KISA: A Unified Keyframe Identifier and Skill Annotator for Long-Horizon Robotics Demonstrations

Longxin Kou [*1] **Fei Ni** [*1] **Yan Zheng** [1] **Jinyi Liu** [1] **Yifu Yuan** [1] **Zibin Dong** [1] **Jianye Hao** [1]

Abstract

Robotic manipulation tasks often span over long horizons and encapsulate multiple subtasks with different skills. Learning policies directly from long-horizon demonstrations is challenging without intermediate keyframes guidance and corresponding skill annotations. Existing approaches for keyframe identification often struggle to offer reliable decomposition for low accuracy and fail to provide semantic relevance between keyframes and skills. For this, we propose a unified **K**eyframe **I**dentifier and **S**kill **A**notator (KISA) that utilizes pretrained visual-language representations for precise and interpretable decomposition of unlabeled demonstrations. Specifically, we develop a simple yet effective temporal enhancement module that enriches frame-level representations with expanded receptive fields to capture semantic dynamics at the video level. We further propose coarse contrastive learning and fine-grained monotonic encouragement to enhance the alignment between visual representations from keyframes and language representations from skills. The experimental results across three benchmarks demonstrate that KISA outperforms competitive baselines in terms of accuracy and interpretability of keyframe identification. Moreover, KISA exhibits robust generalization capabilities and the flexibility to incorporate various pretrained representations. KISA can serve as a reliable tool to unleash scalable keyframes and skill annotation to facilitate efficient policy learning from fine-grained decomposed demonstrations. The details and visualizations are available at the project website.

[*]Equal contribution [1]College of Intelligence and Computing, Tianjin University, Tianjin, China. Correspondence to: Jianye Hao <jianye.hao@tju.edu.cn>, Yan Zheng <yanzheng@tju.edu.cn>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

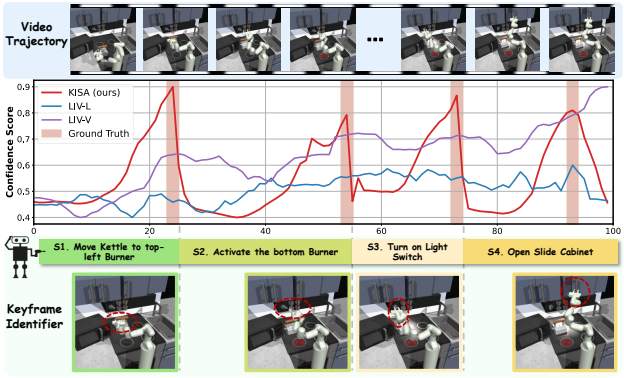

Figure 1. **Overview of Keyframe Identification.** Both goal image similarity (LIV-I) and skill language similarity (LIV-L) struggle to stand out at keyframes. KISA can exhibit conspicuous peaks near groundtruth boundaries for accurate keyframe identification.

1. Introduction

Complex robotics manipulation tasks such as desktop tidying often span over long horizons and encapsulate multiple sub-tasks separated by keyframes (Pertsch et al., 2020; Huang et al., 2023). Directly learning from long-horizon demonstrations in an end-to-end manner is challenging, due to the compounding errors of policy learning without intermediate keyframe guidance and dense skill supervision. Some recent methods (Mees et al., 2022a; Mishra et al., 2023) turn to hierarchical policy learning, by decomposing a complex demonstration into several shorter subtasks to facilitate the reusable skills and further enable modular skill composition for generalization. However, obtaining demonstrations with explicit keyframe boundaries and skill annotations is difficult, especially for real-world human videos. While some methods (Lynch & Sermanet, 2020; Nair et al., 2022a; Lynch et al., 2023) have explored crowd-sourced annotations, but struggle to scale across large amounts of demonstrations for the cost. An alternative is to directly discover latent reusable skills from unlabeled demonstrations, as in (Garg et al., 2022; Xu et al., 2023), but the learned latent skill variables do not have clear semantics alignment with human-interpretable skills for language conditioning. For this, we study the following open question - *can we develop a framework that enables automatic, scalable, and semantically meaningful keyframe identification and skill annotation from unlabeled demonstrations?*

Some existing works focus on identifying keyframes with privileged information such as actions (Arjona-Medina et al., 2019), policy parameters (Guo et al., 2021) or dense rewards (Liu et al., 2023) for each frame, which limits applicability to unlabeled video demonstrations. More importantly, these approaches ignore the rich visual information from video demonstrations, which may potentially provide critical cues for keyframe identification. The recent development of pre-trained robotics representations, such as R3M (Nair et al., 2022b), VIP (Ma et al., 2022), LIV (Ma et al., 2023), have shown promise in capturing temporal task progress for goal-oriented behaviors, acquiring well-behaved embedding distances that often monotonically align with goal image within a short atomic task. An intuitive usage is to identify the keyframe between subtask switches with the abrupt changes in visual embedding distances for long-horizon demonstrations. However, as shown in Figure 1, LIV-V rarely manifests noticeable peaks and struggles to perform demarcating boundaries between constituent subtasks. A recent work by UVD (Zhang et al., 2023) designed greedy heuristics methods upon these pretrained visual features to capture the phase peaks for keyframe identification. However, the manually defined heuristics rules are sensitive to hyperparameters including peak detection tolerance or smoothness window length, which will lead to over-identification or mis-identification. The key limitation lies in that relying solely on visual embedding distance without language grounding cannot fully extract the semantic information within the demonstrations, leading to the identified keyframes being inaccurate and uninterpretable.

To alleviate the above limitation, we propose a unified **K**eyframe **I**dentifier and **S**kill **A**notator (KISA) that leverage pretrained visual-language representations to achieve precise and interpretable decomposition of untrimmed demonstrations. Specifically, KISA measures the similarity score between visual embeddings of individual frames and language embeddings of skill libraries for each frame, the highest score serves as the confidence score for the keyframe. Intuitively, the frame with a peak confidence score has a strong relevance with a specific skill and should be identified as the keyframe with the corresponding skill. However, LIV-L in Figure 1 shows that no frames stand out significantly as keyframes. The core dilemma lies in that these static frame-level visual representations lack dynamic action recognition, leading the inaccurate similarity with language embedding from skills. For this, we design a temporal enhancement module on top of the pretrained visual representations to incorporate historical frames and capture long-range skill dynamics beyond myopic frames. The temporal-enhanced representation with semantic action recognition can expand isolated frame-level representation to video-level representation, bridging the gap between static image and dynamic video. Then we design history-aware contrastive learning to equip KISA the capability to align video-level representation from keyframe with language representation from skills. Additionally, we design an explicit monotonic distance loss to capture the skill-aware progress in the demonstrations, preventing video representation of frames labeled as the same skill from being indiscriminately homogenized and highly identical. In this way, the confidence score calculated by video representation in KISA can exhibit clear peaks near keyframes and monotonic trend within a sub-video-chunk, showed in Figure 1, demonstrating the higher *confidence* as keyframe and *completeness* for the corresponding skill. The contributions of this work are as follows:

- **Accuracy and Interpretability**: we propose a unified **K**eyframe **I**dentifier and **S**kill **A**notator (KISA) that leverages pretrained robotics representations to achieves much more precise and interpretable decomposition of untrimmed demonstrations than competitive baselines.
- **Flexibility and Generalizability**: We design a simple yet effective temporal enhanced module that can flexibly equip any existing pre-trained representations with video-level understanding capability, which enjoys robust generalization across varied object placements, skill compositions, and cross-embodiment transfer.
- **Effectiveness and Broad Applicability**: KISA can serve as a reliable tool to unleash scalable keyframes and skill annotation to facilitate efficient policy learning from fine-grained decomposed demonstrations.

2. Related Works

2.1. Long-horizon Manipulation Tasks in Robotics

Long-horizon robotics demonstrations often encompass multiple implicit subtasks or skills (Pertsch et al., 2020; Huang et al., 2023). Directly imitation learning from long-horizon demonstrations in an end-to-end manner is challenging, due to the compounding errors of policy learning without intermediate keyframe guidance and dense skill supervision (Mandlekar et al., 2020; Jang et al., 2022; Ni et al., 2023). Some recent methods (Wang et al., 2023; Garg et al., 2022) leverage a hierarchical framework that focuses on learning the high-level predicted skills and low-level parameterized skill conditioned policy generates actions. However, obtaining demonstrations with explicit keyframe boundaries and skill annotations is still difficult. While some methods (Nair et al., 2022a; Lynch & Sermanet, 2020; Lynch et al., 2023) have explored crowd-sourced annotations, but struggle to scale across large amounts of demonstrations for the cost. An alternative is to directly discover latent reusable skills from unlabeled demonstrations via clustering, as in Garg et al. (2022) and Xu et al. (2023), but the learned latent skill variables do not have clear semantics alignment with human-interpretable skills with language description. For this, we propose KISA to achieve automatic, scalable,

and semantically meaningful keyframe identification and skill annotation from unlabeled demonstrations, which can facilitate policy learning for complex manipulation tasks.

2.2. Keyframe Identification in Robotics Demonstration

Unsupervised keyframe detection techniques like spectral clustering (Potapov et al., 2014), KTS (Afham et al., 2023) originate from the computer vision domain and cluster frames based on the visual features without explicit supervision. Some works in the robotics domain focus on identifying critical states with the need for privileged information such as actions (Arjona-Medina et al., 2019) or policy parameters (Guo et al., 2021) from vector trajectories rather than videos. A recent work VideoRLCS (Liu et al., 2023) performs keyframe detection by predicting rewards and assuming frames critical to reward prediction are keyframes. These methods in robotics demonstrations rely on privileged information, such as action or dense rewards, which limits applicability to video demonstrations, especially for open-ended manipulation demonstrations in the wild or human demonstrations. More importantly, these approaches ignore the rich visual semantics of videos. A very recent work UVD (Zhang et al., 2023) exploits phase shifts in embeddings from pretrained robotic representations to identify keyframes. However, deviations in embedding distances do not strictly correspond to keyframes as subtask boundaries. By only relying on visual cues without language grounding, UVD struggles to produce semantically interpretable decompositions aligned with distinct subtasks. Our work similarly builds on top of pretrained robotic representation but further incorporates historical frames to capture action semantics and align to language representation from skills to achieve more precise and interpretable video decomposition.

2.3. Pre-trained Representations for Control

An emerging body of work in robot learning studies learning visual representations for robotics control, seeking to use pre-existing data, typically out-of-domain, to pre-train effective representations for downstream unseen robotic tasks. R3M (Nair et al., 2022b) which is also pre-trained on the Ego4D dataset and attempts to capture temporal information in the demonstrations. VIP (Ma et al., 2022) proposes a self-supervised value-based pre-training objective that is highly effective in providing both the visual reward and representation for downstream unseen robotics tasks. LIV (Ma et al., 2023) trains a multi-modal representation that implicitly encodes a universal value function for tasks specified as language or image goals. These representations have acquired well-behaved embedding distances that can progress nearly monotonically within short-horizon demonstrations. However, directly applying them to keyframe identification for complex long-horizon tasks remains difficult, due to the lack of semantic recognition from long-range skill dynamics. KISA aggregates historical observations to equip these pretrained frame-level representations with expanded receptive fields to capture the rich dynamics cues, bridging the gap between a static image and video.

3. Preliminaries

3.1. Problem Formulation

For a long-horizon video demonstration or other privileged information like reward information or action labels, we aim to decompose it into several sub-video chunks that contain separate semantics. We hope that the boundaries between these sub-video chunks are keyframes, possessing clear semantics relevance with specific skill language descriptions. Formally, given a untrimmed demonstration $V = (o_0, ..., o_T)$, the annotation can be formulated as:

$$\texttt{KISA}(V = (o_0, ..., o_T)) \rightarrow V_{annotated} := (o_{k_i}, l_i)_{i=0}^{m}$$

where $(k_0, ..., k_m)$ are indexes of m keyframes, which may vary across different video demonstrations, and $(l_0, ..., l_m)$ are corresponding language descriptions of skills.

3.2. Contrastive Learning for Representation

Contrastive learning has emerged as a popular technique for learning effective representations in an unsupervised manner, as exemplified by the popular CLIP (Radford et al., 2021) and BLIP (Li et al., 2022) models, which also serve as the foundation for our work.. The key idea behind contrastive learning is to learn a representation that pulls similar positive samples closer while pushing dissimilar negative samples apart in the embedding space. One of the most widely used contrastive learning objectives is InfoNCE (Gutmann & Hyvärinen, 2010), which is derived from the principle of noise contrastive estimation. Given an anchor point x, a distribution of positive instances x_{pos}, and negative instances x_{neg}, the InfoNCE objective follows as:

$$\min_{\phi} \mathbb{E}_{x_{\text{pos}}} \left[-\log \frac{\mathcal{S}_\phi(x, x_{\text{pos}})}{\mathbb{E}_{x_{\text{neg}}} \mathcal{S}_\phi(x, x_{\text{neg}})} \right] \tag{1}$$

where ϕ denotes the representation to be learned, and $\mathcal{S}_\phi(\cdot, \cdot)$ is a similarity function between two representations. For more details, please refer to Appendix A.

4. Method

To tackle the challenge of reliable decomposition for long-horizon demonstrations, we propose a unified **K**eyframe **I**dentifier and **S**kill **A**notator (KISA) that leverages pretrained robotics representations to achieve precise and interpretable decomposition of untrimmed demonstrations. An overall framework is illustrated in Figure 2. Building upon the pretrained robotics representation, KISA first leverages

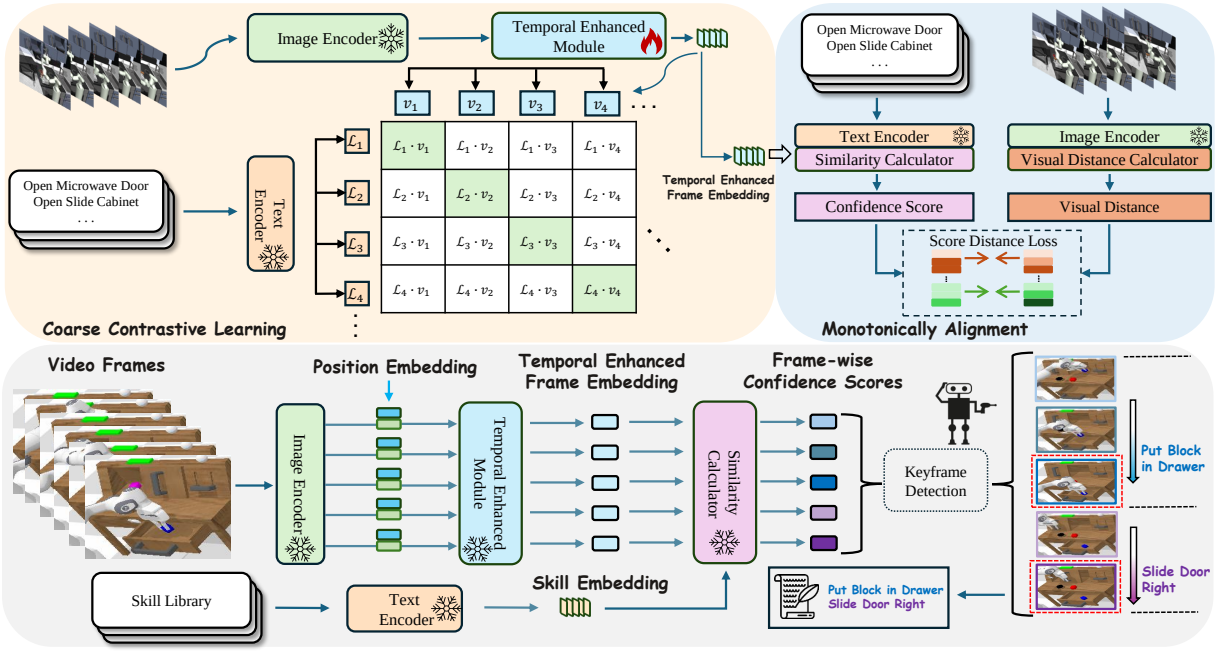

Figure 2. **Overview framework of KISA**: KISA first leverages a simple yet effective temporal enhancement module upon the pre-trained vision-language representation to obtain the video-level representation for each frame. During training, the alignment involves two branches: **Inter-skill**: we design coarse history-aware contrastive learning via constructing hard negative samples with mismatched historical contexts and incorrect skills. **Intra-skill**: we additionally fine-grained monotonic alignment to encourage the capture of skill-aware progress within the sub-task, and prevent representation collapse to highly similarity within the same skill. During the evaluation, KISA measures the similarity score between temporal-enhanced embedding and language embeddings of skill libraries for each frame. The highest score serves as the confidence score for the keyframe and the frame with the peak confidence score has strong relevance with a specific skill and should be identified as the keyframe with the corresponding skill.

a simple yet effective temporal enhancement module to incorporate historical frames to capture long-range semantic dynamics for skill prediction. Then we design history-aware contrastive learning by constructing hard negative samples with mismatched historical frames or unpaired skill labels to align the visual representation and language representation from skills. Moreover, to avoid the representation collapse to homogeneity, we propose a monotonic distance loss to encourage skill alignment predictions to exhibit monotonic trends over time to capture the skill-aware progress with more fine-grained distinguishability.

4.1. Temporal Enhanced Representation

Given a video clip $V \in \mathbb{R}^{T\times H\times W\times 3}$ of T sampled frames with H and W denotes the spatial resolution, we utilize only images as robot state information without reward or action labels attached. The key is to find a discriminative representation to make keyframes salient among all frames. A straightforward approach is to directly leverage the vision representation of vision encoder ϕ from the existing pretrained robotics representation. However, the vanilla visual representation fails to make keyframes distinguish from other frames, regardless of the metric of the visual distance embedding or the similarity between the visual embedding and skill embedding from language encoder ψ, denoted as LIV-V and LIV-L in Figure 1.

