourages attention to visual and tactile features **relevant to the queried skill**. Visual and tactile encoders are fine-tuned with the fusion transformer, while the text encoder remains frozen.

Because event selection relies on predicted progress, we first train the model without past event observations, then introduce event memory.

Warmup training without event memory. We first train the model to predict skill progress from the current observation windows. Event slots remain in the input sequence, but their sensory inputs are set to zero. After this initial training, we run the predictor over each training episode and identify event frames using the progress thresholds in Section 3.4. The observation windows ending at these frames are retained for the next stage.

Training with event memory. We continue training the same model using both current observations and the selected event windows. Every R epochs, we run the updated predictor over all training episodes to select a new set of event frames. Between these updates, the selected frames remain fixed, while the model continues to learn from current and retained observations. Progress targets are still derived from the annotations; the model's predictions determine which past observation windows are retained in memory.

4 Contact-rich Datasets and Tasks

We study three contact-rich manipulation tasks with complementary sensing and memory requirements (Figure 9). **Lamp replacement** combines reaching and grasping with thread alignment and tightening. Successive twists look similar, while touch captures increased resistance at completion; we disable the light-on cue to remove this visual shortcut. **Bottle capping** requires detecting a brief slip during repetitive cap sliding to transition to the final twist. Both tasks involve thread alignment, but their distinct contact patterns require object-specific interpretation of tactile observations. **Blind cube search** involves searching among three boxes containing a cube or

distractors (a cylinder and a pentagonal prism), withdrawing after unsuccessful inspections, and inserting the cube into a puzzle box once found. Touch helps identify the shape, while inspection history and outcomes determine progress for box-specific skills. For example, finding the cube in box 1 leaves "reach box 2" unexecuted, so its progress should remain zero. During later insertion, this history also identifies which box contained the cube. Event memory retains context for these distinctions beyond the current observation window. Together, these tasks test short-term contact dynamics and longer-term context retained through sparse events. They provide a setting to evaluate whether sparse event memory preserves useful task history beyond short-term observation windows without retaining a dense sequence of past observations. See the dataset section in the supplementary material for task details, skill definitions, and termination conditions.

Hardware setup. Demonstrations are collected with UMI-Tac, a customized UMI gripper (Chi et al., 2024), equipped with two GelSight Mini tactile sensors at fingertips and an Intel RealSense D455 RGB-D wrist camera (Figure 4). Synchronized RGB and dual-tactile image streams provide visual context and contact observations for all three tasks.



Figure 4: **UMI-Tac.** a handheld gripper with a wrist camera and tactile sensors.

Dataset size and split. The datasets contain 429 episodes in total: 134 for lamp replacement, 151 for bottle capping, and 144 for blind cube search. Approximately 10% of each original dataset is held out at the episode level as a test set.

5 SkillFormer Remembers Task-Relevant Events

In this section, we compare SkillFormer to a number of language-conditioned representation baselines. Across our challenging tasks, SkillFormer effectively prioritizes and remembers the task-relevant events, reducing event detection delays and improving progress estimation. We describe baselines and metrics and expose the core design decisions that contribute to the strong performance relative to other multi-modal or vision-only alternatives that do not employ memory.

Baselines. We compare SkillFormer against two language-conditioned manipulation reward models and four ablations. Robometer is a vision-language reward model built on Qwen3-VL-4B and pretrained on large-scale datasets with both per-frame progress and cross-trajectory preference objectives. We report it zero-shot and after fine-tuning a rank-32 LoRA adapter and its progress heads on our dataset noted as Robometer*. ReWiND employs a transformer over frozen DINOv2 image and MiniLM language features; we train it on our dataset together with Open X-Embodiment O'Neill et al. (2024) data noted as ReWiND*. Both use RGB observations without touch or event memory. . Our ablations isolate individual design choices under the matched data and evaluation protocol: **Vision Only** removes the tactile encoders; **w/o Event** removes the event tokens; **FiLM** replaces our bidirectional self-attention with FiLM conditioning of the visual features on tactile features and task description; and **SkillFormer (h=2)** shortens the tactile horizon from six to two frames. and **w/o FAF** replaces FAF with an ImageNet-pretrained DINOv2. On blind search, we additionally evaluate SkillFormer with a single event slot (K=1) instead of three, and the vision-only ablation is trained with K=3 to match the full model.

Evaluation pipeline. As described in Section 3, we learn a multi-modal representation that detects skill transitions with low delay and uses retained events to interpret subsequent observations. We query every skill throughout each held-out episode and compare predicted progress with ground-truth targets derived from annotated skill boundaries and progress types. Progress accuracy measures how well the representation distinguishes skill advancement and completion over the entire episode, while event-detection delay measures how accurately it detects the events for memory retention. We report **MSE** to measure progress prediction error, **Pearson r** correlation to measure agreement with the target's temporal variation, and **Delay@0.9** to measure the absolute timing error between predicted and ground-truth crossings of the 0.9 progress threshold.

SkillFormer excels at skills that require reading tactile patterns over time. Table 1 shows that vision suffices for reaching, but struggles with repetitive contact interactions. On bulb tightening, Robometer* and ReWiND* achieve correlations of only 0.370 and 0.635, compared with 0.899 for our method, indicating difficulty tracking fine-grained skill progress. Relative to Vision Only,

Table 1: Progress estimation results across various visual or tactile dominant skills.

Method	Metric	Lamp replacement		Bottle capping	
		Reach bulb	Twist bulb	Reach cap	Slide cap
Robometer Liang et al. (2026)	Pearson $r \uparrow$	0.638 ± 0.054	0.209 ± 0.053	0.819 ± 0.084	0.578 ± 0.058
	MSE ($\times 10^{-3}$) $\downarrow$	84.5 ± 19.6	499.8 ± 59.3	46.8 ± 17.4	232.2 ± 42.2
Robometer* Liang et al. (2026)	Pearson $r \uparrow$	0.886 ± 0.032	0.370 ± 0.106	$\underline{0.885 \pm 0.018}$	0.875 ± 0.020
	MSE ($\times 10^{-3}$) $\downarrow$	27.4 ± 6.2	98.6 ± 46.3	51.3 ± 9.5	$\underline{40.7 \pm 4.3}$
ReWiND* Zhang et al. (2025)	Pearson $r \uparrow$	0.847 ± 0.060	0.635 ± 0.136	0.880 ± 0.035	$\underline{0.878 \pm 0.037}$
	MSE ($\times 10^{-3}$) $\downarrow$	$\underline{13.7 \pm 10.4}$	21.4 ± 10.2	11.5 ± 4.7	45.5 ± 11.9
Vision Only	Pearson $r \uparrow$	$\underline{0.997 \pm 0.002}$	$\underline{0.842 \pm 0.095}$	$\mathbf{0.997 \pm 0.003}$	0.798 ± 0.103
	MSE ($\times 10^{-3}$) $\downarrow$	$\mathbf{0.2 \pm 0.1}$	$\underline{9.3 \pm 6.4}$	$\mathbf{0.3 \pm 0.3}$	72.7 ± 45.5
SkillFormer	Pearson $r \uparrow$	$\mathbf{0.998 \pm 0.001}$	$\mathbf{0.899 \pm 0.061}$	$\mathbf{0.997 \pm 0.003}$	$\mathbf{0.948 \pm 0.073}$
	MSE ($\times 10^{-3}$) $\downarrow$	$\mathbf{0.2 \pm 0.1}$	$\mathbf{5.9 \pm 3.9}$	$\underline{0.4 \pm 0.3}$	$\mathbf{15.2 \pm 20.6}$