The core lies in the static frame-level representation fails to adequately capture the dynamics context across extended time horizons and multiple phases. Consider two visually similar frames from different skills in a long-horizon robotics video, shown in Figure 3. Relying solely on frame-level visual representations would induce training confusion for aligning them to distinct skills, as their isolated representations are nearly identical. Skills possess action semantics while static frames reflect state - the same frame of a robotics arm touching an oven cannot disambiguate between "open the drawer" or "close the drawer" without considering preceding history. Moreover, directly mapping frame representations to skills can overfit, for example falsely recognizing any frame with an arm near a closed oven as a keyframe of "open/close the drawer". The same drawer-touching state may coincide in a sub-video of "light the bulb", but should not serve as a keyframe, which will lead over-detection.

Incorporating context and action semantics is key for robust keyframe identification and alignment. Inspired by this, we propose a simple yet effective temporal-enhanced module on top of pretrained visual representations. By aggregating historical frames $h_i = \{o_0, \ldots, o_{i-1}\}$, we provide ex-

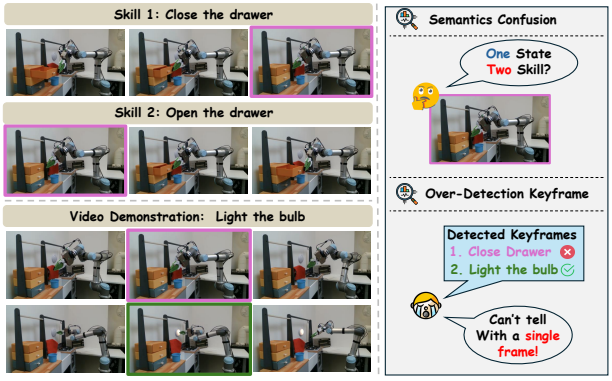

Figure 3. **Motivation Example:** Without historical context for semantic action cognition, two visually similar frames from different skills can confuse the alignment. Furthermore, the representation might overfit the alignment between isolated frames and skills, leading to over-identification or false-identification.

panded receptive fields to capture skill transitions over time. Formally, our video encoder is:

$$v_i = \Phi(h_i, o_i) = \Phi_{\text{TEMP}}(\{\phi(o_0), \ldots, \phi(o_{i-1}), \phi(o_i)\}) \tag{2}$$

where Φ_{TEMP} refers to a temporal-enhanced module, which is a multi-layer Transformer Encoder, consisting of Multi-head Self-attention, Layer Norm, and MLP blocks, applied identically across time steps. To indicate the temporal order, we also add temporal positional encoding onto image features explicitly. This simple enhancement equips static frame-level representations with substantially improved video understanding skills with minimal computational overhead, providing a lightweight plug-and-play solution for bridging the image-video gap without expensive video pretraining.

4.2. History-aware Contrastive Training

With vision representations enriched by historical contextual frames, the next step is the alignment with language representation from skills. Specifically, we follow a widely-used contrastive learning technique to fine-tune the pre-trained robotics representation by learning a cross-modal joint embedding, aligning vision modalities from video and language modalities from skills using metric learning on paired positive and unpaired negative samples. To compute the InfoNCE objective for contrastive alignment, we define $\mathcal{S}(o, h, \ell) = cos(\Phi(h_i, o_i), \mathcal{L}(\ell))$ with the cosine similarity cos. We sample positive data $\{o^+, h^+, \ell^+\} \sim \mathcal{D}$ by selecting the pairwise current frame, historical frames, and skill annotations from the same video demonstration. Indeed, the key to efficient contrastive training lies in the design of negative examples. Hard or challenging negative examples can significantly enhance the robustness of representation learning, as highlighted in Yang et al. (2021).

To this end, instead of merely treating video embedding as traditional static image embedding and skill embedding for negative samples, we propose a more enriched and fine-grained design aimed to create more challenging negative samples. Specifically, we devise three types of negatives to enhance alignment: 1) ***Incorrect Skill Alignments***: we sample negative examples denoted as $\{o^+, h^+, l^-\}$ by altering the pairwise skill to the skill language annotation from another random demonstration. 2) ***Disjoint Frame-History Compositions***: we stitch the pairwise keyframe and skill with mismatched historical frames randomly selected from other demonstrations, thereby constructing negative samples $\{o^+, h^-, l^+\}$. This prevents overfitting between isolated frames and skills. 3) ***Semantic Reversals via Video Inversion***: we keep the set of frames the same but reverse the video order, which can significantly change its semantic interpretation. For example, a video showing the action of "opening the door" when reversed, semantically becomes "closing the door". By temporally reversing the entire sub-video chunk, we construct more challenging negative samples counterfactually, depicted as $\{(o^+, h^+)^{rev}, l^+\}$. Formally, the contrastive loss can be denoted as:

$$\begin{aligned}
\mathcal{L}_{\text{video}} &= -\log \frac{e^{\mathcal{C}(o^+, h^+, \ell^+)}}{e^{\mathcal{C}(o^+, h^+, \ell^+)} + \sum_{j=1}^{k} e^{\mathcal{C}(o^+, h^+, \ell_j^-)}} \\
\mathcal{L}_{\text{history}} &= -\log \frac{e^{\mathcal{C}(o^+, h^+, \ell^+)}}{e^{\mathcal{C}(o^+, h^+, \ell^+)} + \sum_{z=1}^{k} e^{\mathcal{C}(o^+, h_z^-, \ell^+)}} \\
\mathcal{L}_{\text{reverse}} &= -\log \frac{e^{\mathcal{C}(o^+, h^+, \ell^+)}}{e^{\mathcal{C}(o^+, h^+, \ell^+)} + \sum_{w=1}^{k} e^{\mathcal{C}(\{(o^+, h^+)_w^{rev}, l^+\})}} \\
\mathcal{L}_{\text{contrastive}} &= \mathcal{L}_{\text{video}} + \mathcal{L}_{\text{history}} + \mathcal{L}_{\text{reverse}}
\end{aligned} \tag{3}$$

where k is the number of negative samples. The core insight is to enhance the semantic alignment between skill and video embedding with historical frames and avoid overfitting with isolated keyframes. Meanwhile, we aim to strengthen the connection between the historical context and the current frame, without neglecting the temporal relationship of the historical context. Overall, the history-aware contrastive learning fine-tunes the representation at the inter-skill level, encouraging temporal-enhanced representation to distinguish from diverse skills.

4.3. Fine-grained Monotonic Alignment

The coarse contrastive learning non-discriminatively aligns the video representation of all frames within each sub-video to the same language representation from the same corresponding skill. However, this risks potential representation collapse, which means representation of frames labeled as the same skill can become indiscriminately homogenized and highly identical. To encourage more fine-grained distinguishability, we propose an additional monotonic alignment objective: the similarity between frame representations and skill embeddings for clips of the same sub-task should exhibit a monotonically increasing trend over time. There are two key insights motivating this design: Firstly, as integrated

representations accumulate richer historical observations towards the latter frames within a sub-video, *confidence* in predicting the associated skill should objectively increase. Secondly, within a sub-video, frames temporally closer to the terminal keyframe should correspond to more advanced *completion* stages of the depicted skill.

Indeed, the visual distance within the sub-video separated by keyframes should exhibit an overall monotonicity naturally. For training, we can further compute the visual similarity distance $\mathcal{D}(\phi(o_i), \phi(o_i^{\mathcal{K}}))$ for each frame o_i, where $o_i^{\mathcal{K}}$ are the terminal keyframe of sub-video chunk which o_i belongs to. Within the sub-video segmented from the groundtruth keyframes, as a free lunch without additional human annotation cost for additional supervision signal to encourage a finer-grained inter-skill representation and alignment. Formally, the score distance loss is defined as the mean squared error between the skill alignment score and ground truth visual distance score at each time step:

$$\mathcal{L}_{score} = \frac{1}{T}\sum_{i=1}^{T} ||S(\Phi(o_i, h_i); \psi(\ell_i)) - \mathcal{D}(\phi(o_i), \phi(o_i^{\mathcal{K}}))|| \quad (4)$$

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{contrastive}} + \alpha \cdot \mathcal{L}_{\text{score}} \quad (5)$$

where α is the coefficient weight to balance off these losses between the coarse inter-skill contrastive loss and the fine-grained intra-skill monotonicity alignment. To avoid the potential overfit to absolute values of distance, we normalize the distance scores to relative difference, which can be viewed as a regularization term to contrastive learning loss from another perspective.

5. Experiments

We conduct experiments on various benchmarks to evaluate the proposed KISA. We aim to empirically answer the following questions: 1) Can KISA achieve better **accuracy** and **interpretable** skill alignment compared to other competitive baselines? 2) Can KISA enjoy the robust zero-shot **generalization** ability across objects, compositional or even cross embodiments tasks? 3) Can KISA be a **flexible** framework to incorporate with any pretrained robotics representations? 4) Can the long-horizon unlabeled demonstration with explicit keyframe and skill annotations **reliable** and **effective** be enough to facilitate policy learning?

5.1. Experiment Setup

Benchmark To provide comprehensive evaluations, we conduct experiments across three typical long-horizon manipulation environments covering diverse skills.

- **Maniskill2** (Gu et al., 2022): a unified benchmark for generalizable manipulation skills, including manipulation task families with various color schemes and customizable configurations for manipulation scenes and objects.
- **CALVIN** (Mees et al., 2022b): a benchmark for long-horizon language-conditioned manipulation, with a Franka robot arm required to complete a chain of several language instructions with various manipulable objects.
- **Franka Kitchen** (Gupta et al., 2019): is a simulated kitchen environment in which a 7-DoF Franka robot is tasked with manipulating common household kitchen objects to pre-specified configurations.

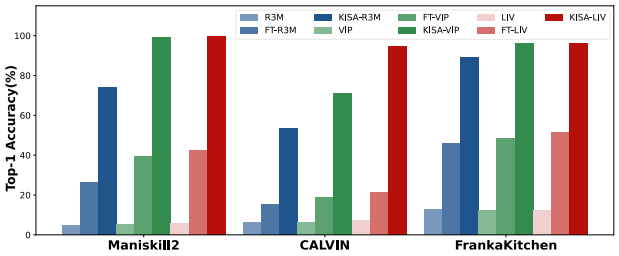

Figure 4. **The comparisons of skill annotation.** On all three classic robotics representations, the skill annotation accuracy of the 'KISA-' version outperforms that of the 'FT-' version, which follows the vanilla alignment technique with static representation.

Evaluation Metrics The evaluation metrics we use in the experiments cover two aspects.

- *Keyframe:* We use keyframe number errors, mean absolute error (MAE) and F1-score to evaluate the accuracy of identified keyframes.
- *Skill:* We use Top-1 accuracy to evaluate if the predicted skill category with maximum model confidence matches the true label. For details, please refer to Appendix C.3.

5.2. Baslines

- **Kernel Temporal Segmentation (KTS)** (Potapov et al., 2014; Afham et al., 2023): the typical unsupervised keyframe identification method from CV domain, which applies an adaptive kernel density estimator to identify dissimilar consecutive frames as boundary or keyframes.
- **VideoRLCS** (Liu et al., 2023): assumes that frames critical to reward prediction are keyframes and perform keyframe detection based on the reward predictor trained with the supervision of groundtruth reward.
- **UVD** (Zhang et al., 2023): the concurrent work for keyframe identification, which designs a heuristic rule to recursively detect peak value of representation distance as keyframes, based on pre-trained visual representations.
- **Pretrained Robotics Representations:** including R3M (Nair et al., 2022b), VIP (Ma et al., 2022) and LIV (Ma et al., 2023), which re-purposed for keyframe identification via the matching scores of frames and skills.

5.3. The Accuracy of Keyframes and Skills Annotation

Here we conduct a comprehensive evaluation between baselines across two dimensions as main quatative results.

Precision in identified keyframe As evidenced in Table 1, KISA significantly outperforms all baselines across three metrics, highlighting its superior ability in keyframe

Table 1. **The evaluation results of keyframe identification.** We evaluate several baselines on the collected long-horizon demonstrations dataset with groundtruth skill labels three typical manipulation environments and report the mean and variance across 5 seeds.

	Maniskill			CALVIN			FrankaKitchen		
Model	**Number Error**↓	**F1 Score** ↑	**MAE** ↓	**Number Error**↓	**F1 Score** ↑	**MAE** ↓	**Number Error**↓	**F1 Score** ↑	**MAE** ↓
VideoRLCS	10.1 ± 2.1	$15.2 \pm 0.3\%$	34.4 ± 0.4	9.5 ± 2.4	$15.4 \pm 0.5\%$	54.8 ± 1.0	0.6 ± 0.0	$5.5 \pm 0.8\%$	39.3 ± 0.7
KTS	5.1 ± 0.0	$15.7 \pm 4.0\%$	24.2 ± 6.8	0.9 ± 0.6	$20.2 \pm 0.7\%$	50.9 ± 7.4	0.5 ± 0.2	$13.8 \pm 3.4\%$	35.6 ± 6.9
R3M	5.0 ± 0.2	$17.1 \pm 0.8\%$	38.2 ± 3.0	5.4 ± 0.2	$21.1 \pm 1.3\%$	63.2 ± 1.1	1.0 ± 0.1	$53.7 \pm 0.8\%$	30.4 ± 0.1
VIP	4.0 ± 0.2	$31.7 \pm 2.5\%$	24.8 ± 1.4	4.6 ± 0.2	$24.3 \pm 1.6\%$	63.4 ± 1.2	0.9 ± 0.1	$57.2 \pm 1.1\%$	31.4 ± 0.1
LIV	3.9 ± 0.4	$30.3 \pm 2.2\%$	23.9 ± 1.4	5.6 ± 0.1	$25.9 \pm 1.1\%$	61.7 ± 1.5	1.4 ± 0.0	$64.2 \pm 0.8\%$	30.7 ± 0.1
UVD	0.7 ± 0.1	$40.2 \pm 1.1\%$	20.3 ± 0.1	0.8 ± 0.1	$36.9 \pm 0.5\%$	40.6 ± 1.7	0.6 ± 0.1	$64.8 \pm 2.4\%$	31.1 ± 0.2
KISA	$\mathbf{0.0 \pm 0.0}$	$\mathbf{99.7 \pm 0.2\%}$	$\mathbf{0.2 \pm 0.1}$	$\mathbf{0.1 \pm 0.1}$	$\mathbf{85.2 \pm 0.9\%}$	$\mathbf{11.2 \pm 2.4}$	$\mathbf{0.0 \pm 0.0}$	$\mathbf{98.7 \pm 0.6\%}$	$\mathbf{0.4 \pm 0.0}$

identification. Reward-driven keyframe extraction methods like VideoRLCS perform the worst among baselines. A potential reason is that the importance of the keyframe for reward prediction decreases when the horizon extends and the visual representation is not exploited. On the other hand, unsupervised methods like KTS without additional training, which solely relies on the similarity of visual embeddings for clustering, can already achieve a higher accuracy than VideoRLCS. This provides empirical insight that the visual information has great potential for keyframe identification. Moreover, we directly utilize pretrained robotics representations such as R3M, VIP, and LIV to assess the ability to identify keyframes. The accuracy improves in the order of R3M, VIP, and LIV, with the latter enjoying more fine-grained representation properties. Based on these representations, UVD improves the accuracy of keyframe identification by manually designing heuristic rules compared to directly tracking the peaks of the original representation distance curves. But UVD still underforms than KISA with language representation to facilitate the identification, the reported results are all based on LIV as backbone.

Accuracy of skill annotation Moreover, we evaluate the accuracy of skill annotation based on the identified keyframe, shown in Figure 4. The methods including KTS and UVD do not support skill annotation, so we turn to repurposing the pretrained robotics representations used in UVD as baselines for comparison. Specifically, we leverage pretrained visual representations to compute frame-wise similarity scores with language representations from skill libraries and annotate the frame with the skill that has the highest confidence score. Furthermore, we fine-tune these representations by directly aligning the vision and skills language representations on a frame-wise basis using vanilla contrastive learning without challenging negative samples we specially designed, referred to as the 'FT-' version, with the same amount of data and training steps as 'KISA-' version. The results indicate that vanilla fine-tuning on the frame-level static representation can improve the accuracy of skill annotation, compared to the original representation without fine-tuning. For this, in this paper, we default to the 'FT-' version for the three typical representation backbones, without expressly mentioning 'FT', for the sake of simplicity. However, the 'FT-' version remains notably inferior to KISA, highlighting the critical role of proposed temporal-enhanced representation and our specially designed history-aware contrastive learning.

Table 2. **Zero-shot Results.** Comparisons between methods on three zero-shot levels across objects, composability, embodiments.

	Maniskill2 (L1)		CALVIN (L2)		RealKitchen (L3)	
Model	**F1 Score** ↑	**MAE** ↓	**F1 Score** ↑	**MAE** ↓	**F1 Score** ↑	**MAE** ↓
KTS	$12.3 \pm 2.7\%$	25.2 ± 5.3	$20.2 \pm 0.7\%$	54.9 ± 7.4	$11.9 \pm 5.4\%$	44.9 ± 21.4
R3M	$16.8 \pm 1.4\%$	38.8 ± 2.5	$20.9 \pm 1.1\%$	63.4 ± 1.2	$20.7 \pm 0.5\%$	44.9 ± 21.4
VIP	$30.6 \pm 1.8\%$	23.3 ± 1.8	$23.6 \pm 1.9\%$	63.2 ± 1.1	$20.5 \pm 17.8\%$	34.8 ± 11.0
LIV	$30.2 \pm 1.7\%$	23.6 ± 1.0	$25.4 \pm 1.3\%$	63.2 ± 1.1	$21.7 \pm 20.9\%$	44.9 ± 21.4
UVD	$39.2 \pm 1.1\%$	21.5 ± 0.2	$36.9 \pm 0.5\%$	40.6 ± 1.7	$26.3 \pm 11.9\%$	30.2 ± 6.2
KISA	$\mathbf{80.7 \pm 0.9\%}$	$\mathbf{6.4 \pm 0.7}$	$\mathbf{89.4 \pm 1.8\%}$	$\mathbf{14.2 \pm 0.9}$	$\mathbf{40.7 \pm 14.8\%}$	$\mathbf{27.8 \pm 5.0}$

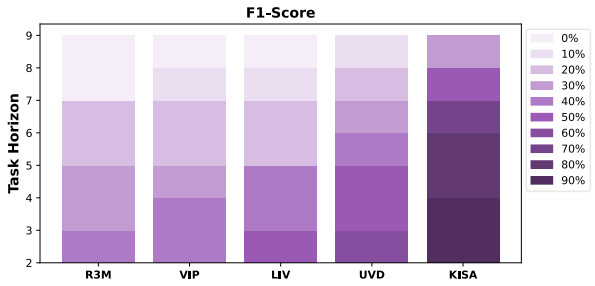

Figure 5. **Combinatorial Generalization (L2) on CALVIN.** The detailed results on tasks with various sub-task horizons.