SkillFormer reduces tightening and cap-sliding MSE by 37% and 79%, supporting the value of tactile cues for distinguishing resistance and slip when successive motions look similar.

Event Memory resolves the representation ambiguity. In Figure 5, Robometer, FiLM, and ReWiND trigger before the bulb is fully tightened prematurely. The w/o Events model detects the event only one frame late, but its prediction subsequently drops, effectively *forgetting* completion. SkillFormer triggers only four frames late while visual observations still resemble ongoing twisting, suggesting that it prioritizes tactile observations to differentiate between the last twist and earlier twists without a tightening tactile pattern. For "*Inspect Box 2*", w/o Events and SkillFormer (*K=1*) predict a negative outcome during an earlier *Inspect Box 1*, when the "*Inspect Box 2*" progress remains zero. This suggests confusion between similar inspections across boxes. In contrast, SkillFormer (*K=3*) detects the positive outcome a few frames before the ground-truth crossing and preserves it afterward, supporting the value of retained history for distinguishing and remembering inspection outcomes. Tab. 2 further illustrates this in the blind cube search task where a vision only transformer with 3 stored events outperforms our SkillFormer (*K=1*) variant. Moreover,

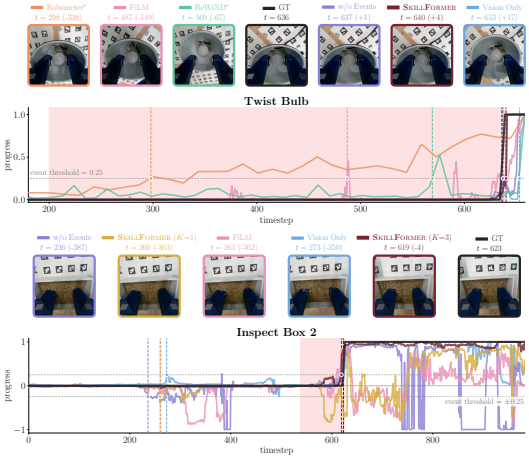


Figure 5: We show observations and their timesteps which correspond to moments when an event is being detected where progress crosses the event threshold for a sample discrete **(top)** and conditional skill **(bottom)** in *Lamp Replacement* and *Blind Cube Search* tasks.

The feature projections in Figure 6 provide complementary qualitative evidence: for the same observations queried with *Reach Box 2*, SkillFormer produces clearer separation between progress states, while the other configurations exhibit greater overlap. For a similar continuous *Reach Bottle Cap*, all models show a smooth progression from early reaching states to completion, consistent with the current visual observations providing sufficient context for this skill.

Limitations of naïve multi-modal token fusion. Tactile VLAs such as Cheng et al. (2026) and Tactile-VLA Huang et al. (2025) concatenate visual, tactile, and language tokens for joint transformer processing. Our w/o Events ablation follows this fusion pattern over recent observations, while FiLM uses tactile-conditioned feature modulation. The SkillFormer (h=2) ablation uses two timesteps for both modalities, testing the benefit of extending the tactile horizon while keeping the visual horizon fixed. In Table 3, FiLM and w/o Events exceed 41 frames of cap-sliding detection delay, compared with 11.73 for SkillFormer. Removing events also more than doubles overall MSE on both tasks, showing the importance of retaining event observations that can never be recovered later. Restricting both

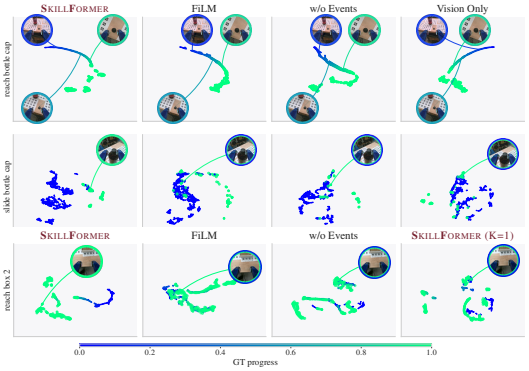


Figure 6: **t-SNE projections** of produced transformer tokens colored by ground-truth progress. Inner/outer image borders indicate gt/predicted progress.

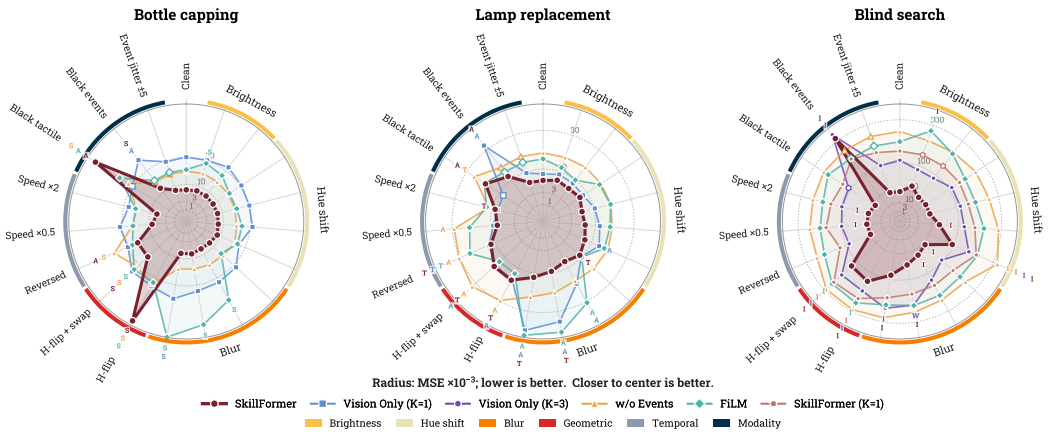


Figure 7: Robustness to visual, geometric, temporal, and modality perturbations across three tasks. closer to the center is better. Colored letters identify the subtask with the largest error increase over clean inputs (R: reach, A: align, S: slide, T: twist, I: inspect, W: withdraw).

modalities to the same two timesteps increases cap-sliding delay to 25.55 frames and bulb twist to 20.07, suggesting that tactile observations benefit from a longer temporal context than vision for skills that require detecting a tactile pattern over a longer horizon. Replacing FAF with generic visual pretraining further weakens the benefit of touch: align cap MSE rises from 6.4 to 12.9, approaching Vision Only's 18.4. Tactile inputs therefore benefit from both appropriate temporal context and specialized features.

SkillFormer is robust to visual and temporal perturbations. Fig. 7 evaluates every model under different types of perturbations. On bottle capping and lamp replacement, SkillFormer's error stays close to its clean value across darkening, hue shifts and blur up to 15 px, while FiLM and Vision Only degrade markedly under strong blur, FiLM's bottle capping error roughly tripling at 7 px. Temporal perturbations are equally benign: halving or doubling the timestep downsampling speed changes SkillFormer's error by at most a few units ($\times 10^{-3}$), and jittering every detected event by ± 5 frames leaves it essentially unchanged on all three tasks, showing that the event memory can tolerate a few frames latency around the exact event frame as it might last for a few frames or in the observation window. Blind search is the most sensitive task, with strong hue shifts and blur raising SkillFormer's error, yet it remains below every ablation at every level. SkillFormer degrades sharply only when the input it relies on are removed, blacked-out tactile on the contact-rich bottle skills and a blacked-out event memory on blind search. These targeted failures provide evidence that SkillFormer actively exploits the modalities and memory tokens prioritized for each skill, rather than succeeding while ignoring them. We refer the reader to the appendix for more discussions on robustness tests.

Table 2: Blind search results averaged over all subtasks.

Method	Delay@0.9 ↓	MSE ($\times 10^{-3}$) ↓	Pearson r ↑
w/o Events	63.46 ± 119.43	143.6 ± 220.0	0.683 ± 0.301
FiLM	77.81 ± 118.55	116.0 ± 253.4	0.727 ± 0.263
SkillFormer (K=1)	42.77 ± 79.83	83.1 ± 201.7	0.791 ± 0.287
Vision Only (K=3)	37.46 ± 87.46	57.1 ± 155.1	0.857 ± 0.213
SkillFormer (K=3)	$\mathbf{16.10 \pm 37.20}$	$\mathbf{11.6 \pm 34.1}$	$\mathbf{0.953 \pm 0.082}$

Table 3: Progress estimation results across various tasks and skills. Overall metrics are reported over the entire skills in each task.

Method	Metric	Lamp replacement				Bottle capping			
		Reach bulb	Align bulb	Twist bulb	Overall	Reach cap	Align cap	Slide cap	Overall
Vision Only	Delay@0.9	3.75 ± 2.09	15.50 ± 17.22	19.75 ± 8.69	10.08 ± 11.48	5.36 ± 3.31	29.64 ± 25.97	82.18 ± 82.08	27.67 ± 46.81
	MSE ($\times 10^{-3}$)	$\mathbf{0.2 \pm 0.1}$	13.8 ± 18.8	9.3 ± 6.4	5.4 ± 9.7	$\mathbf{0.3 \pm 0.3}$	18.4 ± 14.8	72.7 ± 45.5	17.8 ± 32.7
	Pearson r	0.997 ± 0.002	0.967 ± 0.039	0.842 ± 0.095	0.904 ± 0.160	0.997 ± 0.003	0.963 ± 0.028	0.798 ± 0.103	0.950 ± 0.085
w/o Events	Delay@0.9	4.17 ± 6.24	15.50 ± 15.96	9.92 ± 9.30	7.92 ± 10.41	5.00 ± 3.41	17.18 ± 8.63	41.09 ± 14.99	17.36 ± 15.50
	MSE ($\times 10^{-3}$)	0.4 ± 0.8	40.3 ± 53.0	8.4 ± 11.6	9.0 ± 26.4	$\mathbf{0.3 \pm 0.2}$	8.1 ± 7.7	46.8 ± 13.1	11.2 ± 18.1
	Pearson r	0.991 ± 0.023	0.915 ± 0.095	0.811 ± 0.285	0.906 ± 0.173	$\mathbf{0.998 \pm 0.002}$	0.984 ± 0.015	0.847 ± 0.046	0.961 ± 0.061
w/o FAF	Delay@0.9	–	–	–	–	6.00 ± 4.24	18.82 ± 12.88	48.27 ± 12.13	19.52 ± 20.51
	MSE ($\times 10^{-3}$)	–	–	–	–	$\mathbf{0.3 \pm 0.3}$	12.9 ± 11.3	55.3 ± 16.5	13.6 ± 21.2
	Pearson r	–	–	–	–	0.997 ± 0.003	0.974 ± 0.023	0.817 ± 0.057	0.953 ± 0.071
FiLM	Delay@0.9	5.08 ± 2.63	18.17 ± 22.05	11.75 ± 18.66	8.97 ± 14.55	7.73 ± 5.71	7.27 ± 3.82	41.82 ± 15.10	14.27 ± 15.70
	MSE ($\times 10^{-3}$)	0.4 ± 0.4	26.1 ± 24.7	14.2 ± 13.2	14.7 ± 22.1	0.6 ± 0.5	8.0 ± 8.6	50.7 ± 20.3	11.6 ± 20.6
	Pearson r	0.990 ± 0.011	0.939 ± 0.051	0.749 ± 0.242	0.828 ± 0.276	0.995 ± 0.004	0.984 ± 0.016	0.834 ± 0.067	0.960 ± 0.069
SkillFormer (h=2)	Delay@0.9	2.92 ± 2.66	$\mathbf{11.83 \pm 10.12}$	75.75 ± 170.18	20.07 ± 81.29	$\mathbf{4.55 \pm 3.11}$	10.64 ± 5.65	25.55 ± 29.64	14.44 ± 18.63
	MSE ($\times 10^{-3}$)	$\mathbf{0.2 \pm 0.1}$	$\mathbf{12.2 \pm 12.2}$	14.0 ± 22.6	6.2 ± 13.0	0.4 ± 0.3	11.2 ± 12.9	24.0 ± 38.6	7.7 ± 18.9
	Pearson r	0.997 ± 0.002	$\mathbf{0.970 \pm 0.029}$	0.827 ± 0.210	0.914 ± 0.176	0.996 ± 0.003	0.978 ± 0.025	0.914 ± 0.146	0.973 ± 0.068
SkillFormer	Delay@0.9	$\mathbf{2.33 \pm 1.31}$	13.92 ± 14.92	$\mathbf{6.42 \pm 3.25}$	$\mathbf{7.60 \pm 9.20}$	5.36 ± 3.11	$\mathbf{6.27 \pm 5.64}$	$\mathbf{11.73 \pm 16.58}$	$\mathbf{10.70 \pm 15.78}$
	MSE ($\times 10^{-3}$)	$\mathbf{0.2 \pm 0.1}$	12.5 ± 14.6	$\mathbf{5.9 \pm 3.9}$	$\mathbf{3.4 \pm 7.7}$	0.4 ± 0.3	$\mathbf{6.4 \pm 5.8}$	$\mathbf{15.2 \pm 20.6}$	$\mathbf{4.6 \pm 10.3}$
	Pearson r	$\mathbf{0.998 \pm 0.001}$	$\mathbf{0.970 \pm 0.033}$	$\mathbf{0.899 \pm 0.061}$	$\mathbf{0.971 \pm 0.045}$	0.997 ± 0.003	$\mathbf{0.987 \pm 0.012}$	$\mathbf{0.948 \pm 0.073}$	$\mathbf{0.985 \pm 0.035}$