5.4. The Zero-shot Generalization Ability

To systematically examine generalization capacities, we establish a 3-level protocol evaluating models on progressively more challenging unseen distributions without additional training: (L1) ***Object Generalization***: We evaluate L1 generalization on rich manipulation scenes from Maniskill2, with diverse object colors, shapes, numbers, and placements. (L2) ***Combinatorial Generalization***: We evaluate generalization on CALVIN for novel skill compositions that never appear in training and also examine the accuracy of decomposition in longer horizon cases. (L3) ***Embodiment Generalization***: Based on the long-horizon kitchen demonstration datasets (Xu et al., 2023) including real human or robots, we evaluate whether the representation can generalize across embodiments from simulator to real. The overall results in Table 2 demonstrate that KISA shows great performance

gain against all baselines in three levels. As the horizon extends, the combinatorial generalization ability of the KISA remains significantly stronger than baselines, shown in Figure 5. This is primarily due to KISA's ability to integrate historical frames to expand receptive fields, which allows it to better capture long-range skill dynamics beyond isolated frames. Rather than overfitting to superficial environmental details, KISA focuses on learning on core semantics - reducing dependence on specific configurations or scenes. This also explains why static representations like LIV are prone to overfit to the mapping to static frames to skill, leading to over-identification or wrong-identification of keyframes without the reasonable semantics shown in Figure 6, especially when migrating across robot embodiments.

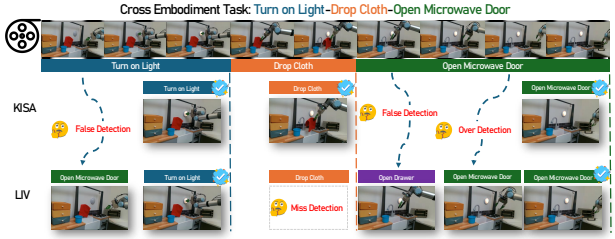

Figure 6. **Cross Embodiment Generalization (L3).** The illustrating example of LIV and KISA for zero-shot generalization from simulators on long-horizon real robotics demonstration datasets.

5.5. The Flexibility for Pre-trained Representations

KISA equips static frame-level representations with video-level understanding capability, which is flexible to incorporate with any existing visual representation backbone. We conduct comprehensive evaluations across R3M, VIP, and LIV through ablations studies of the proposed temporal enhancement module, history-aware contrastive learning, and monotonicity alignment components respectively, shown in Table 3. We observe the ablation performance trends are similar across different representation backbones and KISA-LIV performs better than KISA-R3M and KISA-VIP. The potential reason is LIV has already accomplished a vision-language alignment at the static frame level, which implicitly provides a certain benefit for video-level and skill alignment in KISA. When we ablate the temporal enhancement module and retain contrastive learning and monotonicity alignment, it results in the most significant performance drop. This reconfirms the fundamental importance of considering temporal information in action semantic recognition for keyframe identification. Additionally, we replaced history-aware contrastive learning with vanilla contrastive learning and found a performance drop, demonstrating that constructing challenging negative samples with confusing historical frames or semantics is crucial for efficient video-level representation learning. We empirically find the monotonicity alignment can further enhance the finer-grained alignment between keyframe and skills, shown in Figure 7. Overall, the results highlight that the proposed modules in KISA are all *effective* and *representation-agnostic*.

Table 3. Quantitative comparisons of Top-1 Accuracy with different robotics representation as backbones on all three environments.

Methodology	Maniskill2	CALVIN	FrankaKitchen
KISA-R3M	$\mathbf{71.8 \pm 3.9\%}$	$\mathbf{53.6 \pm 1.1\%}$	$\mathbf{88.9 \pm 0.6\%}$
- w/o monotonic align	$63.0 \pm 3.2\ \% \downarrow$	$50.1 \pm 0.8\ \% \downarrow$	$81.5 \pm 1.0\% \downarrow$
- w/o historical contrastive	$41.3 \pm 2.2\ \% \downarrow$	$45.7 \pm 2.3\ \% \downarrow$	$76.6 \pm 0.5\ \% \downarrow$
- w/o temporal enhance	$23.1 \pm 0.5\ \% \downarrow$	$24.0 \pm 2.2\ \% \downarrow$	$21.9 \pm 0.8\ \% \downarrow$
KISA-VIP	$\mathbf{99.6 \pm 0.1\%}$	$\mathbf{70.9 \pm 2.7\%}$	$\mathbf{96.4 \pm 0.3\%}$
- w/o monotonic align	$88.9 \pm 0.5\ \% \downarrow$	$64.0 \pm 0.4\ \% \downarrow$	$90.1 \pm 0.3\% \downarrow$
- w/o historical contrastive	$58.9 \pm 1.5\ \% \downarrow$	$53.1 \pm 1.5\ \% \downarrow$	$81.8 \pm 0.9\% \downarrow$
- w/o temporal enhance	$24.0 \pm 0.3\ \% \downarrow$	$23.4 \pm 0.8\ \% \downarrow$	$21.4 \pm 0.8\% \downarrow$
KISA-LIV	$\mathbf{99.2 \pm 0.1\%}$	$\mathbf{94.7 \pm 1.1\%}$	$\mathbf{96.1 \pm 0.2\%}$
- w/o monotonic align	$90.2 \pm 0.3\ \% \downarrow$	$82.1 \pm 0.4\ \% \downarrow$	$89.1 \pm 0.7\% \downarrow$
- w/o historical contrastive	$59.1 \pm 1.9\ \% \downarrow$	$58.1 \pm 1.3\ \% \downarrow$	$73.7 \pm 0.4\% \downarrow$
- w/o temporal enhance	$22.0 \pm 1.6\ \% \downarrow$	$25.0 \pm 1.6\ \% \downarrow$	$19.0 \pm 0.3\% \downarrow$

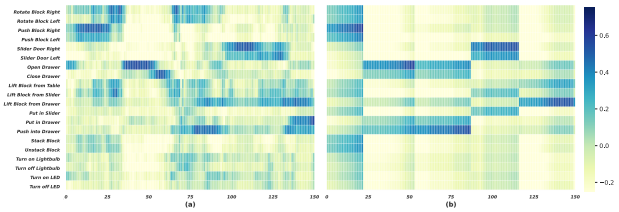

Figure 7. **The heatmap between skills and frames from a long-horizon demonstration example on CALVIN.** The comparisons between KISA w/o monotonic alignment (left) and KISA (right).

5.6. The Effectiveness for Policy Learning

We aim to explore whether demonstrations labeled with precise keyframes and corresponding skill annotations by KISA can aid in long-horizon policy learning. We selected several baselines for comparison, including Language-Conditioned Behavior Cloning (LCBC) which learns directly from demonstrations, conditioned on general language instruction in an end-to-end manner. We also compare with LISA (Garg et al., 2022), a hierarchical framework that first discovers implicit skills and then learns to combine skills for complex tasks. The results in 4 show that LCBC underperformed compared to LISA due to the lack of hierarchical structure. Based on LISA, we compare when provided with privileged information including explicit keyframes and skill annotations to avoid re-discovering skills, while retaining low-level skill-conditioned policy learning for fair comparisons. Unsurprisingly, LISA+KISA achieves a significant performance improvement with annotated demonstrations, particularly in tasks with longer horizons. Skills discovered by LISA lack clear semantics as they are represented in latent code and cannot establish a one-to-one relationship with action primitives. In contrast, skills annotated by KISA are more accurate and interpretable, thus reducing the learning burden. Furthermore, keyframes can serve as accurate boundaries for sub-trajectories to prevent overlap, facilitating the skill-conditioned policy learning in low-level. However, we also noticed that annotation with low quality has no benefit for policy learning, considering the performance drop with demonstrations annotated by LIV.

Table 4. **Success rates on CALVIN** including LCBC, LISA, and the variation with demonstration annotated by LIV and KISA.

Instruction Num	Method			
	LCBC	LISA	LISA+LIV	LISA+KISA
1	$34.2\% \pm 4.2$	$55.4\% \pm 2.7$	$51.2\% \pm 2.6$	$\mathbf{82.5\% \pm 1.6}$
2	$6.8\% \pm 1.4$	$41.7\% \pm 1.3$	$33.9\% \pm 1.7$	$\mathbf{61.0\% \pm 1.8}$
3	$1.1\% \pm 0.3$	$26.2\% \pm 2.0$	$23.1\% \pm 2.5$	$\mathbf{47.8\% \pm 2.3}$
4	$0.2\% \pm 0.0$	$15.4\% \pm 1.7$	$14.9\% \pm 1.2$	$\mathbf{30.1\% \pm 1.7}$
5	$0.1\% \pm 0.0$	$9.3\% \pm 0.8$	$7.5\% \pm 1.1$	$\mathbf{17.6\% \pm 1.0}$
Avg Len	0.4 ± 0.1	2.2 ± 0.1	1.9 ± 0.2	$\mathbf{2.7 \pm 0.2}$

5.7. The Effect of Historical Frames

To investigate the effect of different historical frame settings, we conduct detailed ablation studies. Due to the varying lengths of skill demonstrations, fixed-size sliding windows may not comprehensively cover every complete demonstration. We evaluate multiple fixed window sizes that cover different proportions relative to the average sub-skill length. The results in Table 5 show that larger window sizes generally lead to higher keyframe identification accuracy, suggesting that incorporating more historical frames within intra-skill sub-segments enhances the understanding of skill temporal information and action semantics. We further investigate the impact of the scope of historical context by comparing the use of the entire history versus only the past history within the current skill segment. The "Past-His-within-Segment" variant, which filters out the history from previous skill segments, achieves comparable performance to the whole history version. Historical frames outside the current skill segment do not significantly contribute to keyframe identification, corroborating the marginal performance gain observed with extended fixed windows beyond the average demonstration length.

Table 5. **Quantitative comparisons of Top-1 Accuracy** with varies historical lengths.

Methodology	F1 score ↑	MAE ↓	Top-1 Accuracy ↑
KISA	$\mathbf{98.7 \pm 0.6\%}$	$\mathbf{0.4 \pm 0.0}$	$\mathbf{96.4 \pm 0.3\%}$
- Fix His Len 8	$41.1 \pm 2.7\ \%$	18.6 ± 1.0	$72.2 \pm 2.3\%$
- Fix His Len 16	$63.6 \pm 2.5\ \%$	16.5 ± 0.7	$83.1 \pm 1.8\%$
- Fix His Len 25	$77.9 \pm 1.3\ \%$	8.0 ± 0.6	$87.3 \pm 2.1\%$
- Fix His Len 32	$87.1 \pm 1.6\ \%$	7.1 ± 1.4	$89.4 \pm 2.8\%$
- Fix His Len 40	$87.4 \pm 1.0\ \%$	7.8 ± 0.6	$92.3 \pm 1.2\%$
- Past_His_within_Segm	$96.3 \pm 0.4\ \%$	0.7 ± 0.2	$95.0 \pm 0.0\%$

5.8. The Scalability of Framework

To investigate the scalability and generalization of our proposed approach, we conducted additional experiments by combining the demonstration datasets across all three domains for joint fine-tuning. We found that jointly trained model exhibited slightly worse performance compared to the separate domain-specific models, but still outperformed other baselines by a significant margin, as shown in Figure 8. The potential reason is that different datasets may contain similar skills, but the visual demonstrations of those skills could exhibit vastly different styles across domains, leading to conflicts and confusion during joint training. However, the results show that increasing the model size can effectively mitigate this issue. Moreover, model size scaling demonstrates KISA's scalability and potential for handling diverse and heterogeneous demonstrations, towards scalable internet-scale demonstrations.

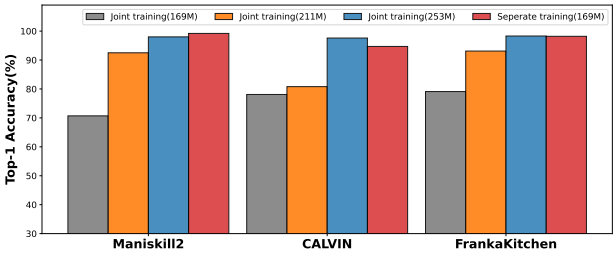

Figure 8. **The comparisons of accuracy of keyframe identification with different training manner and model size.** KISA show scalability and potential for handling diverse and heterogeneous demonstrations, towards scalable internet-scale demonstrations.

6. Conclusion

In this paper, we introduce KISA, a unified framework to achieve accurate keyframe identification and skills annotation for long-horizon manipulation demonstrations. We propose a simple yet effective temporal enhanced module that can flexibly equip any existing pre-trained representations with expanded receptive fields to capture long-range semantic dynamics, bridging the gap between static frame-level representation and video-level understanding. We further design coarse contrastive learning and fine-grained monotonic encouragement to enhance the alignment between keyframes and skills. The experiment results demonstrate that KISA achieves more accurate and interpretable keyframe identification than competitive baselines and enjoys the robust zero-shot generalization ability. Furthermore, demonstrations with accurate keyframes and interpretable skills annotated by KISA can significantly facilitate policy learning. One limitation of KISA is that the current skill annotation is to retrieve from the skill library and lacks the ability of open-vocabulary generation for skill annotation. A potential solution is to leverage the multi-modal large language modelto equip with the generation ability for more diverse skills. We believe KISA, as both a reliable annotation tool and a source of fine-grained video-level data representation, could provide the robotics research community with valuable insights and conveniences.

Acknowledgement

This work is supported by the National Natural Science Foundation of China (Grant Nos. 92370132, 62106172), the National Key R&D Program of China (Grant No. 2022ZD0116402) and the Xiaomi Young Talents Program of Xiaomi Foundation.

Impact Statement

This paper presents work whose goal is to advance the field of Machine learning. There are many potential social consequences of our work, none of which we feel must be specifically highlighted here.

References

Afham, M., Shukla, S. N., Poursaeed, O., Zhang, P., Shah, A., and Lim, S. Revisiting kernel temporal segmentation as an adaptive tokenizer for long-form video understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1189–1194, 2023.

Arjona-Medina, J. A., Gillhofer, M., Widrich, M., Unterthiner, T., Brandstetter, J., and Hochreiter, S. Rudder: Return decomposition for delayed rewards. *Advances in Neural Information Processing Systems*, 32, 2019.

Garg, D., Vaidyanath, S., Kim, K., Song, J., and Ermon, S. Lisa: Learning interpretable skill abstractions from language. *Advances in Neural Information Processing Systems*, 35:21711–21724, 2022.

Gu, J., Xiang, F., Li, X., Ling, Z., Liu, X., Mu, T., Tang, Y., Tao, S., Wei, X., Yao, Y., et al. Maniskill2: A unified benchmark for generalizable manipulation skills. In *The Eleventh International Conference on Learning Representations*, 2022.

Guo, W., Wu, X., Khan, U., and Xing, X. Edge: Explaining deep reinforcement learning policies. *Advances in Neural Information Processing Systems*, 34:12222–12236, 2021.

Gupta, A., Kumar, V., Lynch, C., Levine, S., and Hausman, K. Relay policy learning: Solving long-horizon tasks via imitation and reinforcement learning. *arXiv preprint arXiv:1910.11956*, 2019.

Gutmann, M. and Hyvärinen, A. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 297–304. JMLR Workshop and Conference Proceedings, 2010.

Huang, W., Wang, C., Zhang, R., Li, Y., Wu, J., and Fei-Fei, L. Voxposer: Composable 3d value maps for robotic manipulation with language models. *arXiv preprint arXiv:2307.05973*, 2023.

Jang, E., Irpan, A., Khansari, M., Kappler, D., Ebert, F., Lynch, C., Levine, S., and Finn, C. Bc-z: Zero-shot task generalization with robotic imitation learning. In *Conference on Robot Learning*, pp. 991–1002. PMLR, 2022.

Ju, C., Han, T., Zheng, K., Zhang, Y., and Xie, W. Prompting visual-language models for efficient video understanding. In *European Conference on Computer Vision*, pp. 105–124. Springer, 2022.

Langley, P. Crafting papers on machine learning. In Langley, P. (ed.), *Proceedings of the 17th International Conference on Machine Learning (ICML 2000)*, pp. 1207–1216, Stanford, CA, 2000. Morgan Kaufmann.

Li, J., Li, D., Xiong, C., and Hoi, S. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pp. 12888–12900. PMLR, 2022.

Li, P., Tang, H., HAO, J., ZHENG, Y., Fu, X., and Meng, Z. ERL-re$ˆ2$: Efficient evolutionary reinforcement learning with shared state representation and individual policy representation. In *International Conference on Learning Representations*, 2023.

Liu, H., Zhuge, M., Li, B., Wang, Y., Faccio, F., Ghanem, B., and Schmidhuber, J. Learning to identify critical states for reinforcement learning from videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1955–1965, 2023.

Luo, H., Ji, L., Zhong, M., Chen, Y., Lei, W., Duan, N., and Li, T. Clip4clip: An empirical study of clip for end to end video clip retrieval and captioning. *Neurocomputing*, 508:293–304, 2022.

Lynch, C. and Sermanet, P. Language conditioned imitation learning over unstructured data. *arXiv preprint arXiv:2005.07648*, 2020.