Conclusion We presented SkillFormer, a skill-conditioned multimodal representation that combines tactile temporal context with event memory. Through extensive evaluation of common multimodal representations across three manipulation tasks, we demonstrate improved progress estimation and event detection that would not possible with simple multimodal concatenation or without event memory. Predicted progress could provide rewards for policy learning, while the representation could condition action generation on current observations and retained history. Evaluating these applications in closed-loop control remains future work.

AI USE STATEMENT

In this work, we used generative AI tools for implementing methods and for assisting with automated experiment running. We did not use generative AI tools for the other tasks requiring disclosure under the ICLR 2027 AI Policy for Authors, including generating synthetic datasets, developing theoretical models or conceptual frameworks, formulating mathematical claims, providing ingredients for proofs, assisting in the writing of proofs, proposing or refining hypotheses, designing or providing feedback on research methodology or experiments, translation, cleaning or reformatting datasets, supporting qualitative or thematic data analysis, or interpreting results. Additionally, we used generative AI tools to assist with drafting and revising portions of the manuscript, including improving readability, organization, and the clarity of technical descriptions. All AI-assisted code, experiment automation, and text were reviewed by the authors. Code and experiment outputs were checked through execution and manual inspection, and the manuscript text was revised and verified by the authors for accuracy. We take responsibility for the final content of this work, including text, claims, code, and other artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We will open-source all data and skill annotations, trained models, and code for training and evaluation. Section 3 describes the architecture, skill-progress supervision, and event-memory construction, while Sections 4 and 5 describe the datasets, episode-level splits, baselines, and evaluation metrics. Our skill-progress formulation supports reproducibility by allowing the model and its sparse event memory to be evaluated on recorded visual and tactile sequences, without recreating the physical data-collection setup or executing a robot policy. Progress targets are derived from explicit skill annotations, and memory updates follow the specified event-detection and retention rules, allowing both the supervision and the retained observation history to be reconstructed and inspected using the released data, models, and code.

REFERENCES

Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, et al. Do as i can, not as i say: Grounding language in robotic affordances. In *Conference on Robot Learning (CoRL)*, 2022.

Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Misha Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. Decision transformer: Reinforcement learning via sequence modeling. *Advances in neural information processing systems*, 34:15084–15097, 2021.

Qianzhong Chen, Justin Yu, Mac Schwager, Pieter Abbeel, Fred Shentu, and Philipp Wu. Sarm: Stage-aware reward modeling for long horizon robot manipulation. In *International Conference on Learning Representations*, volume 2026, pp. 17091–17116, 2026.

Yizhou Chen, Andrea Sipos, Mark Van der Merwe, and Nima Fazeli. Visuo-tactile transformers for manipulation. *arXiv preprint arXiv:2210.00121*, 2022.

Zhengxue Cheng, Yiqian Zhang, Anni Tang, Keyu Wang, Wenkang Zhang, Haoyu Li, Hengdi Zhang, and Li Song. Omnivtla: Vision-tactile-language-action models with semantic-aligned tactile sensing, 2026. URL https://arxiv.org/abs/2508.08706.

Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Robotics: Science and Systems (RSS)*, 2023.

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, 2024. URL https://arxiv.org/abs/2402.10329.

Ruoxuan Feng, Jiangyu Hu, Wenke Xia, Tianci Gao, Ao Shen, Yuhao Sun, Bin Fang, and Di Hu. Any-touch: Learning unified static-dynamic representation across multiple visuo-tactile sensors. In *International Conference on Learning Representations*, volume 2025, pp. 31265–31285, 2025.

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.

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.

Jialei Huang, Shuo Wang, Fanqi Lin, Yihang Hu, Chuan Wen, and Yang Gao. Tactile-vla: Unlocking vision-language-action model's physical knowledge for tactile generalization, 2025. URL https://arxiv.org/abs/2507.09160.

Oliver Kroemer, Scott Niekum, and George Konidaris. A review of robot learning for manipulation: Challenges, representations, and algorithms. *Journal of Machine Learning Research*, 22(30):1–82, 2021.

Michelle A Lee, Yuke Zhu, Peter Zachares, Matthew Tan, Krishnan Srinivasan, Silvio Savarese, Li Fei-Fei, Animesh Garg, and Jeannette Bohg. Making sense of vision and touch: Learning multimodal representations for contact-rich tasks. *arXiv preprint arXiv:1907.13098*, 2019.

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.

Anthony Liang, Yigit Korkmaz, Jiahui Zhang, Minyoung Hwang, Abrar Anwar, Sidhant Kaushik, Aditya Shah, Alex S. Huang, Luke Zettlemoyer, Dieter Fox, Yu Xiang, Anqi Li, Andreea Bobu, Abhishek Gupta, Stephen Tu, Erdem Biyik, and Jesse Zhang. Robometer: Scaling general-purpose robotic reward models via trajectory comparisons, 2026. URL https://arxiv.org/abs/2603.02115.

Yecheng Jason Ma, Shagun Sodhani, Dinesh Jayaraman, Osbert Bastani, Vikash Kumar, and Amy Zhang. Vip: Towards universal visual reward and representation via value-implicit pre-training. *arXiv preprint arXiv:2210.00030*, 2022.

Yecheng Jason Ma, William Liang, Shikhar Bahl, Dinesh Wang, Deepak Pathak, and Mustafa Mukadam. LIV: Language-image representations and rewards for robotic control. In *International Conference on Machine Learning (ICML)*, 2023.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2023.

Abby O'Neill, Abdul Rehman, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, et al. Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6892–6903. IEEE, 2024.

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.

Amir Hossein Shahidzadeh, Ga Caddeo, Koushik Alapati, Lorenzo Natale, Cornelia Fermüller, and Yiannis Aloimonos. Feelanyforce: Estimating contact force feedback from tactile sensation for vision-based tactile sensors. *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 251–257, 2024.

Hao Shi, Bin Xie, Yingfei Liu, Lin Sun, Fengrong Liu, Tiancai Wang, Erjin Zhou, Haoqiang Fan, Xiangyu Zhang, and Gao Huang. Memoryvla: Perceptual-cognitive memory in vision-language-action models for robotic manipulation. In *International Conference on Learning Representations*, volume 2026, pp. 18567–18602, 2026.

Lucy Xiaoyang Shi, Brian Ichter, Michael Equi, Liyiming Ke, Karl Pertsch, Quan Vuong, James Tanner, Anna Walling, Haohuan Wang, Niccolo Fusai, et al. Hi robot: Open-ended instruction following with hierarchical vision-language-action models. *arXiv preprint arXiv:2502.19417*, 2025.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Ganlin Yang, Zhangzheng Tu, Yuqiang Yang, Sitong Mao, Junyi Dong, Tianxing Chen, Jiaqi Peng, Jing Xiong, Jiafei Cao, Jifeng Dai, et al. Eventvla: Event-driven visual evidence memory for long-horizon vision-language-action policies. *arXiv preprint arXiv:2606.20092*, 2026.

En Yu, Haoran Lv, Jianjian Sun, Kangheng Lin, Ruitao Zhang, Yukang Shi, Yuyang Chen, Ze Chen, Ziheng Zhang, Fan Jia, Kaixin Liu, Meng Zhang, Ruitao Hao, Saike Huang, Songhan Xie, Yu Liu, Zhao Wu, Bin Xie, Pengwei Zhang, Qi Yang, Xianchi Deng, Yunfei Wei, Enwen Zhang, Hongyang Peng, Jie Zhao, Kai Liu, Wei Sun, Yajun Wei, Yi Yang, Yunqiao Zhang, Ziwei Yan, Haitao Yang, Hao Liu, Haoqiang Fan, Haowei Zhang, Junwen Huang, Yang Chen, Yunchao Ma, Yunhuan Yang, Zhengyuan Du, Ziming Liu, Jiahui Niu, Yucheng Zhao, Daxin Jiang, Wenbin Tang, Xiangyu Zhang, Zheng Ge, Erjin Zhou, and Tiancai Wang. Dm0: An embodied-native vision-language-action model towards physical ai, 2026. URL https://arxiv.org/abs/2602.14974.

Yihan Zeng, Minghao Ye, Yiyuan Chen, Yide Shentu, Philipp Wu, Zike Yan, and Zhongyu Li. Kemo: Event-driven keyframe memory for long-horizon robot manipulation with vla policies. *arXiv preprint arXiv:2606.23589*, 2026.

Jiahui Zhang, Yusen Luo, Abrar Anwar, Sumedh Anand Sontakke, Joseph J Lim, Jesse Thomason, Erdem Biyik, and Jesse Zhang. Rewind: Language-guided rewards teach robot policies without new demonstrations, 2025. URL https://arxiv.org/abs/2505.10911.

A Appendix

A.1 Tasks

Each task is represented as an ordered sequence of language-addressable primitive skills. For example, bottle capping is decomposed into *reach bottle cap*, *grasp cap*, *reach bottle*, *align cap*, and *slide cap*. Since progress is estimated conditioned on the queried skill description, the same representation can be reused across different phases of a task while changing only the language query. This formulation separates skill-level progress estimation from the particular long-horizon task sequence, enabling primitive skills to serve as reusable units for composing longer manipulation behaviors. We describe the task designs below.

A.1.1 Lamp Replacement

In the lamp replacement task, the robot picks up a bulb, moves it toward the lamp socket, aligns it with the opening, and twists it until the bulb reaches the end of the thread. The task consists of five skills: *reach bulb*, *grasp bulb*, *reach lamp socket*, *align bulb*, and *twist bulb*. The reaching skills are primarily visual, since progress is correlated with the relative position of the gripper and target object. In contrast, grasping and twisting require tactile feedback. The twist skill is especially contact-dependent: it repeats until a completion is indicated by increased tactile force (see Figure 8) once the bulb is fully tightened. This setting emphasizes the need for tactile temporal context, a single tactile frame may show contact, but the sequence of tactile frames reveals whether the interaction corresponds to normal rotation or terminal tightening. We disable the light-on visual cue so that twist completion cannot be inferred from appearance alone.

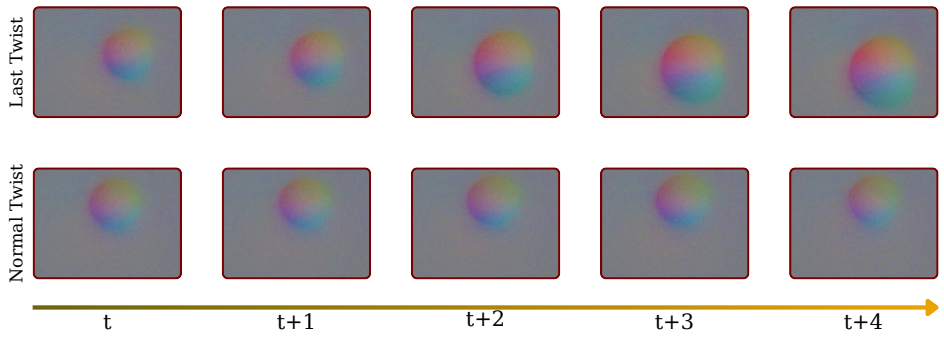


Figure 8: Tactile evolution during normal and final bulb twisting. During the last twist (top row), the tactile image shows an increasing-force pattern as resistance builds near the end of the thread. During normal twisting, the contact pattern remains relatively stable and changes only slightly across frames. This temporal tactile cue defines completion of the *twist bulb* skill.

A.1.2 Bottle Capping

In the bottle capping task, the robot grasps a cap, moves it to the bottle opening, aligns it with the bottle, and slides it until terminal contact or slippage is detected. The task is decomposed into *reach bottle cap*, *grasp cap*, *reach bottle*, *align cap*, and *slide cap*. This task tests whether the skill-conditioned representation can facilitate switching between vision-dominant and touch-dominant phases. Reaching and coarse alignment depend mostly on RGB observations, while grasping and sliding depend on tactile deformation. The slide skill is difficult because the cap can remain visually close to the bottle while the tactile signal changes sharply due to contact and slippage.