Lynch, C., Wahid, A., Tompson, J., Ding, T., Betker, J., Baruch, R., Armstrong, T., and Florence, P. Interactive language: Talking to robots in real time. *IEEE Robotics and Automation Letters*, 2023.

Ma, Y. J., Sodhani, S., Jayaraman, D., Bastani, O., Kumar, V., and Zhang, A. Vip: Towards universal visual reward and representation via value-implicit pre-training. *arXiv preprint arXiv:2210.00030*, 2022.

Ma, Y. J., Liang, W., Som, V., Kumar, V., Zhang, A., Bastani, O., and Jayaraman, D. Liv: Language-image representations and rewards for robotic control. *arXiv preprint arXiv:2306.00958*, 2023.

Mandlekar, A., Xu, D., Martín-Martín, R., Savarese, S., and Fei-Fei, L. Learning to generalize across long-horizon tasks from human demonstrations. *arXiv preprint arXiv:2003.06085*, 2020.

Mees, O., Hermann, L., and Burgard, W. What matters in language conditioned robotic imitation learning over unstructured data. *IEEE Robotics and Automation Letters*, 7(4):11205–11212, 2022a.

Mees, O., Hermann, L., Rosete-Beas, E., and Burgard, W. Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. *IEEE Robotics and Automation Letters (RA-L)*, 7(3):7327–7334, 2022b.

Mishra, U. A., Xue, S., Chen, Y., and Xu, D. Generative skill chaining: Long-horizon skill planning with diffusion models. In *Conference on Robot Learning*, pp. 2905–2925. PMLR, 2023.

Nair, S., Mitchell, E., Chen, K., Savarese, S., Finn, C., et al. Learning language-conditioned robot behavior from offline data and crowd-sourced annotation. In *Conference on Robot Learning*, pp. 1303–1315. PMLR, 2022a.

Nair, S., Rajeswaran, A., Kumar, V., Finn, C., and Gupta, A. R3m: A universal visual representation for robot manipulation. *arXiv preprint arXiv:2203.12601*, 2022b.

Ni, F., Hao, J., Mu, Y., Yuan, Y., Zheng, Y., Wang, B., and Liang, Z. Metadiffuser: Diffusion model as conditional planner for offline meta-rl. In *International Conference on Machine Learning*, pp. 26087–26105. PMLR, 2023.

Pertsch, K., Rybkin, O., Ebert, F., Zhou, S., Jayaraman, D., Finn, C., and Levine, S. Long-horizon visual planning with goal-conditioned hierarchical predictors. *Advances in Neural Information Processing Systems*, 33:17321–17333, 2020.

Potapov, D., Douze, M., Harchaoui, Z., and Schmid, C. Category-specific video summarization. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI 13*, pp. 540–555. Springer, 2014.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763. PMLR, 2021.

Sermanet, P., Lynch, C., Chebotar, Y., Hsu, J., Jang, E., Schaal, S., Levine, S., and Brain, G. Time-contrastive networks: Self-supervised learning from video. In *2018 IEEE international conference on robotics and automation (ICRA)*, pp. 1134–1141. IEEE, 2018.

Stepputtis, S., Campbell, J., Phielipp, M., Lee, S., Baral, C., and Ben Amor, H. Language-conditioned imitation learning for robot manipulation tasks. *Advances in Neural Information Processing Systems*, 33:13139–13150, 2020.

Wang, C., Fan, L., Sun, J., Zhang, R., Fei-Fei, L., Xu, D., Zhu, Y., and Anandkumar, A. Mimicplay: Long-horizon imitation learning by watching human play. *arXiv preprint arXiv:2302.12422*, 2023.

Xu, M., Xu, Z., Chi, C., Veloso, M., and Song, S. Xskill: Cross embodiment skill discovery. In *Conference on Robot Learning*, pp. 3536–3555. PMLR, 2023.

Yang, J., Bisk, Y., and Gao, J. Taco: Token-aware cascade contrastive learning for video-text alignment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 11562–11572, 2021.

Zhang, Z., Li, Y., Bastani, O., Gupta, A., Jayaraman, D., Ma, Y. J., and Weihs, L. Universal visual decomposer: Long-horizon manipulation made easy. *arXiv preprint arXiv:2310.08581*, 2023.

A. Contrasitive Learning for Robotics Representations

A.1. InfoNCE & Classical Contrastive Learning

InfoNCE is an unsupervised contrastive learning objective derived from the principle of noise contrastive estimation as referenced in (Gutmann & Hyvärinen, 2010). The popular learning technique for representation ϕ is classical contrastive learning. More specifically, given an 'anchor' point x (also known as context), and a distribution of positive instances x_{pos} and negative instances x_{neg}, the InfoNCE objective seeks to optimize the following equation:

$$\min_{\phi} \mathbb{E}_{x_{\text{pos}}} \left[-\log \frac{\mathcal{S}_\phi(x, x_{\text{pos}})}{\mathbb{E}_{x_{\text{neg}}} \mathcal{S}_\phi(x, x_{\text{neg}})} \right] \tag{6}$$

In this equation, $\mathbb{E}_{x_{\text{neg}}}$ is frequently approximated using a fixed number of negatives in practical applications.

A.2. Contrastive Learning for Single Modality

For robotics tasks or decision-making scenarios, an important property is to distinguish the representations from various states within a trajectory or demonstration. For a single modality (vision modality), time contrastive learning is an effective approach used in R3M (Nair et al., 2022b) and VIP (Ma et al., 2022).

To encourage vision encoder ϕ to capture features relevant to physical interaction and sequential decision-making, R3M and VIP leverage a unified time contrastive loss (TCN) (Sermanet et al., 2018). TCN is a contrastive learning objective that learns a representation in time-series data (e.g., video trajectories). The original work (Sermanet et al., 2018) considers multi-view videos and performs contrastive learning over frames in separate videos; in this work, we consider the single-view variant. At a high level, TCN attracts representations of frames that are temporally close, while pushing apart those of frames that are farther apart in time.

$$\mathcal{L}(\phi) = \mathbb{E}_{p(g)}[(1-\gamma)\mathbb{E}_{\mu_0(o;g)}\left[-\mathcal{S}(\phi(o);\phi(g))\right] + \log \mathbb{E}_{(o,o';g)\sim D}\left[\exp\left(\mathcal{S}(\phi(o);\phi(g)) + 1 - \gamma\mathcal{S}(\phi(o');\phi(g))\right)\right]] \tag{7}$$

A.3. Contrastive Learning for Vision-Language Alignment

A standard way to learn a vision-language representation is by learning a cross-modal joint embedding that aligns the modalities via contrastive learning. Specifically, the two modalities are semantically aligned by minimizing the InfoNCE objective as Equation (6):

$$\mathcal{L}_{\text{InfoNCE}}(\phi, \psi) = \mathbb{E}_{p(o,l)} \left[-\log \frac{e^{\mathcal{S}(\phi(o);\psi(l))}}{\mathbb{E}_{D(o')}\left[e^{\mathcal{S}(\phi(o');\psi(l))}\right]} \right] \tag{8}$$

where $\mathcal{S}$ is a choice of similarity metric. Intuitively, this objective aims to attract the representations of matching image-text pairs (o, l), while repelling mismatching pairs. Many state-of-the-art vision-language models (Radford et al., 2021; Li et al., 2022; Ma et al., 2023) train with this InfoNCE objective at scale to deliver strong zero-shot performance on a myriad of vision-language tasks.

Based on VIP, LIV extends the framework to multi-modal goal specifications. This is straightforward given the goal-conditioned nature of Eq. (7), since LIV can simply replace encoded image goal $\phi(g)$ with encoded text goal $\psi(l)$ and optimize for a *multi-modal* VIP objective:

$$\begin{aligned}
\mathcal{L}(\phi, \psi) = &+ \mathbb{E}_{p(g)}[(1-\gamma)\mathbb{E}_{\mu_0(o;g)}[-\mathcal{S}(\phi(o);\phi(g))] \\
&\underbrace{+ \log \mathbb{E}_{(o,o';g)\sim D}\left[\exp\left(\mathcal{S}(\phi(o);\phi(g)) + 1 - \gamma\mathcal{S}(\phi(o');\phi(g))\right)\right]]}_{\text{VIP-I}} \\
&+ \mathbb{E}_{p(l)}[(1-\gamma)\mathbb{E}_{\mu_0(o;l)}\left[-\mathcal{S}(\phi(o);\psi(l))\right] \\
&\underbrace{+ \log \mathbb{E}_{(o,o';l)\sim D}\left[\exp\left(\mathcal{S}(\phi(o);\psi(l)) + 1 - \gamma\mathcal{S}(\phi(o');\psi(l))\right)\right]]}_{\text{VIP-L}}
\end{aligned} \tag{9}$$

As shown, this objective consists of two independent components; VIP-I (Image) encourages the representation to encode an *image* goal-conditioned value function, and likewise, VIP-L (Language) for *language* goal.

A.4. Constrastive Learning for Video-Language Alignment

Intuitively, the alignment between video and language, in essence, doesn't differ significantly from the alignment between image and language. Some methods (Yang et al., 2021; Ju et al., 2022; Luo et al., 2022; Li et al., 2023) in the computer vision field aggregate images within a video segment, aligning them in a uniform manner as image-language alignment.

Specifically, given a set of N video-text pairs $\{(v_i, t_i)\}_{i=1}^{N}$, the goal is to learn an optimal scoring function s such that paired video and text (v_i, t_i) have higher scores than all the other unmatched pairs $(v_j, t_k), j \neq k$. From the probabilistic perspective, aligning v_i to t_i is equivalent to maximizing the conditional probability $p(v_i|t_i)$ while minimizing the probability for all negative pairs $p(v_j|t_i), j \neq i$. Similarly with Equation (6), $p(v_j|t_i)$ can be approximated by:

$$p(v_j|t_i) \sim \frac{\exp^{s(v_j,t_i)}}{\sum_{k=1}^{N} \exp^{s(v_k,t_i)}} \tag{10}$$

where $s(v, t)$ is the alignment score between v and t; the denominator is a sum over all possible videos, which is a partition function for normalization. Adding cross-entropy loss on $p(v_j|t_i)$, we can then derive the NCE loss:

$$\mathcal{L}_{\text{InfoNCE}} = \sum_{i=1}^{N} -\log p(v_i|t_i) \sim \sum_{i=1}^{N} -\log \left(\frac{\exp^{s(v_i,t_i)}}{\exp^{s(v_i,t_i)} + \sum_{k\neq i} \exp^{s(v_k,t_i)}} \right) \tag{11}$$

A.5. Historical-aware Contrastive Learning for Keyframe-Skill Alignment

However, these video-language Alignment methods that match a single label to the entire video representation can fall short when dealing with long-horizon videos that contain more information, i.e., more labels. Considering the challenges of keyframe identification and skill annotation in robotics demonstrations, this indicates the necessity for a more fine-grained representation.

For this, we propose a historical-aware contrastive learning to construct more challenging negative examples, which can significantly enhance the robustness of representation learning as pointed out in (Yang et al., 2021). The core insight here is to enhance the influence of historical frames on the alignment of the current frame to the relevant skill, avoiding isolated frames without action recognition and singular skill alignment.

Specifically, we devise three types of negatives to thoroughly examine the models' skill grounding capacities: 1) ***Incorrect Skill Alignments***: we sample negative examples denoted as $\{o^+, h^+, l^-\}$ by altering the pairwise skill to the skill language annotation from another random demonstration. 2) ***Disjoint Frame-History Compositions***: we stitch the pairwise keyframe and skill with mismatched historical frames randomly selected from other demonstrations, thereby constructing negative samples $\{o^+, h^-, l^+\}$. This prevents overfitting between isolated frames and skills. 3) ***Semantic Reversals via Video Inversion***: we keep the set of frames the same but reverse the video order, which can significantly change its semantic interpretation. For example, a video showing the action of "opening the door" when reversed, semantically becomes "closing the door". By temporally reversing the entire sub-video chunk, we construct more challenging negative samples counterfactually, depicted as $\{(o^+, h^+)^{rev}, l^+\}$.

$$\mathcal{L}_{\text{contrastive}} = \underbrace{-\log \frac{e^{\mathcal{C}(o^+,h^+,\ell^+)}}{\sum_{j=1}^{k} e^{\mathcal{C}(o^+,h^+,\ell_j^-)}}}_{\text{Incorrect Skill Misalignments}} \underbrace{-\log \frac{e^{\mathcal{C}(o^+,h^+,\ell^+)}}{\sum_{z=1}^{k} e^{\mathcal{C}(o^+,h_z^-,\ell^+)}}}_{\text{Disjoint Frame-History Compositions}} \underbrace{-\log \frac{e^{\mathcal{C}(o^+,h^+,\ell^+)}}{\sum_{w=1}^{k} e^{\mathcal{C}(\{(o^+,h^+)_w^{rev},l^+\})}}}_{\text{Semantic Reversals via Video Inversion}}$$

where k is the number of negative samples. The core insight is to enhance the semantic alignment between skill and video embedding with historical frames and avoid overfitting with isolated keyframes. Meanwhile, we aim to strengthen the connection between the historical context and the current frame, without neglecting the temporal relationship of the historical context. Overall, history-aware contrastive learning finetunes the representation at the inter-skill level, encouraging temporal-enhanced representation to distinguish from diverse skills.

B. Environment Details

Maniskill2. ManiSkill2 is a large-scale benchmark for evaluating generalizable robotic manipulation skills, built using the SAPIEN simulator. It encompasses 20 diverse manipulation task families with over 2000 customizable 3D object models.

The tasks involve both stationary and mobile robots, single & bi-manual arms, and rigid & soft bodies, with 2D/3D visual observations from dynamic simulation. A key feature is the procedural scene and object generation allowing systematic topology and geometry variations for studying generalization. It provides over 4 million demonstration frames rendered at 2000FPS along with ground-truth keyframes enabling algorithms like imitation learning. It has a unified interface to support various approaches including classic planning methods, reinforcement learning agents, and learning from demonstration. The tasks cover a wide range of manipulation skills from picking, pushing, and opening doors/drawers to more complex daily activities via skill composition. ManiSkill2 also implements a render server to optimize memory usage when training models like CNN policies that require fast simulation.

CALVIN. The Composing Actions from Language and Vision (CALVIN) benchmark focuses on long-horizon, language-guided robotic manipulation tasks. It consists of a simulated Franka robot arm placed next to a desk with interactive objects including a drawer, cabinets, a light switch, and colored blocks. CALVIN procedural generation supports variation across table textures, furniture positions, and block configurations to enable studying generalization. In CALVIN, the robot must follow a chained sequence of natural language instructions provided in unrestricted free-form, such as "Open the red drawer, pick up the blue block near the drawer..." requiring complex temporally dependent behavior. The tasks feature longer action horizons, larger action spaces, and more complex language conditioning compared to prior vision-and-language datasets. CALVIN also allows configuring different sensor inputs like RGB, depth, and proprioception that the agents must utilize for solving tasks using their learned environment models and language grounding.

FrankaKitchen. The FrankaKitchen environment consists of a simulated kitchen scene containing a 7-degree-of-freedom Franka robot arm that interacts with various common household objects including a microwave, a kettle, two stove burners, a light switch, and sliding & hinged cabinets. Robot demonstration episodes were collected across 7 manipulation sub-tasks - opening/closing the microwave, kettle, and cabinets, operating the light switch, and controlling the stove burners. Each demonstration episode comprises an arbitrary order completion of 4 out of the 7 sub-tasks, leading to 24 possible ordered sequences. These refined human demonstrations serve as the training dataset encompassing 7 diverse manipulation skills for robots to learn. During evaluation, the Franka robot must mimic the demonstrations by manipulating the kitchen objects to match specified target configurations across a complete trajectory involving multiple temporally dependent sub-tasks. This makes FrankaKitchen well-suited for benchmarking long-horizon robotic manipulation learning approaches with interactive objects under simulated home settings.

C. Additional Details of Experiments

C.1. Data Collection

We employ several ways to collect long-horizon demonstrations with groundtruth keyframes and skills for training. For the 80+ tasks in Maniskill2, the Maniskill official provides 100 demonstrations per task[1], totaling 80k trajectories. We used the privileged information provided by Maniskill simulator to mark the groundtruth keyframes and corresponding skill descriptions for training purposes.

For CALVIN, we borrowed checkpoints released on HUlC[2] (Mees et al., 2022a) and gathered 25k long-horizon demonstrations . Specifically, we first randomized the initial configuration of the manipulation scene. Then we designed diverse valid instruction chains ranging from 2 to 10 sub-tasks. We collected successful long-horizon demonstrations step by step, following varied language instruction chains *from bottom to top*. In this way, we could automatically identify the transition frame between adjacent sub-tasks as keyframes and annotate pairwise descriptions as groundtruth skills.

For Franka's kitchen tasks, we trained a GCBC policy for multiple tasks. In each episode, four out of the seven objects are manipulated in an arbitrary sequence, yielding 24 unique completion orders. For each task, we collected 500 trajectories each, resulting in a total of 12k demonstrations. We also annotate the groundtruth keyframe and corresponding skill with privileged information. Regarding all the data collected, we divided it into an 80% proportion for the training dataset $\mathcal{D}^{train}$ and 20% for the testing dataset $\mathcal{D}^{test}$.

[1] https://haosulab.github.io/ManiSkill2/concepts/demonstrations.html

[2] https://github.com/lukashermann/hulc/blob/main/checkpoints/download_model_weights.sh

C.2. Architecture of KISA

The architecture of KISA is listed in Table 6. Specifically, KISA employs different image encoders including R3M, VIP, and LIV depending on the variant. For modeling temporal information, a temporal enhancement module is utilized in all model versions as denoted as True. The text encoder is sourced from the CLIP model in KISA-VIP and KISA-R3M while KISA-LIV uses its own text encoder. The multimodal fusion happens through the temporal enhancement transformer module, which functions as the core temporal reasoning component. The Temporal-Enhance Module consists of a 6-layer transformer architecture to model temporal context and dynamics across video frames. Each transformer layer employs multi-headed self-attention, using 8 parallel attention heads to jointly relate different regions of the input frame features. The output hidden state representations from this module have a dimension of 1024 per frame, capturing enhanced temporal aware semantics. Through stacked self-attention layers and multi-head injections, the module encodes both short and long-range dependencies between video frames using the transformer mechanism. The encoded sequence captures the temporal evolution of semantics, surroundings, and actions, providing an expressive spatio-temporal video representation for improved understanding. Overall, this modular architecture with shared components allows analyzing the contribution from different visual backbones by comparing performances of KISA-LIV, KISA-VIP, and KISA-R3M under the same fusion methodology.

Table 6. KISA Architecture

	KISA-LIV	KISA-VIP	KISA-R3M
Image-Encoder	LIV-I	VIP	R3M
Language-Encoder	LIV-L	CLIP	CLIP
Transformer Layers	6	6	6
Transformer Heads	8	8	8
Hidden Size	1024	1024	1024
Temporal-Enhance Module	True	True	True
Contrastive Learning	History-aware	History-aware	History-aware

C.3. Metric Details

- **Keyframe Number Errors:** Measured by the ratio between predicted and ground truth number of keyframes. It directly quantifies how accurately the model estimates the true counts of keyframes needed. A lower value demonstrates better capability of the approach to identify salient steps. This metric profiles the model's comprehension of skill progression and structure irrespective of the individual frames' precision.
- **Mean Absolute Error (MAE):** MAE calculates the absolute differences between the predicted keyframe timestamp and ground truth timestamp averaged over the entire test set. It evaluates the accuracy of temporal localization - how precisely the model can identify the exact moments of importance within the execution sequence. MAE directly summarizes the deviation between predicted and ground truth keyframe positions, enabling an intuitive interpretation of how well the model has understood the progression structure and critical steps underlying a process. The lower the MAE, the better a model has learned to accurately pinpoint timely step transitions critical to decomposing and reasoning about skills
- **F1-Score:** F1-Score is the harmonic mean of Precision and Recall for evaluating keyframe detection performance. Precision measures the percentage of predicted keyframes that are true positives, capturing relevance. Recall calculates the percentage of ground-truth keyframes that are correctly detected, quantifying completeness. By combining precision and recall, the F1-Score ranges from 0 to 1 with higher values indicating optimal performance in precisely discovering true keyframes. An ideal detector should achieve an F1-Score approaching 1, perfectly pinpointing keyframes without redundancy.
- **Top-1 accuracy:** Measures if the predicted skill category with maximum model confidence matches the true label. It assesses the correctness of categorical classification without considering temporal localization. Top-1 Accuracy profiles the model's capability in disambiguating and discriminating between fine-grained skills based on observing key execution patterns even if $start/stop$ timings are imprecise.

C.4. Details of Training

The training hyperparameters used during the pre-training stages are listed in Table 7. Specifically, the AdamW optimizer was utilized for all pre-training across the Maniskill2, CALVIN, and FrankaKitchen datasets. The batch sizes were set to 32

and 64 depending on the dataset complexity. All models were trained from scratch with a consistent learning rate of 0.0001 and weight decay of 0.00002 applied for regularization. Additionally, alpha α was set to 0.1 universally for scaling the skill alignment score loss in the overall training objective, which comprises the contrastive alignment loss and scaled score prediction components. This enables the model to focus more on capturing trends in skill progressions across steps rather than exact score differences. These calibrated hyperparameter settings ensure stable optimization and generalization during the pre-training phase. We develop separate models for each environment leveraging their full unique demonstration datasets spanning over 100,000 examples. The extensive training times of up to 200 hours exemplify modeling long, contextual skill sequences. For the ManiSkill2 environment, 80,000 demonstration trajectories were used to train the corresponding model for 200 hours. On the CALVIN dataset, a distinct model was trained using 25,000 trajectories for 200 hours. 10,000 demonstration sequences from the Kitchen environment were utilized to train the specialized kitchen model for 20 hours. All training and testing results are obtained on a server equipped with an Intel(R) Xeon(R) Gold 6226R CPU @ 2.90GHz and an NVIDIA A800 PCIe 80 GB.

Table 7. Pre-Training Hypermeters

	Maniskill2	CALVIN	FrankaKitchen
Optimizer	AdamW	AdamW	AdamW
Max Epoch	200	300	200
Batch Size	32	64	32
Learning Rate	1e-4	1e-4	1e-4
Weight Decay	2e-5	2e-5	2e-5
Alpha	0.1	0.1	0.1

C.5. Details of Finetuning

To compare the enhancement of pretrained representations by the KISA design components, we have also designed a series of baselines, directly fine-tuning these representations frame-wise for keyframe identification, denoted as FT-R3M, FT-VIP, and FT-LIV. Specifically, we added an MLP layer to both the frozen vision encoder and language encoder, and leveraged vanilla contrastive learning to construct positive and negative samples of frame and skill pairs for fine-tuning. Note that vanilla contrastive learning means we ablate the history-aware contrastive learning we design for the no fine-grained temporal-enhanced module. To ensure fair comparisons, we keep other details like the amount of data and the number of training iterations the same as KISA. The finetuning hyperparameters and architecture are listed in Table 8.

Table 8. Finetuning Model Architecture

	FT-LIV	FT-VIP	FT-R3M
Image-Encoder	LIV-I	VIP	R3M
Language-Encoder	LIV-L	CLIP	CLIP
Hidden Size	1024	1024	1024
MLP Input-dim	1024	1024	2048
MLP Output-dim	1024	1024	1024
Temporal-enhanced	-	-	-
Contrastive Learning	Standard	Standard	Standard

D. Details of Baselines

D.1. VideoRLCS

For short-horizon, single-stage demonstrations, keyframes often align with reward progression extremes that prove most explanatory for final outcomes. However, as temporal complexity increases across long-horizon, multi-task sequences, interim peaks may lack correspondence to transitions between substantive phases. Intermediate rewards may produce extremes without corresponding to explanatory keyframes that cleanly separate adjacent skills. Reward signals alone risk confusing model uncertainty and temporary deviations for representationally salient states. Some underlying reasons may be: complex trajectories interleave and superimpose varying sub-policies with entangled dynamics. Local rewarded moments may not adequately disambiguate between constituent skills. Meanwhile, prolonged executions accumulate drift, making accurate terminal reward credit assignment to transient factors challenging. In essence, reliance solely on reward

proxies risks keyframes localizing to misleading or incoherent points lacking skill grounding. Our approach demonstrates incorporating declarative annotations and enhanced receptive history better filters noise to focus on explanatory pivot states between meaningful phases.

The VideoRLCS approach consists of two main modules. 1)Predictor $\mathcal{G}$: This module is responsible for predicting expected returns y_i directly from input video demonstrations s_i, without access to underlying states or rewards; 2)Detector $\mathcal{D}$: This module focuses on identifying critical states most salient for the return predictions made by the first module. It employs a mask-based sensitivity analysis by systematically masking video segments and examining prediction changes, to pinpoint important moments that provide pivotal behavior cues for estimating returns. The detector is trained using three loss functions - importance preservation, compactness, and reverse loss.

$$\mathcal{L}_{\mathcal{D}} = \lambda_s \mathcal{L}_{\mathcal{D}}^{imp} + \lambda_r \mathcal{L}_{\mathcal{D}}^{com} + \lambda_v \mathcal{L}_{\mathcal{D}}^{rev}, \tag{12}$$

where $\lambda_s, \lambda_r, \lambda_v$ are the weights for importance preservation loss, compactness loss, and reverse loss respectively. The importance preservation loss $\mathcal{L}_{\mathcal{D}}^{imp}$ ensures states identified by the detector preserve information critical for return prediction. The compactness loss $\mathcal{L}_{\mathcal{D}}^{com}$ encourages sparsity in detected states. The reverse loss $\mathcal{L}_{\mathcal{D}}^{rev}$ validates unidentified states are unimportant by masking them and verifying inability to estimate returns.

$$\mathcal{L}_{\mathcal{D}}^{imp} = \sum_i \mathcal{L}_{\mathcal{G}}(\mathcal{G}(\mathbf{s}_i \circ \mathcal{D}(\mathbf{s}_i)), \mathbf{y}_i), \mathcal{L}_{\mathcal{D}}^{com} = \sum_i ||\mathcal{D}(\mathbf{s}_i)||_1, \mathcal{L}_{\mathcal{D}}^{rev} = -\sum_i \mathcal{L}_{\mathcal{G}}(\mathcal{G}(\mathbf{s}_i \circ (1 - \mathcal{D}(\mathbf{s}_i))), \mathbf{y}_i). \tag{13}$$

D.2. KTS

Given an input video with n frames represented as feature descriptors $x_i \in X$, KTS first computes a positive definite kernel matrix $K \in R^{n \times n}$ that encodes the similarity between every frame pair under feature mapping ϕ. Based on this affinity graph, KTS efficiently partitions the video to maximize an information-theoretic objective function using dynamic programming. Specifically, for a segmentation candidate defined by change points $t_0, t_1, \cdots, t_{m-1}$ that splits the video into m contiguous sections, KTS optimizes:

$$\underset{m;t_0,\ldots,t_{m-1}}{\text{Minimize}} \quad J_{m,n} := L_{m,n} + Cg(m,n) \tag{14}$$

where $L_{m,n}$ measures the within-segment feature variance, rewarding self-similarity. $g(m,n)$ penalizes over-segmentation. C balances the two terms and is cross-validated.

$$L_{m,n} = \sum_{i=0}^{m} v_{t_{i-1},t_i}, \quad v_{t_i,t_{i+1}} = \sum_{t=t_i}^{t_{i+1}-1} \|\phi(x_t) - \mu_i\|_{\mathcal{H}}^2, \quad \mu_i = \frac{\sum_{t=t_i}^{t_{i+1}-1} \phi(x_t)}{t_{i+1} - t_i} \tag{15}$$

By trading off the competing criteria of coherent sections and complexity control, the optimized change points returned by dynamic programming constitute optimal scene boundaries that maximize inter-section distinctiveness.

D.3. UVD

In a demonstration, the last frame o_T is naturally a goal. Now, conditioned on a subgoal o_t, UVD attempts to extract the first frame o_{t-n} in the sub-sequence of frames that depicts visual task progression to o_t. To discover this first frame, UVD exploits the fact that several state-of-the-art pre-trained visual representations for robot control (Nair et al., 2022b; Ma et al., 2022; 2023) are trained to capture temporal progress within short videos depicting a single solved task; these representations can effectively produce embedding distances that exhibit *monotone* trend over a short goal-reaching video sequence $\tau = (o_{t-n}, ..., o_t)$:

$$d_\phi(o_s; o_t) \geq d_\phi(o_{s+1}; o_t), \forall s \in \{t-n, \ldots, t-1\}, \tag{16}$$

where d_ϕ is a distance function in the ϕ-representation space; in this work, we set $d_\phi(o; o') := ||\phi(o) - \phi(o')||_2$ because several state-of-the-art pre-trained representations use the L_2 distance as their embedding metric for learning. Given this, UVD sets the previous subgoal to be the temporally closest observation to o_t for which this monotonicity condition fails:

$$o_{t-n-1} := \arg\max_{o_h} d_\phi(o_h; o_t) < d_\phi(o_{h+1}; o_t), h < t\ . \tag{17}$$

The intuition is that a preceding frame that belongs to the same subtask (i.e., visually apparent that it is progressing towards o_t) should have a higher embedding distance than the succeeding frame if the embedding distance indeed captures temporal

progression. As a result, a deviation from the monotonicity indicates that the preceding frame may not exhibit a clear relation to the current subgoal, and instead be a subgoal itself. Now, o_{t-n-1} becomes the new subgoal, and UVD apply (17) recursively until the full sequence τ is exhausted.

UVD designs a greedy heuristics method upon these pretrained visual features to capture the phase peaks. However, the manually defined heuristics rules are sensitive to hyperparameters including peak detection tolerance or smoothness window length, which will lead to over-identification or mis-identification. More importantly, UVD only considers visual embeddings without language grounding, which are semantically uninterpretable keyframes that do not align with distinct skills.

E. Confidence Scores for Keyframe Identification

With the pretrained vision encoder ϕ, pretrained language encoder ψ, temporal enhance module Φ, first we extend the frame-wise representation to video-level $v_i = \Phi(h_i, o_i) = \Phi_{\text{TEMP}}(\{\phi(o_0), \ldots, \phi(o_{i-1}), \phi(o_i)\})$. Then we calculate embedding similarity $\mathcal{C}(o_i, h_i, \ell_s) = \cos(v_i, \ell_s)$ between v_i and skill ℓ_i from skill library $\mathcal{S} := \{l_s\}_{s=0}^{S}$ for each frame. The confidence score is the maximum similarity $f_i = \max\{\mathcal{C}(o_i, h_i, \ell_s)\}_{s=0}^{S}$ between video-level representation and language representation from skills. The local maximums or peaks from all frames of the video demonstrations can be identified as a keyframe and the corresponding skill l_s can be the pairwise skill annotation.

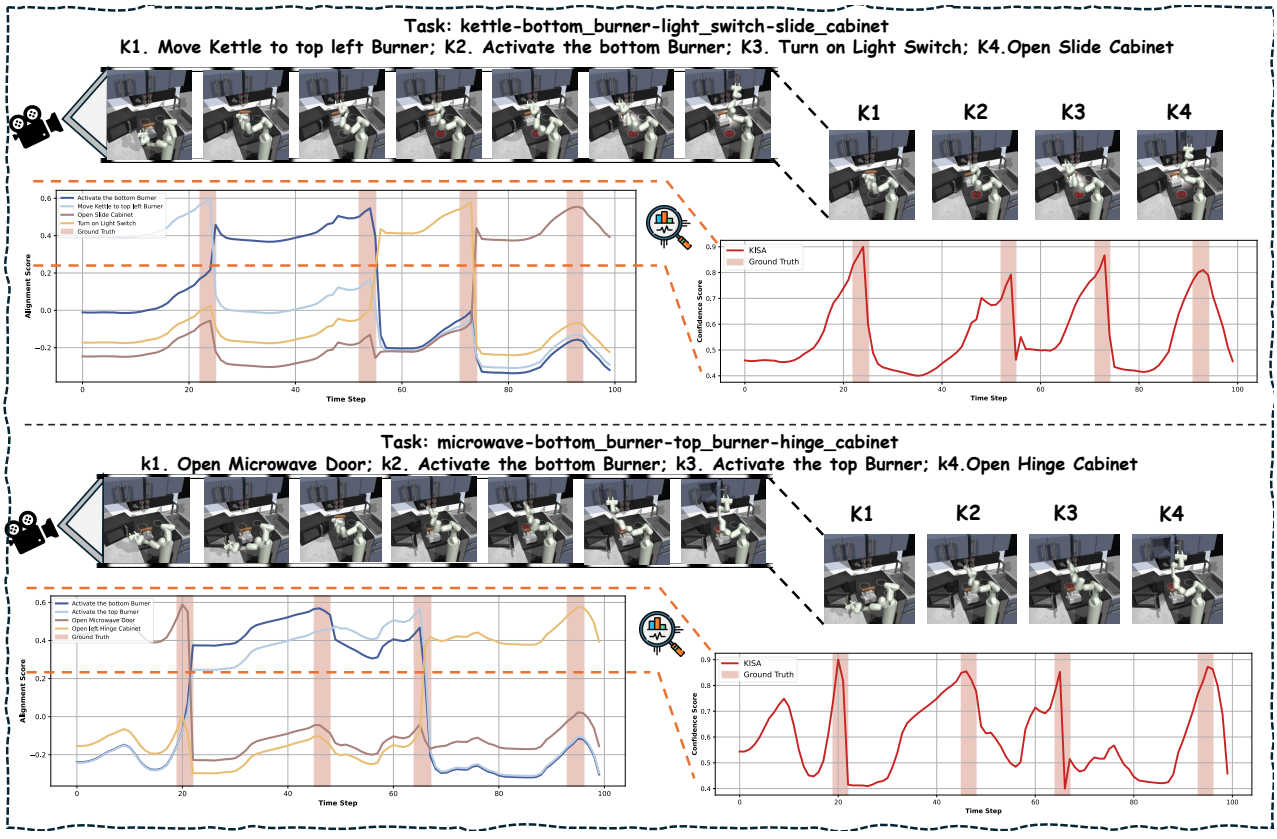

Figure 9. **A example of confidence score calculation.** For each step, the confidence score is the maximum similarity between video-level representation and language representation from skills.

F. The Heatmap of Alignment Score between Skills and Frames

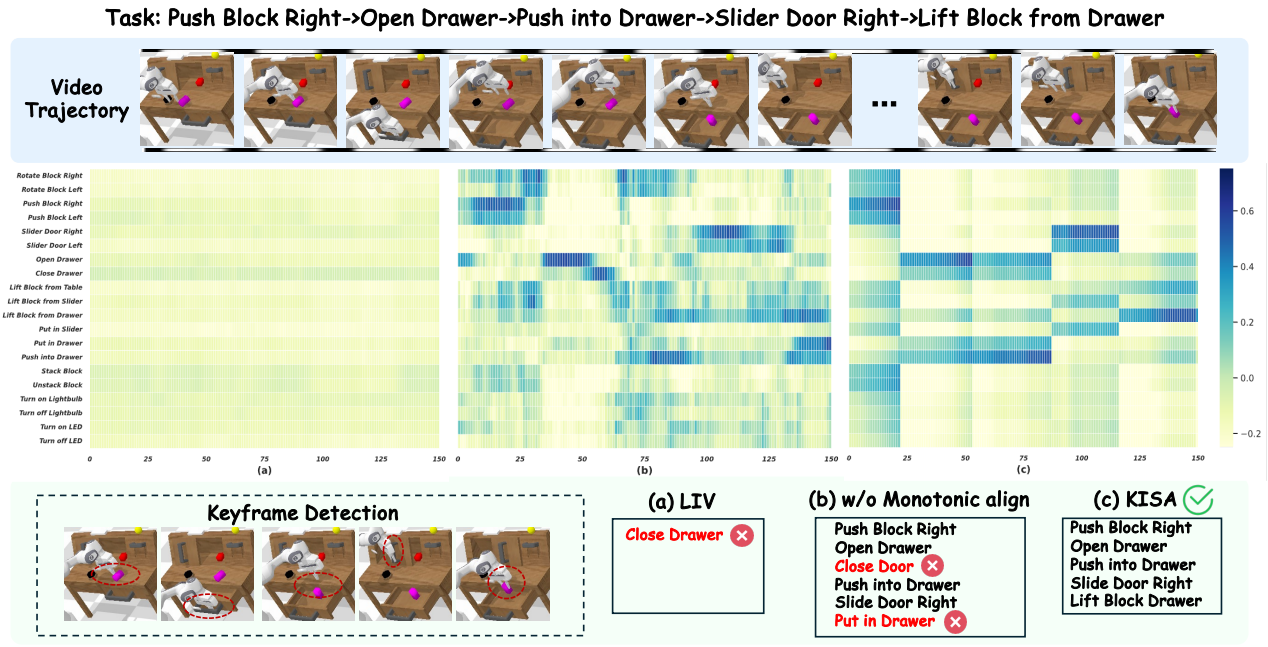

Figure 10. The heatmap between skills and frames from a long-horizon demonstration example on CALVIN. The comparisons between LIV (a), KISA w/o monotonic alignment (b) and KISA (c).