A.1.3 Blind Cube Search

Blind cube search is designed to test tactile object identification. The robot searches candidate boxes containing hidden shapes, including a cube and distractor objects such as pentagonal prism and cylinder. Since the hidden object is not visible, the robot must infer whether it has found the cube from tactile deformation patterns. The task is decomposed into *reach box*, *inspect box*, *withdraw*, *reach puzzle box*, and *insert cube*. If the inspected object is not the cube, the robot withdraws and reaches to the next candidate box. If the cube is detected, the robot carries it to the puzzle box and inserts it into the square hole. This task evaluates whether the representation can use contact geometry when visual observations are ambiguous or uninformative. Table. 4 reports an extensive version of Table 2 per important per various skills in blind cube search task.

Table 5: Skill definitions used for progress supervision.

Task	Skill Description	Terminal condition	Progress type
Lamp replacement	Reach bulb	End-effector reaches the bulb region	Cont.
	Grasp bulb	Stable grasp is established on the bulb	Cont.
	Reach lamp socket	End-effector reaches the socket region while holding the bulb	Cont.
	Align bulb	Bulb is aligned with the socket opening	Disc.
	Twist bulb	Bulb reaches the end of the thread, producing increased tactile force	Disc.
Bottle capping	Reach bottle cap	End-effector reaches the target object	Cont.
	Grasp cap	Stable grasp is established on the cap	Cont.
	Reach bottle	End-effector reaches the bottle opening while holding the cap	Cont.
	Align cap	Cap is aligned with the bottle opening	Disc.
	Slide cap	Cap slides against the bottle until slippage is detected	Disc.
Blind cube search	Reach box k	End-effector reaches the candidate box region	Cont.
	Inspect box k	Grasp force reaches a level at which the cube can be identified or ruled out	Cond.
	Withdraw	End-effector leaves the interaction region before reaching box $k{+}1$	Cont.
	Reach puzzle box	End-effector reaches the puzzle box region while holding the cube	Cont.
	Insert cube	Cube is inserted into the square hole	Cont.

Single-stream versus cross-attention fusion. Our choice of bidirectional self-attention over a unified token sequence is related to the distinction between single-stream and dual-stream vision-language architectures. In dual-stream or cross-attention formulations, language is often used as a query over sensory features. This can be effective for selecting or gating relevant visual tokens, but the resulting representation is still primarily an attended summary of the sensory stream. The language condition may not be explicitly preserved throughout later layers.

This distinction is important for skill progress estimation. In our setting, progress is not a function of the observation alone, but of both the observation and the queried skill. The same RGB and tactile observations can correspond to very different progress values depending on the skill query. For example, a frame near the bottle opening may indicate high progress for *reach bottle*, but low progress for *slide cap*; similarly, a tactile contact pattern may be relevant for *grasp cap* but not for *reach bottle*. Therefore, language should not only gate sensory features, but remain part of the representation used to compute progress.

SKILLFORMER addresses this by treating the skill description as a first-class token in the same sequence as RGB, tactile, event-memory, and output tokens. Self-attention allows language, current observations, event memory, and the output token to mutually update each other across layers. This enables the output representation to remain explicitly skill-conditioned, which is especially important when visually similar observations occur across adjacent skills or when tactile observations must be interpreted differently depending on the queried primitive.

B EXPERIMENTS DETAILS

We compare SKILLFORMER against several baselines that isolate the effect of language conditioning, temporal horizon, tactile sensing, and event memory. We use a batch size of 45, train for 100 epochs, use AdamW with base learning rate 10^{-4}, and fine-tune pretrained encoders with a smaller learning rate of 10^{-5}. All transformer-based models use hidden dimension $d = 512$, 8 attention heads, and 4 transformer layers. We use active-skill sampling with $\alpha = 0.8$, so most training queries correspond to the currently active skill while a small fraction are sampled from other skills to enforce language conditioning.

Table 4: Progress estimation results across blind cube search skills.

Method	Metric	Reach box 1	Inspect box 1	Inspect box 2	Withdraw 2	Overall
w/o Events	Delay@0.9 ↓	25.57 ± 12.53	31.00 ± 15.43	36.10 ± 11.55	66.20 ± 96.58	63.46 ± 119.43
	MSE ($\times 10^{-3}$) ↓	23.5 ± 19.6	496.0 ± 455.7	287.1 ± 255.4	131.4 ± 89.4	143.6 ± 220.0
	Pearson r ↑	0.833 ± 0.128	0.355 ± 0.350	0.462 ± 0.449	0.580 ± 0.116	0.683 ± 0.301
FiLM	Delay@0.9 ↓	58.14 ± 25.27	47.50 ± 18.94	100.80 ± 46.24	63.00 ± 92.80	77.81 ± 118.55
	MSE ($\times 10^{-3}$) ↓	14.7 ± 13.2	495.1 ± 649.8	171.7 ± 132.2	56.0 ± 74.4	116.0 ± 253.4
	Pearson r ↑	0.868 ± 0.101	0.400 ± 0.316	0.621 ± 0.238	0.575 ± 0.250	0.727 ± 0.263
SKILLFORMER (K=1)	Delay@0.9 ↓	17.38 ± 11.81	61.67 ± 67.35	88.50 ± 55.27	198.60 ± 177.28	42.77 ± 79.83
	MSE ($\times 10^{-3}$) ↓	2.0 ± 1.9	388.6 ± 513.1	136.3 ± 127.6	47.2 ± 81.6	83.1 ± 201.7
	Pearson r ↑	0.986 ± 0.012	0.505 ± 0.512	0.705 ± 0.274	0.597 ± 0.297	0.791 ± 0.287
Vision Only (K=3)	Delay@0.9 ↓	$\mathbf{5.38 \pm 4.29}$	$\mathbf{7.33 \pm 6.99}$	15.00 ± 7.10	102.20 ± 123.69	37.46 ± 87.46
	MSE ($\times 10^{-3}$) ↓	1.3 ± 1.6	224.5 ± 377.1	131.0 ± 194.3	24.1 ± 25.4	57.1 ± 155.1
	Pearson r ↑	0.992 ± 0.007	0.712 ± 0.343	0.749 ± 0.261	0.900 ± 0.039	0.857 ± 0.213
SKILLFORMER (K=3)	Delay@0.9 ↓	11.52 ± 6.51	8.33 ± 4.99	$\mathbf{8.70 \pm 8.04}$	$\mathbf{14.20 \pm 21.97}$	$\mathbf{16.10 \pm 37.20}$
	MSE ($\times 10^{-3}$) ↓	$\mathbf{0.9 \pm 0.7}$	$\mathbf{50.7 \pm 62.2}$	$\mathbf{18.6 \pm 24.3}$	$\mathbf{4.6 \pm 10.8}$	$\mathbf{11.6 \pm 34.1}$
	Pearson r ↑	$\mathbf{0.994 \pm 0.005}$	$\mathbf{0.898 \pm 0.106}$	$\mathbf{0.958 \pm 0.045}$	$\mathbf{0.986 \pm 0.023}$	$\mathbf{0.953 \pm 0.082}$

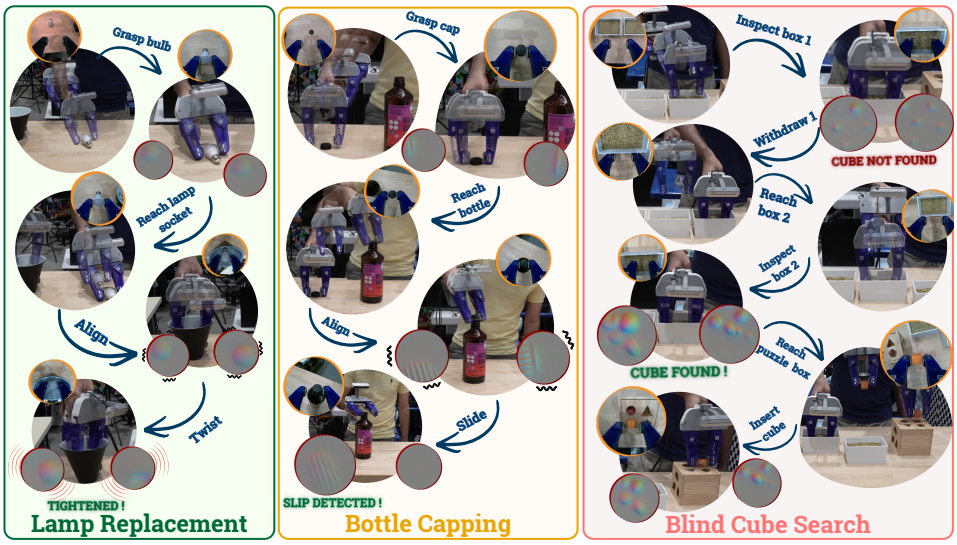


Figure 9: **Contact-rich manipulation tasks and their representation requirements.** *Left:* Lamp replacement involves grasping, socket alignment, and twisting until tightened. *Middle:* Bottle capping involves grasping, thread alignment, and sliding, with slip providing a salient tactile event. *Right:* Blind cube search involves successive inspections and withdrawal after an unsuccessful attempt, followed by insertion into a puzzle box once the target is found. Arrows indicate primitive skills. Large images highlight a task execution; orange-bordered insets show egocentric RGB views, and red-bordered insets show tactile readings from the two fingers. Together, the tasks illustrate skill-specific tactile patterns and complementary needs for sensory history and memory of prior interactions for estimating their progress.

B.1 Baselines Explained

FiLM-conditioned baseline. The FiLM baseline replaces the self-attention-based language-conditioned representation with a FiLM-style conditioning mechanism. It uses the same RGB and tactile horizons as the main model, with $H_{\text{rgb}} = 2$ and $H_{\text{tac}} = 6$, but does not use event memory. The architecture conditions visual and tactile features on the queried skill embedding through feature-wise affine modulation before progress estimation. This baseline tests whether explicit language modulation through FiLM is sufficient for progress estimation, compared to our token-based formulation that jointly attends over language, RGB, tactile, and memory tokens.

SkillFormer (h=2) baseline. The short-horizon baseline uses the same architecture as SkillFormer but restricts both RGB and tactile inputs to two frames, i.e., $H_{\text{rgb}} = H_{\text{tac}} = 2$. Event memory and tactile inputs are still enabled. This baseline reflects the common observation formulation used in manipulation policies, where the model receives only a small fixed window of recent observations. Comparing against this baseline tests whether modality-specific temporal horizons are necessary, especially for tactile signals whose informative patterns often emerge over multiple frames.

Vision only baseline. The Vision-only baseline removes both tactile streams and estimates progress using only RGB observations and the queried skill description. This baseline measures how much progress estimation can be solved from visual information alone. It is particularly important for contact-rich skills such as *twist bulb*, *slide cap*, and *inspect box*, where the terminal condition depends on contact force, slippage, or tactile shape information rather than appearance.

No FAF encoder baseline. The generic tactile encoder baseline keeps the same architecture and tactile inputs as SkillFormer, but replaces the tactile-specialized FAF encoder with a generic ImageNet-pretrained DINOv2 encoder. This baseline isolates the importance of tactile pretraining. Since tactile images differ significantly from natural images, this comparison tests whether contact-aware tactile features are needed for reliable progress estimation.

No event memory baseline. The no-event-memory baseline removes the event-memory tokens while keeping the current RGB, tactile, and language tokens unchanged. This baseline tests whether storing the terminal observation of the previous skill helps reduce ambiguity near skill transitions. Without event memory, the model must infer the current phase only from the recent observation window, which can be ambiguous when adjacent skills have visually similar states but different completion criteria.

Robometer fine-tuning. We start from the released Robometer-4B checkpoint Liang et al. (2026), which builds on Qwen3-VL-4B-Instruct and was pretrained on RBM-1M, and adapt it to our tasks with LoRA (rank 32, α=64, dropout 0.05) on all attention and MLP projections of the language model, while training its progress, success and preference heads in full. Fine-tuning uses the bottle capping and lamp replacement training demonstrations, converted to Robometer's format with per-frame progress labels for each subtask, and mixes progress and preference samples as in the original recipe. Each sample contains 48 frames drawn from the episode. We train with a global batch of 8 on two H-200 GPUs for up to 3,000 steps, using AdamW with a learning rate of 2×10^{-5}, 10% warmup and a cosine schedule, and keep the checkpoint with the best held-out progress correlation.

ReWiND training. ReWiND Zhang et al. (2025) predicts progress with a four-layer transformer (hidden size 512, 8 heads) over frozen DINOv2 ViT-B/14 image features and MiniLM-L12 language embeddings. We train a single model jointly on our bottle capping and lamp replacement demonstrations and on Open X-Embodiment, with 30% of each batch drawn from our data. Following the original method, 80% of training clips are augmented with video rewinding, which appends a reversed segment so that the model learns progress can decrease, and 20% pair a clip with a mismatched instruction as negatives with zero progress. Clips contain up to 16 frames, and we train for 20,000 steps with a batch of 256 and a learning rate of 10^{-4} decaying to 10^{-5} after 500 warmup steps, selecting the checkpoint with the best held-out per-frame correlation.

C Further Discussions on Robustness

What the perturbations test. Figure 7 examines whether the representations remain useful when image appearance, sensor correspondence, temporal sampling, or access to observations changes at test time. These interventions probe different dependencies: appearance tests target visual cues, geometric tests disrupt spatial relationships, temporal tests alter motion information, and input removal tests reliance on touch and memory. All MSE values below are reported in $\times10^{-3}$.