G. Details of Policy Learning

In this section, we introduce more details about the experiments of policy learning in Section 5.6.

G.1. Details of Policy Backbones

We selected several baselines for comparison, including Language-Conditioned Behavior Cloning (LCBC) (Stepputtis et al., 2020) that learns directly from demonstrations, using general language instructions as the sole condition in an end-to-end manner.

$$p_\Theta(\tau_a|\mathcal{L}, x_0) = \prod_{t=1}^{T} p_\psi(a_t|x_t, \mathcal{L}) \tag{18}$$

We conducted an experiment to verify that providing long-horizon demonstrations with some keyframe and skill annotations can offer intermediate landmarks and guides. This avoids relying solely on general language instructions as the only condition for a policy of LCBC. We leverage a hierarchical framework that decomposes a complex demonstration into several shorter subtasks to facilitate reusable skills and further enable modular skill composition. Specifically, the formulation followed as:

$$p_\Theta(\tau_a|\mathcal{L}, x_0) = \underbrace{\left(\prod_{i=1}^{N} p_\phi(m_i|\mathcal{L}, x_0)\right)}_{\text{Skill planning}} \underbrace{\left(\prod_{i=1}^{N}\prod_{t=1}^{T} p_\psi(a_{i,t}|x_t, m_i)\right)}_{\text{Action planning}} \tag{19}$$

For unlabeled demonstrations without the explicit task boundaries with clear skill annotation, LISA (Garg et al., 2022), a hierarchical framework that first discovers implicit skills and then learns to combine skills for complex tasks, which means LISA has an additional objective to learn skills. Specifically, LISA uses Vector Quantization (VQ) to learn a codebook $\mathcal{C} \in \{z^1, \ldots, z^K\}$ of K embedding vectors. Given an embedding $\tilde{z}$ from the skill predictor f, it maps the embedding to the closest vector in the codebook:

$$z = \mathbf{q}(\tilde{z}) =: _{z^k \in \mathcal{C}} \|\tilde{z} - z^k\|_2$$

with the codebook vectors updated to be the moving average of the embeddings z closest to them. This can be classically seen as learning K cluster centers via k-means.

Limited by the need for skill discovery, LISA has to be trained end-to-end using an objective $\mathcal{L}_{\text{LISA}} = \mathcal{L}_{\text{BC}} + \lambda\mathcal{L}_{\text{VQ}}$, where $\mathcal{L}_{\text{BC}}$ is the behavior-cloning loss on the policy p_Θ, λ is the VQ loss weight and $\mathcal{L}_{\text{VQ}}$ is the vector quantization loss on the skill predictor f_ϕ given as:

$$\mathcal{L}_{\text{VQ}}(f) = \mathbb{E}_\tau[\|\text{sg}\left[\mathbf{q}(\tilde{z})\right] - \tilde{z}\|_2^2] \tag{20}$$

with $\tilde{z} = f_\phi(l, (s_t, s_{t-1}, ..))$. The coupled learning is not stable and learned skill latent code cannot establish a one-to-one relationship with action primitives, which may potentially hinder the skill-conditioned policy learning. But with the privileged information including explicit keyframes and skill annotations provided by KISA, LISA can avoid re-discovering skills.

G.2. Training Details

For fair comparisons we implement the LCBC as the same Transformer architecture with LISA, borrowed from the implementation of the flat decision transformer in LISA. In other words, the LCBC is based on DT and is implementation-wise similar to LISA, but without a skill predictor network. The policy here is a Causal Transformer, where we modify DT to condition the language instruction embedding from the pretrained language encoder from LIV. For LISA with the annotation data from KISA or LIV, the skill discovery is not needed and the training of skill prediction and low-leve behavior cloning can be decoupled, as in Equation (19). The hyperparameters of LCBC and LISA are listed in Table 9 and Table 10, borrowed from the official paper. The other hyperparameters of LISA+LIV and LISA+KISA keep the same with LIV for fair comparisons.

Table 9. **LCBC Hyperparameters**

Hyperparameter	CALVIN
Transformer Layers	2
Transformer Embedding Dim	128
Transformer Heads	4
Dropout	0.1
Batch Size	128
Policy Learning Rate	$1e-4$
Optimizer	Adam

Table 10. **LISA Hyperparameters**

Hyperparameter of LISA	CALVIN
Transformer Layers	1
Transformer Embedding Dim	128
Transformer Heads	4
Skill Code Dim	16
Number of Skills	20
Dropout	0.1
Batch Size	128
Policy Learning Rate	$1e-4$
Skill Predictor Learning Rate	$1e-5$
VQ Loss Weight	0.25
Horizon	10
VQ EMA Update	0.99
Optimizer	Adam

H. Pesudocodes of Framework

Algorithm 1 KISA Training

Input: Training dataset $\mathcal{D}^{train} = \{V_n = (o_0, ..., o_T)\}_{n=1}^{N}$ with N long-horizon demonstrations, horizon length T, for each frame o_i with historical frames h_i , skill language description ℓ_i, keyframe of corresponding sub-video chunk $o_i^{\mathcal{K}}$, pretrained vision encoder ϕ, pretrained language encoder ψ, temporal enhance module Φ, coefficient α.
while not done **do**
  Sample a batch of $V := (o_0, ..., o_T)$ from training dataset $\mathcal{D}^{train}$
  for $i = 0$ **to** T **do**
    Extract pair-wise $\{o^i, h^i, \ell^i\}$ for each o_i, denoted as positive samples $\{o^+, h^+, \ell^+\}$
    Construct negative samples $\{o^+, h^+, \ell^-\}$ via incorrect skill alignments, skills sampled from other demonstrations without the skill $\ell+$.
    Construct negative samples $\{o^+, h^-, \ell^+\}$ via disjoint frame-history composition, historical frames h^- sampled from other demonstrations or other sub video-chunk in the same demonstration.
    Construct negative samples $\{(o^+, h^+)_w^{rev}, \ell^+\}$ via reverse the video order but keep the set of frames the same.
    Extend the frame-wise representation to video-level $v_i = \Phi(h_i, o_i) = \Phi_{\text{TEMP}}(\{\phi(o_0), \ldots, \phi(o_{i-1}), \phi(o_i)\})$
    Calculate embedding similarity $\mathcal{C}(o_i, h_i, \ell_i) = \cos(v_i, \ell_i)$ between v_i and skill ℓ_i
    Calculate frame-wise visual similarity $\mathcal{D}(\phi(o_i), \phi(o_i^{\mathcal{K}}))$ between o_i and $o_i^{\mathcal{K}}$
    Calculate contrastive learning loss:

$$\mathcal{L}_{\text{contrastive}} = \underbrace{-\log \frac{e^{\mathcal{C}(o^+, h^+, \ell^+)}}{\sum_{j=1}^{k} e^{\mathcal{C}(o^+, h^+, \ell_j^-)}}}_{\text{Incorrect Skill Misalignments}} \underbrace{-\log \frac{e^{\mathcal{C}(o^+, h^+, \ell^+)}}{\sum_{z=1}^{k} e^{\mathcal{C}(o^+, h_z^-, \ell^+)}}}_{\text{Disjoint Frame-History Compositions}} \underbrace{-\log \frac{e^{\mathcal{C}(o^+, h^+, \ell^+)}}{\sum_{w=1}^{k} e^{\mathcal{C}(\{(o^+, h^+)_w^{rev}, l^+\})}}}_{\text{Semantic Reversals via Video Inversion}}$$

    Calculate monotonic alignment loss: $\mathcal{L}_{\text{monotonicity}} = \frac{1}{T} \sum_{i=1}^{T} ||S(\Phi(o_i, h_i); \psi(\ell_i)) - \mathcal{D}(\phi(o_i), \phi(o_i^{\mathcal{K}}))||$
    Update Φ by minimizing total loss: $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{contrastive}} + \alpha \cdot \mathcal{L}_{\text{monotonicity}}$
  end for
end while

Algorithm 2 KISA Testing

Input: Testing dataset $\mathcal{D}^{test} = \{V_n = (o_0, ..., o_T)\}_{n=1}^{N}$ with N long-horizon demonstrations, variable horizon length T, scalable skill library $\mathcal{S} = \{l_s\}_{s=0}^{S}$, variable skills numbers S, pretrained vision encoder ϕ, pretrained language encoder ψ, temporal enhance module Φ.
Sample a batch of $V := (o_0, ..., o_T)$ from training dataset $\mathcal{D}^{train}$
for $i = 0$ **to** T **do**
  Extract historical frames h_i for each frame o_i
  Extend the frame-wise representation to video-level $v_i = \Phi(h_i, o_i) = \Phi_{\text{TEMP}}(\{\phi(o_0), \ldots, \phi(o_{i-1}), \phi(o_i)\})$
  Calculate embedding similarity $\mathcal{C}(o_i, h_i, \ell_s) = \cos(v_i, \ell_s)$ between v_i and skill ℓ_i from skill library $\mathcal{S} := \{l_s\}_{s=0}^{S}$
  Calculate the maximum similarity $f_i = \max\{\mathcal{C}(o_i, h_i, \ell_s)\}_{s=0}^{S}$ as the confidence score as keyframe and the corresponding skill l_s can be the pairwise skill.
end for
Find local maximimums $\mathcal{F}_{\text{keyframe}} = \{f_{m_1}, f_{m_2}, ..., f_{m_M}\}$ from confidence score buffer $\mathcal{F} = \{f_i\}_{i=0}^{T}$ where M is the number of keyframes and each f_{m_i} is a local maximum in $\mathcal{F}$.

I. Quick Usage

```
from kisa import load_kisa
representation_backbone = ['R3M', 'VIP', 'LIV']
kisa = load_kisa(representation_backbone)
video = load_data('data_path')
skill_library = load_skill('skill_json_path')
annotated_video = kisa.annotate(video, skill_library)
```

J. Pesudo Algorithm

```
def Calculate_embedding(model, videos, skills):
    '''
    Calculate text embedding and historical-enhanced frame embedding
    videos:tensor(b,t,c,h,w)
    skills:list(b,t)
    '''
    bs,t,_,_,_= videos.shape

    skill_Feature = []
    for i in range(bs):
        skill_Feature.append(model.get_text_feature(skills[i]))
    skill_Feature = torch.stack(skill_Feature, dim=0)

    # encode videos
    image_Feature = model.get_frames_feature(einops.rearrange(videos, 'b t c h w -> (b t)
    c h w'))
    temporal_enhanced_frame_Feature = einops.rearrange(image_Feature, '(b t) c -> b t c',
    t=t)

    # temporal modelling
    temporal_enhanced_frame_Feature = model.get_temporal_frames_feature(
    temporal_enhanced_frame_Feature)

    return temporal_enhanced_frame_Feature, skill_Feature

def Calculate_Similarity(temporal_enhanced_frame_Feature, skill_Feature, mode='cos'):
    '''
    temporal_enhanced_frame_Feature:(b,t,c)
    skill_Feature:(b,t,c)
    '''
    if mode == 'cos':
        cosine_sim = F.cosine_similarity(temporal_enhanced_frame_Feature, skill_Feature,
    dim=-1)
        logits = torch.unsqueeze(cosine_sim, dim=-1)
    return logits
```

K. Skill Analysis

K.1. Skill List

The provided skill list Table 11 forms a part of the Manipulation Skill Library in ManiSkill2 environment, which was collected by interacting with the simulated household objects to be utilized for KISA pre-training. The skills predominantly involve picking up items of varying shapes, sizes, weights, and fragility including foods, containers, tools, toys, and furniture parts. They aim to test robotic grasping, motion planning, and placement capabilities across simple to complex geometries. Beyond pick-and-place skills, the list contains more advanced multi-step skills such as stacking blocks, inserting pegs, plugging chargers, rotating handles, and excavating clay. These skills chain primitive actions with precise spatial-temporal coordination.

Task	Description	Skill Instruction
PickCube	Pick up a cube and move it to a goal position.	Pick up Cube → Move Cube
StackCube	Pick up a red cube and place it onto a green one.	Pick up Cube → Stack on Cube
PickYCB	Pick up a YCB Object and move it to a goal position.	Pick up YCB Object → Move YCB Object
PlugCharger	Plug Charger into wall Receptacle.	Pick up Charger → Move Charger → Plug into Receptacle
PegInsertionSide	Insert a peg into the horizontal hole in a box.	Pick up Peg → Move Peg → Insert Peg into Hole
AssemblingKits	Insert an object into the corresponding slot on a board.	Pick an Object → Insert into corresponding Slot
TurnFaucet	Turn on a Faucet by Rotating its Handle.	Grasp Faucet → Turn Faucet
OpenCabinetDrawer	Open a target Drawer on a Cabinet.	Grasp Cabinet Handler → Open Drawer
OpenCabinetDoor	Open a target Door on a Cabinet.	Grasp Cabinet Handler → Open Door
Excavate	Lift a specific amount of clay to a target height.	Lift Clay → Move Clay
Pour	Pour liquid into a beaker and return to the upright position.	Pour Liquid into Beaker → Move Bottle
Fill	Fill clay from a bucket into the target beaker.	Move Clay → Fill into Baker
Hang	Hang a noodle on a target rod.	Pick up Noodle → Hang on Rod

Table 11. **Overview of tasks in Maniskill2 Dataset.**

Category	Object Generalization
Box	CrackerBox, SugarBox, PuddingBox, GelatinBox
Can	MasterChefCan, TomatoSoupCan, TunaFishCan, PottedMeatCan
Bottle	MustardBottle, WindexBottle
Fruit	Banana, Strawberry, Apple, Lemon, Peach, Pear, Orange, Plum
Tool	PowerDrill, Scissors, Padlock, Hammer, Clamp, Wrench, Screwdriver, Marker, WoodBlock
Tableware	PitcherBase, BleachCleanser, Bowl, Cup, Plate, Mug, Fork, Knife, Spoon, Sponge, Spatula
Ball	Softball, Baseball, Tennisball, Recquetball, Golfball

Table 12. **Object Generalization in Maniskill2 Dataset.**

The skills of CALVIN Table 14 are aimed at advancing reusable robotic manipulation skills through natural language guidance to aid in generalized task completion inside human environments following verbal commands. The language commands encode high-level intentions while leaving specifics of object attributes and end poses open, thus requiring robots to proficiently perceive surroundings and map instructions to feasible motion policies on the fly. The set of 24 skills mainly focuses on dexterous object handling like lifting, moving, rotating, and pushing blocks of varying colors and shapes across table and drawer surfaces. Some skills involve spatial reasoning to translate language referring to left/right and goal configurations. The skills also feature interacting with common household objects like toggling switches, opening/closing drawers and doors, and stacking & unstacking blocks which are key functionalities for assistive robots.

Task	Description	Skill Instruction
bottom burner	Turn the oven knob that activates the bottom burner.	Activate the bottom Burner
top burner	Turn the oven knob that activates the top burner.	Activate the top Burner
light-switch	Turn on the light switch.	Turn on the Light Switch
slide-cabinet	Open the slide cabinet.	Open the Slide Cabinet
hinge-cabinet	Open the left hinge cabinet.	Open the left Hinge Cabinet
microwave	Open the microwave door.	Open the Microwave Door
kettle	Move the kettle to the top left burner.	Move the Kettle to the top left Burner

Table 13. **Overview of tasks in Franka-Kitchen Dataset.**

The skills of FrankaKitchen Table 13 predominantly involve opening/closing doors and drawers of microwave, cabinet, and kettle using different wrist orientations and grasps. They also feature operating switches and knobs that actuate electronics like lightbulbs and stove burners.

Task	Description	Skill Instruction
rotate-red-block-right	Rotate the red Block 90 degrees to the right.	Rotate Block Right
rotate-blue-block-right	Rotate the blue Block 90 degrees to the right.	Rotate Block Right
rotate-pink-block-right	Rotate the pink Block 90 degrees to the right.	Rotate Block Right
rotate-red-block-left	Rotate the red Block 90 degrees to the left.	Rotate Block Left
rotate-blue-block-left	Rotate the blue Block 90 degrees to the left.	Rotate Block Left
rotate-pink-block-left	Rotate the pink Block 90 degrees to the left.	Rotate Block Left
push-red-block-right	Push the red block right.	Push Block Right
push-blue-block-right	Push the blue block right.	Push Block Right
push-pink-block-right	Push the pink block right.	Push Block Right
push-red-block-left	Push the red block left.	Push Block Left
push-blue-block-left	Push the blue block left.	Push Block Left
push-pink-block-left	Push the pink block left.	Push Block Left
move-slider-right	Slide the Door to the right.	Slider Door Right
move-slider-left	Slide the Door to the left.	Slider Door Left
open-drawer	Open the Drawer.	Open Drawer
close-drawer	Close the Drawer.	Close Drawer
lift-red-block-table	Lift red Block from Table.	Lift Block from Table
lift-blue-block-table	Lift blue Block from Table.	Lift Block from Table
lift-pink-block-table	Lift pink Block from Table.	Lift Block from Table
lift-red-block-slider	Lift red Block from Slider.	Lift Block from Slider
lift-blue-block-slider	Lift blue Block from Slider.	Lift Block from Slider
lift-pink-block-slider	Lift pink Block from Slider.	Lift Block from Slider
place-in-slider	put the grasped object in the slider.	Put in Slider
place-in-drawer	put the grasped object in the drawer.	Put in Drawer
push-into-drawer	push the object into the drawer.	Push into Drawer
stack-block	stack blocks on top of each other.	Stack Block
unstack-block	go to the tower of blocks and take off the top one.	Unstack Block
turn-on-lightbulb	toggle the light switch to turn on the light bulb.	Turn on Lightbulb
turn-off-lightbulb	toggle the light switch to turn off the light bulb.	Turn off Lightbulb
turn-on-led	push the button to turn on the green light.	Turn on LED
turn-off-led	push the button to turn off the green light.	Turn off LED

Table 14. **Overview of tasks in CALVIN Dataset.**

K.2. Word Clouds

To visually summarize the key aspects covered by the diverse manipulation skills in ManiSkill2, CALVIN, and FrankaKitchen benchmarks, we created the word cloud in Figure 11. We now tokenize the instruction and for each skill code used in the trajectory, record *all* the tokens from the language instruction. Once we have this mapping from skills to tokens, we can plot heat maps and word clouds. The predominant terms reflect a heavy focus on interacting with household objects like boxes, cans, bottles, tools, and tableware. Terms such as ”pick up”, ”move”, ”rotate”, and ”stack” indicate the skills that aim to test fundamental robot capabilities in grasping, manipulating, and placing common items. The emergence of words depicting spatial reasoning like ”left/right” and goal configurations highlights skills requiring contextual understanding and mapping instructions to feasible actions.

Overall the cloud highlights the emphasis on generalizable skills for everyday human environments grounded in dexterous multi-step object handling, language grounding, and adaptable policies. The variety of objects, interfaces and multi-stage behaviors pose active challenges in embodied AI requiring both robust perception and control. The word distributions visualize the scope of the benchmarks focused on manipulating objects in human spaces by instruction in unpredictable scenarios.

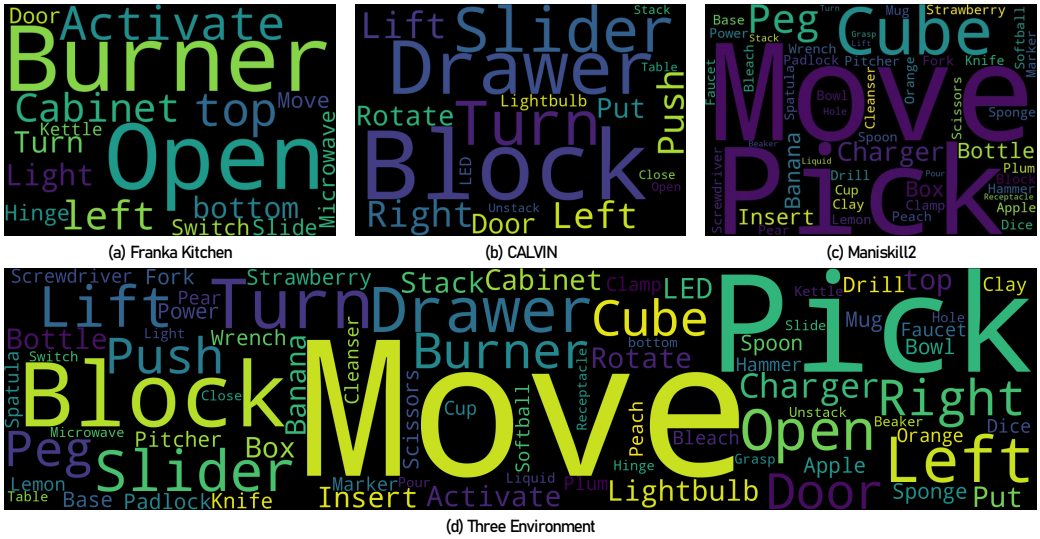

Figure 11. **World Cloud:** We created the word cloud to visually summarize the key aspects covered by the diverse manipulation skills in ManiSkill2, CALVIN, and FrankaKitchen benchmarks.

K.3. The t-SNE visualization of skills with KISA

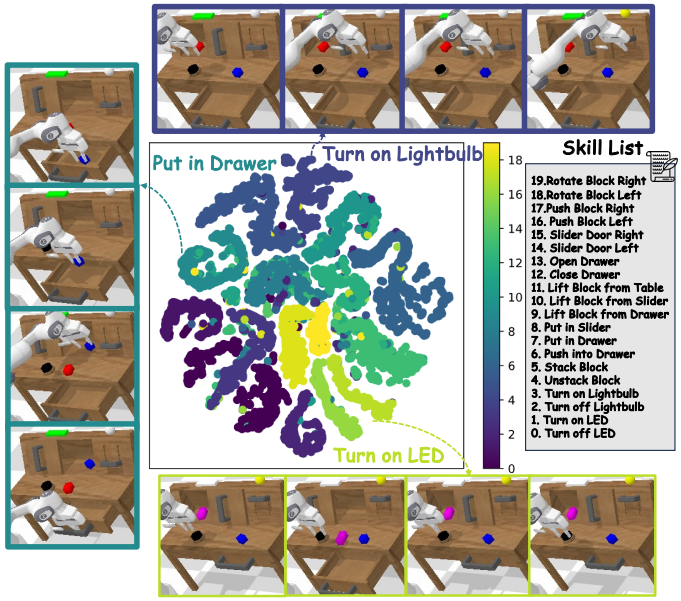

Figure 12. **t-SNE Visualization of Skill Space in CALVIN** reveals the relationship between the defined skills based on their visual dynamics. Each point denotes one skill type, colored by category. The modular relationships indicate promise for the zero-shot composition of new behaviors by leveraging similarity within the embedding space during policy learning.

L. Limitation & Future Work

Limitation Even though the skill library is not restricted and can be infinitely expanded, one of the limitations of KISA is that the current skill annotation is to retrieve from the skill library based on alignment score and lacks the ability of open-vocabulary generation for skill annotation. A potential solution is to leverage the multi-modal large language model (MLLM) to equip with the generation ability for more diverse skills. Another limitation is that KISA is now only finetuned in manipulation demonstrations in simulators with groundtruth keyframe labels. Although KISA already possesses a certain level of zero-shot generalization ability across embodiments, if KISA aims to gain the capacity to annotate more internet-scale diverse videos for more wildly usage, it needs to consider how to utilize more real human unlabeled data for self-supervised fine-tuning or continual fine-tuning.

Future Work Moreover, there are many interesting directions for potential extension based on KISA, as future works are to be explored. For example, the temporal-enhanced representation in KISA can also be a powerful plug-and-play representation integrated into existing learning pipelines. The temporal-enhanced representationKISA can explicitly encode smooth temporal task progress in their embedding distances. By detecting progress toward skill completion, we can trigger transitions based on representation cues rather than fixed decision intervals. This may potentially enhance flexibility compared to previous hierarchical imitation learning approaches that were constrained by pre-defined horizons. Another potential usage is to leverage KISA to detect incomplete or unsuccessful subtasks to prune suboptimal trajectories, as a demonstration filter to facilitate robust policy learning from sub-optimal demonstrations.

Anyway, we believe that whether it's KISA as a reliable annotation tool itself or the fine-grained video-level representation it offers, both could bring interesting insights and conveniences to the research in the robotics community.

M. More visualization example across different benchmarks

M.1. Visualization example in Maniskill

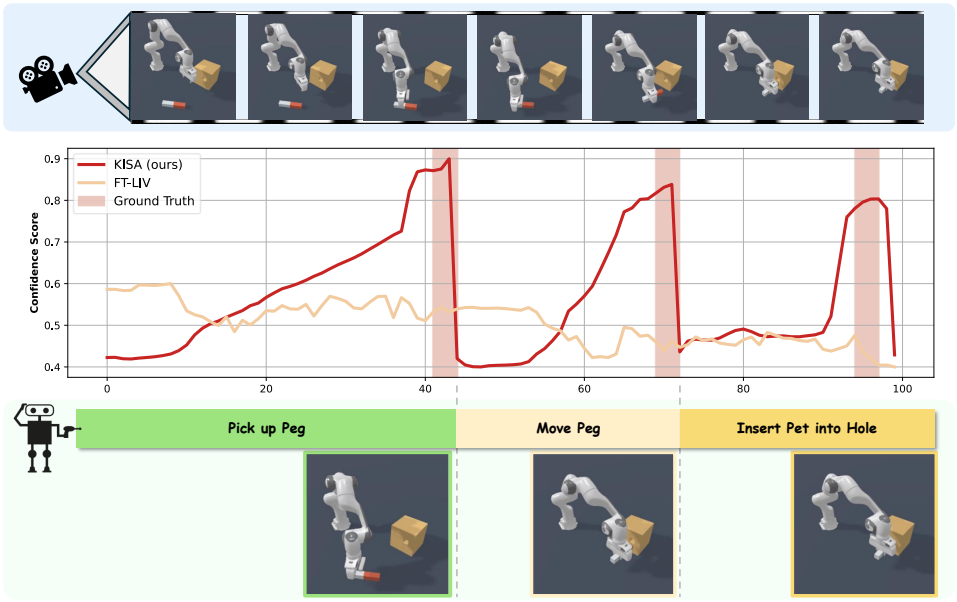

Figure 13. **Maniskill Example 1:** Pick up the peg, move the peg, and insert the peg into the hole.

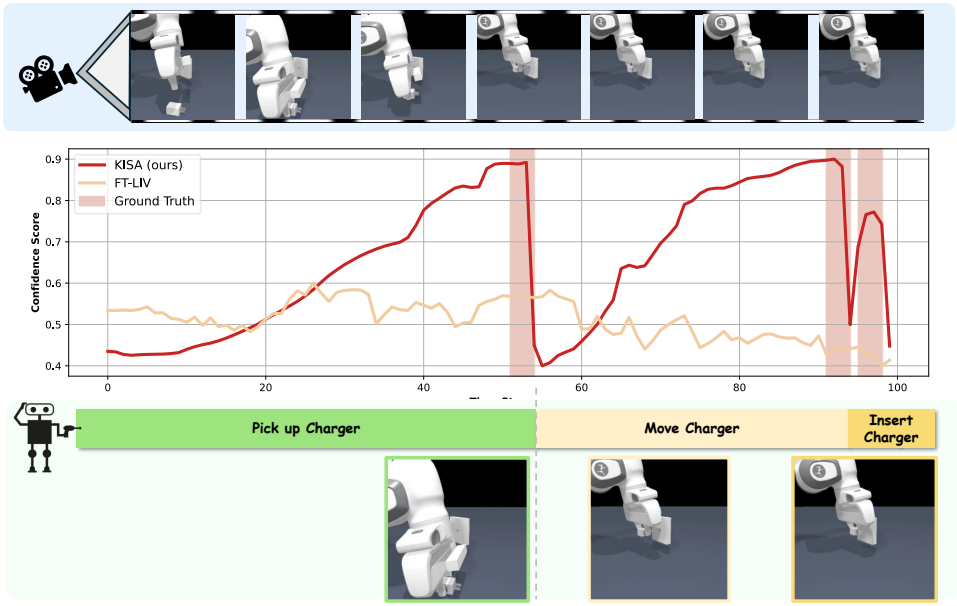

Figure 14. **Maniskill Example 2:** Pick up the charger, move the charger, and insert the charger into Receptacle.

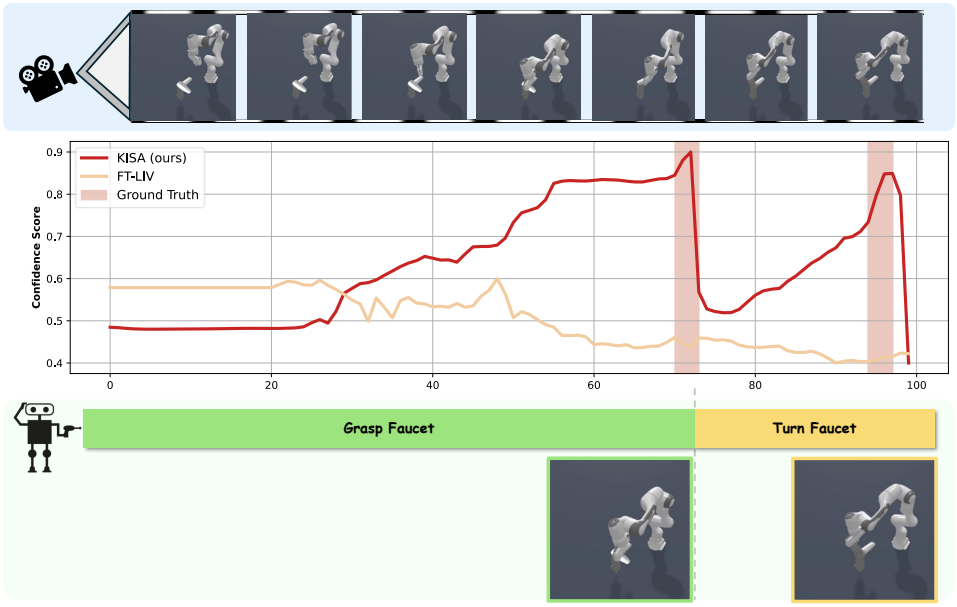

Figure 15. **Maniskill Example 3:** Grasp faucet and rotate faucet.

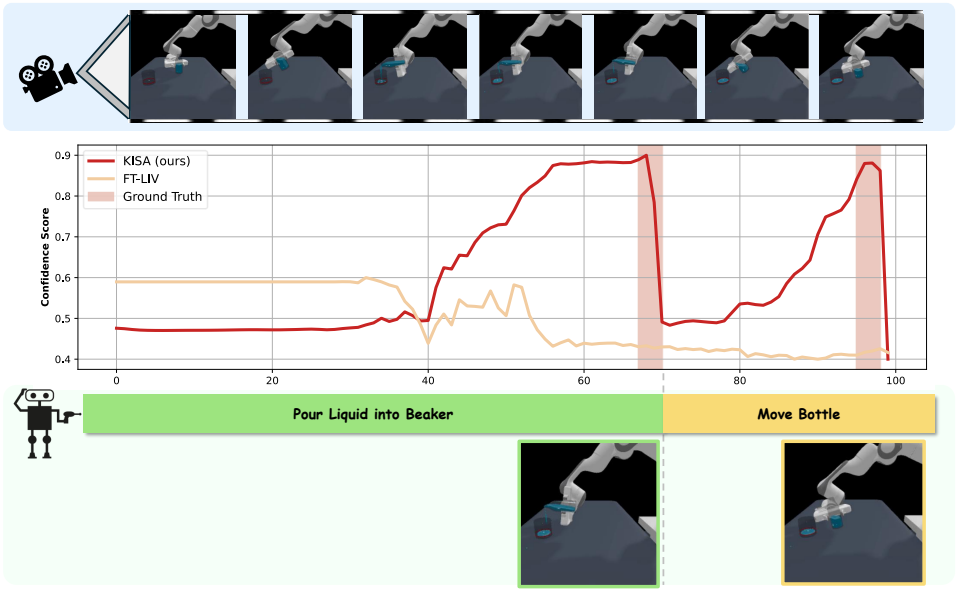

Figure 16. **Maniskill Example 4:** Pour liquid into a beaker and move the bottle to the upright position at the end.

M.2. Visualization example in CALVIN

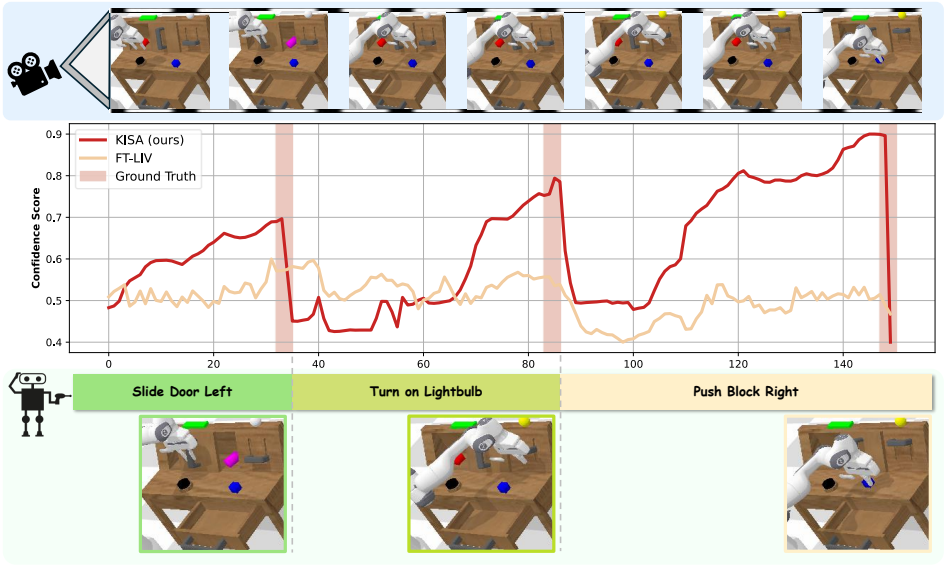

Figure 17. **CALVIN Example 1:** Slide door left, turn on lightbulb, and push block right.