Brightness tests sensitivity to image intensity. Darkening by factors of 0.75 and 0.5 tests whether predictions depend on the image intensity seen during training. Unlike a change in task state, this transformation preserves the scene layout while reducing the visibility of image details. At half brightness, SkillFormer's MSE changes from 3.4 to 3.8 on lamp replacement and from 4.6 to 4.7 on bottle capping, but rises from 15.1 to 25.3 on blind search. This suggests greater sensitivity to visual appearance when interpreting the repeated interactions and spatial context of the search task.

Hue shifts test dependence on color. Hue shifts from 18° to 108° alter color while preserving spatial arrangement, testing whether predictions depend on familiar colors rather than task-relevant geometry and contact. At 108°, lamp and bottle MSE remain low at 4.2 and 5.1, respectively, whereas blind-search MSE rises to 54.4. The assembly results suggest that exact color is not essential to their progress predictions. The search result reveals sensitivity to altered visual appearance, but does not by itself establish that the model uses color to identify boxes.

Blur tests dependence on fine image detail. Gaussian blur with kernels from 3 to 15 px suppresses edges and texture, probing whether the model requires precise image details to interpret manipulation. At 15 px, bottle-capping MSE remains close to clean performance (4.7 versus 4.6), whereas lamp MSE rises from 3.4 to 6.8 and blind-search MSE from 15.1 to 54.4. FiLM's bottle-capping MSE already rises from 11.5 to 30.9 at 7 px. The differing responses are consistent with complementary sensing requirements: contact observations can help resolve cap sliding, while visual detail remains useful for alignment and interpreting search interactions.

Sampling speed tests tolerance to temporal scale. The speed perturbation multiplies the spacing between sampled frames by 0.5 or 2, changing the elapsed span of an observation window without changing its number of timesteps. It therefore tests sensitivity to temporal sampling rather than the physical effects of executing the robot faster or slower. SkillFormer's MSE changes by at most 3.0 across the three tasks, suggesting that its predictions tolerate these changes in the sampled rate of motion and contact evolution.

Frame reversal tests the importance of temporal order. Reversal preserves the images in each window but presents them in the opposite order. This distinguishes sensitivity to the sequence of observations from dependence on their presence alone. For example, a developing contact followed by slip conveys different information from the reversed sequence, even though the individual images are unchanged. Reversal increases SkillFormer's MSE from 3.4 to 6.6 on lamp replacement and from 4.6 to 11.7 on bottle capping, compared with only 15.1 to 16.2 on blind search. The larger assembly effects support the importance of ordered contact dynamics in these tasks.

Geometric perturbations probe visual–tactile correspondence. H-flip mirrors the wrist-camera image and both tactile images while keeping each tactile stream in its original input slot. This changes the apparent locations of the fingers and contacts without exchanging their sensor identities. H-flip + swap additionally exchanges the tactile streams, accounting for the reversal of finger positions. Comparing these interventions

probes whether the representation depends on correspondence between visually observed contact and finger-specific tactile patterns. However, image transformations alone do not guarantee a physically equivalent mirrored interaction, and their effects do not directly identify a particular attention mechanism.

Bottle capping depends on finger-specific contact patterns. During cap alignment and repetitive sliding, contact can develop differently across the two fingers. The location and evolution of these contacts can help distinguish continued sliding from the slip event preceding the final twist. Mirroring images without exchanging sensor identities can disrupt the learned association between these tactile patterns and their visual locations. Consistent with this interpretation, SKILLFORMER's MSE increases from 4.6 to 68.4 under H-flip, then falls to 12.4 with the additional swap. The substantial recovery supports sensitivity to sensor correspondence, while the remaining gap indicates that swapping does not remove all effects of mirroring.

Lamp replacement shows a different sensitivity. Bulb tightening involves a grasp in which increasing resistance can affect both tactile sensors. If completion cues are shared across the fingers, their exact left–right assignment may matter less than their evolution over time. H-flip increases MSE from 3.4 to 8.9, and swapping the tactile streams leaves it at 8.9. This is consistent with disruption of visual geometry or within-sensor contact patterns that exchanging streams cannot correct; it does not establish that the two fingers experience identical forces.

Tactile blackout tests dependence on contact observations. Blacking out tactile images removes contact information from both current and retained observations. This tests whether the model uses touch when visually similar motions have different physical outcomes. Lamp MSE rises from 3.4 to 17.8, and bottle MSE from 4.6 to 63.4, consistent with the importance of resistance during tightening and slip during cap sliding. Blind-search MSE changes little (15.1 to 14.2), indicating limited dependence on tactile input in this evaluation. Because blackout introduces an unfamiliar input, it measures sensitivity to removing touch at deployment rather than performance after retraining without it.

Event blackout tests dependence on retained history. Blacking out event memory removes access to earlier interactions while preserving the current observation window. Lamp and bottle MSE rise to 6.4 and 13.2, respectively, while blind-search MSE rises from 15.1 to 171.4. This task requires remembering inspection outcomes to interpret box-specific skill progress: similar current observations may belong to different branches depending on where the cube was found. The large degradation supports the contribution of retained history beyond local sensing. This intervention complements w/o Events, which is trained to operate without memory rather than having memory removed at test time.

Event jitter tests sensitivity to retention timing. Perturbing detected event timestamps by ± 5 frames tests whether memory depends on selecting an exact observation at a skill transition. Performance changes little across the three tasks, suggesting tolerance to small localization errors. A plausible explanation is that each memory entry contains an observation window: nearby event timestamps can retain overlapping observations, and some contact outcomes remain visible beyond a single frame. This result concerns small timing offsets, rather than missing events or incorrect event identities.

Memory complements the current observation window. Without events, the model must infer progress entirely from its recent observations. Frame reversal produces larger error increases for w/o Events on both assembly tasks, consistent with greater dependence on local temporal patterns. Retained events provide additional context after informative interactions leave that window. This benefit is task-dependent: memory improves robustness in several settings, but does not prevent failures when geometric perturbations disrupt the sensory relationships the model has learned.

D LIMITATIONS.

Our main focus in this work was the representation quality that we assessed through progress estimation. However, our current formulation assumes access to a skill graph of predefined language-described skills and annotated skill boundaries for progress supervision. While this allows controlled analysis of representation quality, future work should learn skill segmentation and progress labels with weaker supervision. Integrating SKILLFORMER into downstream control policies and measuring improvements in task success remains an important next step. Additionally, All experiments use the same sensing platform and evaluate unseen episodes from the three collected task domains. Our results therefore establish robustness across different skill types and sensory dependencies within this setting, but do not establish transfer to unseen object categories, environments, embodiments, or sensor configurations. Evaluating such transfer is an important next step.