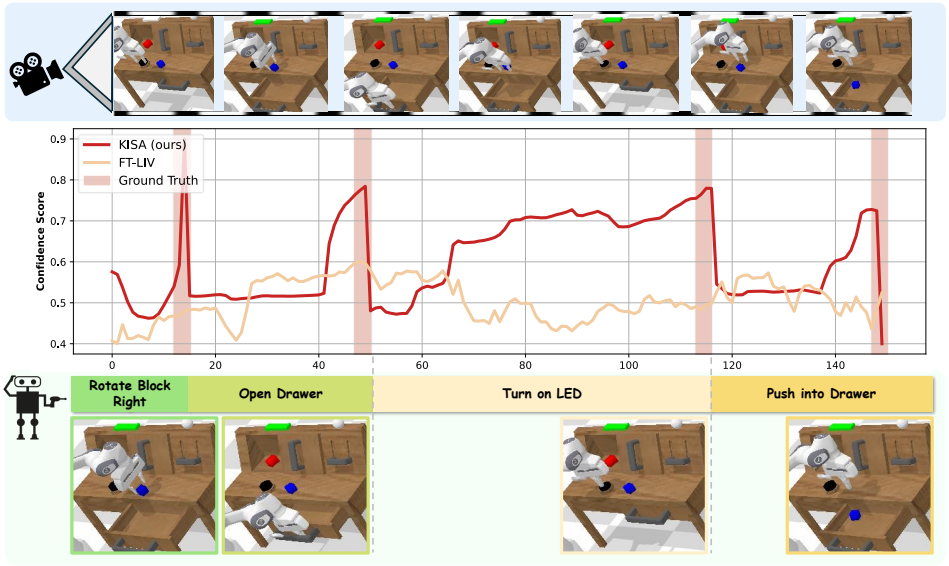

Figure 18. **CALVIN Example 2:** Rotate block right, open drawer, turn on led and push into drawer.

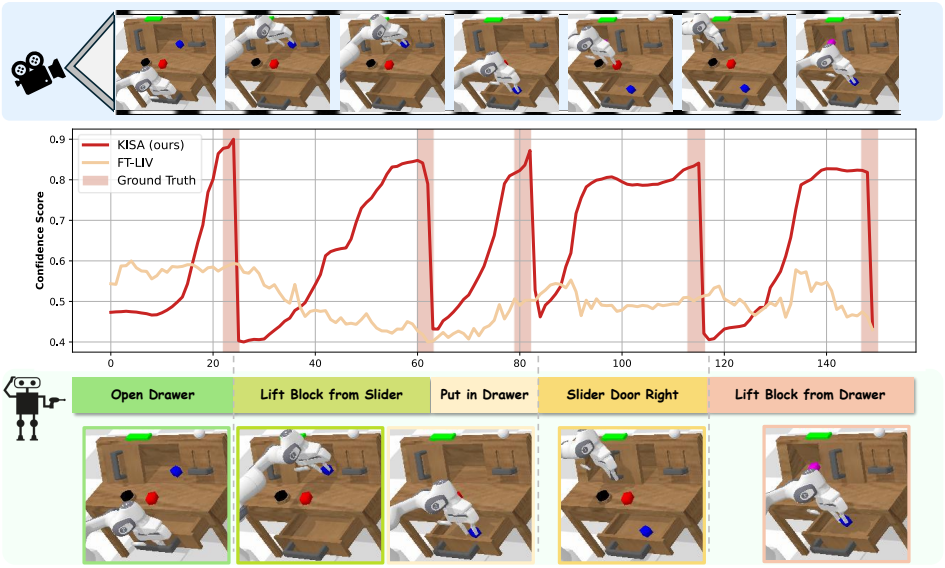

Figure 19. **CALVIN Example 3:** Open drawer, lift block from slider, put block in drawer, slide door right and lift block from drawer.

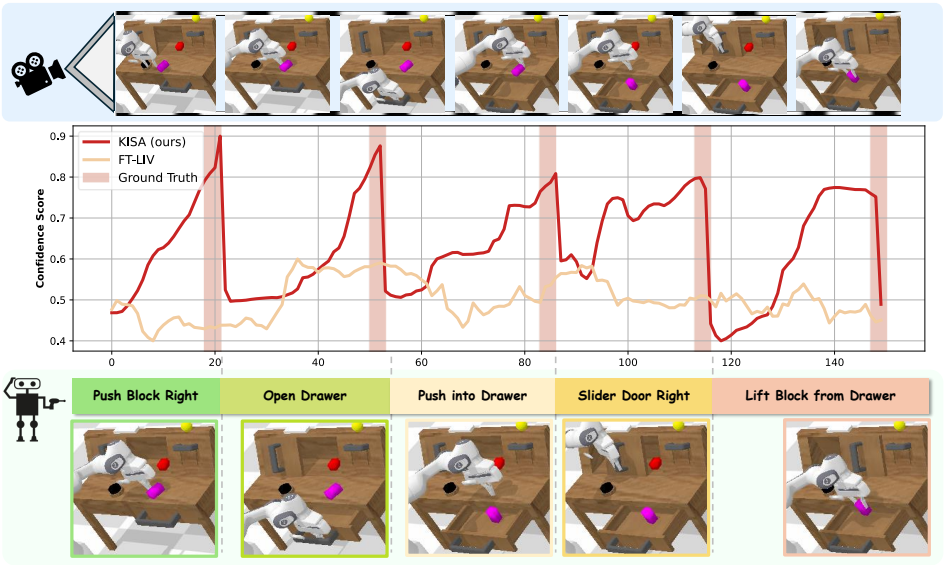

Figure 20. **CALVIN Example 4:** Push block right, open drawer, push block into drawer, slide door right and lift block from drawer.

M.3. Visualization example in FrankaKitchen

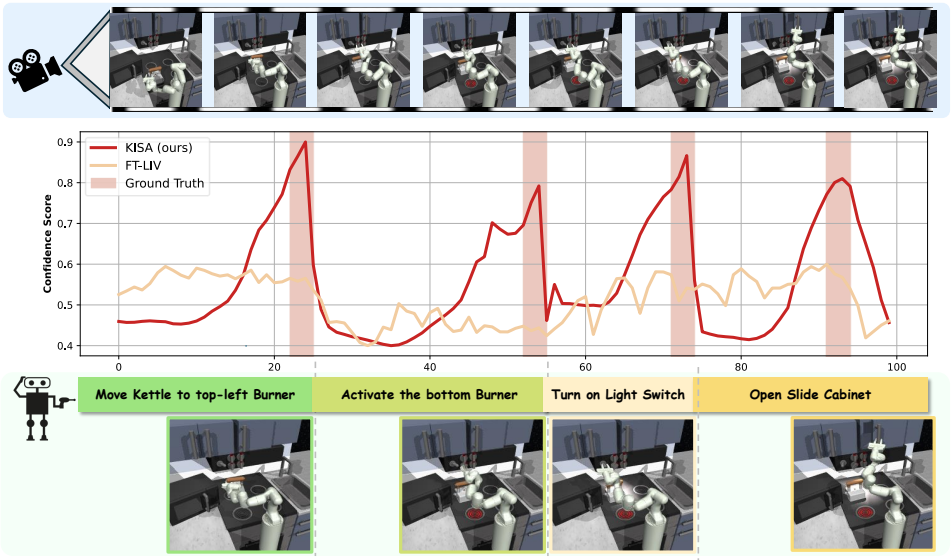

Figure 21. **FrankaKitchen Example 1:** Move the kettle to the top-left burner, activate the bottom burner, turn on the light switch, and open the slide cabinet.

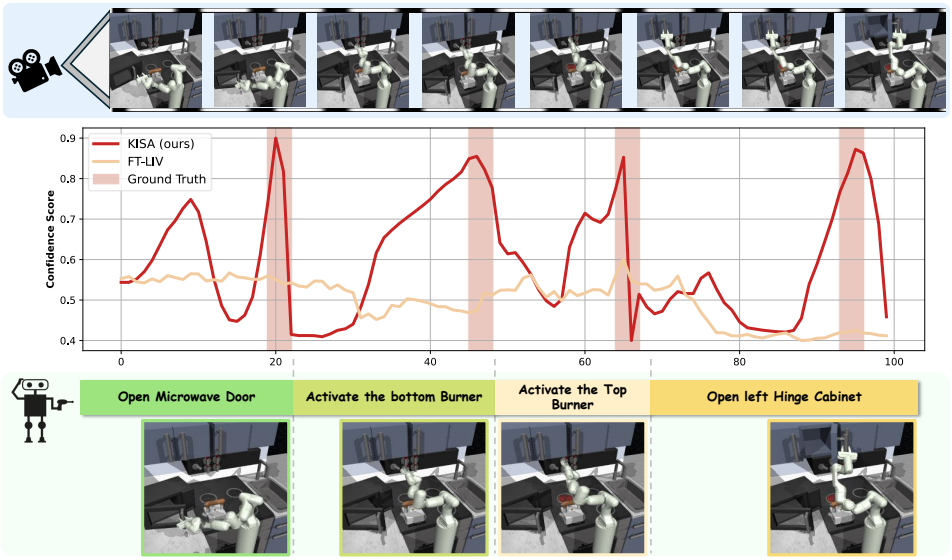

Figure 22. **FrankaKitchen Example 2:** Open the Microwave Door, activate the bottom burner, activate the top burner and Open the left Hinge Cabinet.

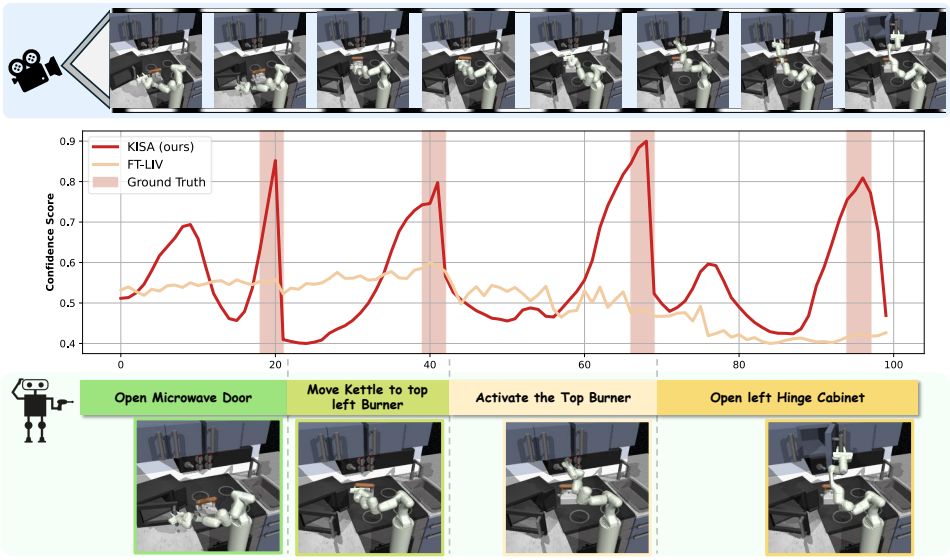

Figure 23. **FrankaKitchen Example 3:** Open the microwave door, move the kettle to the top-left burner, activate the top burner, and open the left hinge cabinet.

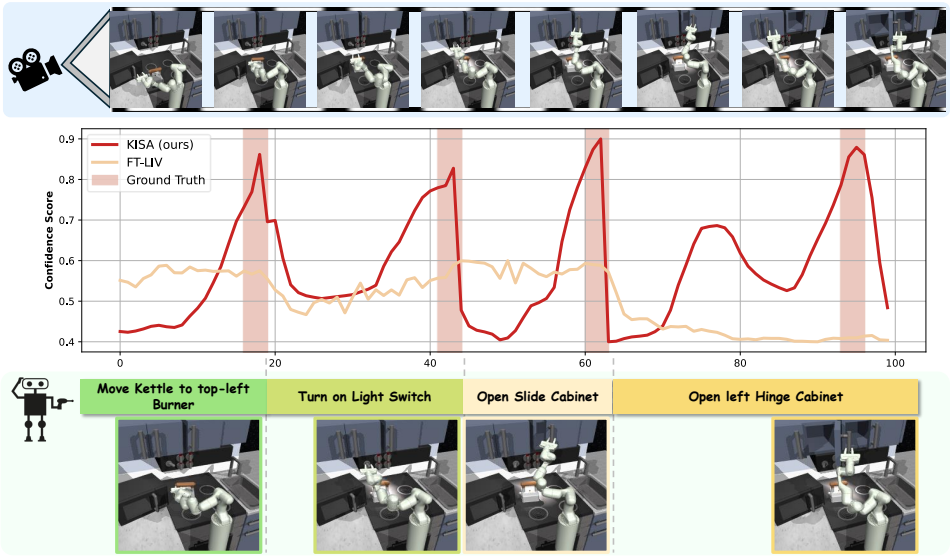

Figure 24. **FrankaKitchen Example 4:** Move the kettle to the top-left burner, turn on a light switch, open the slide cabinet, and open the left hinge cabinet.

M.4. Object Generalization

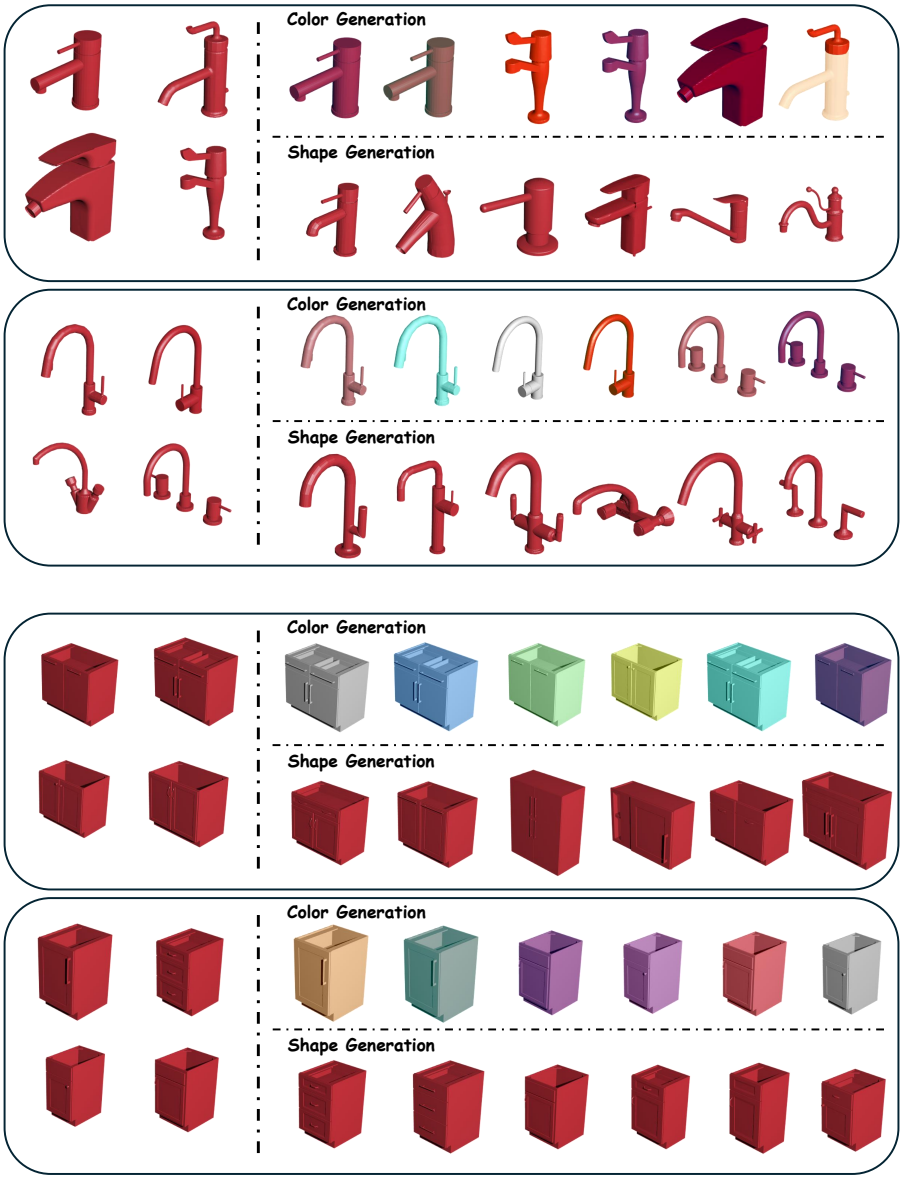

Figure 25. **Zero-shot Generalization across Objects Setting.**The visualization example of various color and shapes in Maniskill.

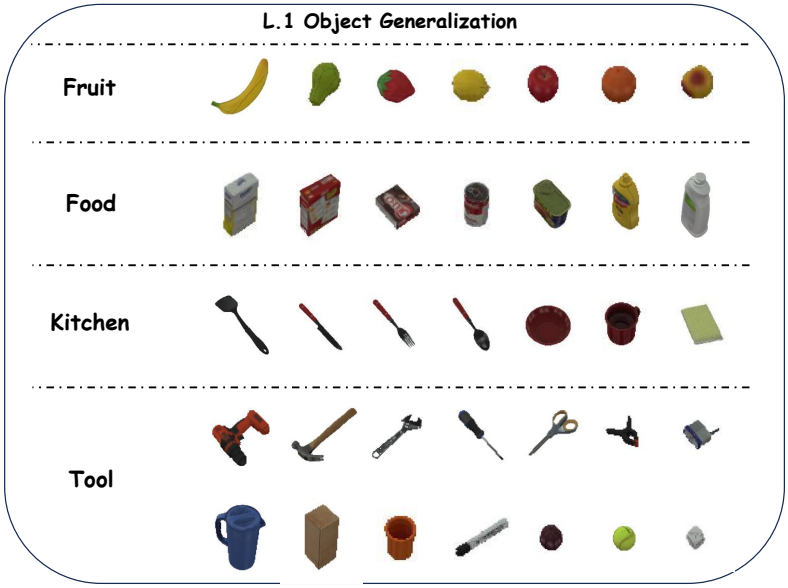

Figure 26. **Zero-shot Generalization across Objects Setting.**The visualization example of various objects in Maniskill.

Figure 27. **Zero-shot Generalization across Objects Setting.**The demonstration example of PickYCB task in Maniskill. The skills concludes the *pick up* {*YCB Object*} and *move* {*YCB Object*}, with various {YCB Object} showed in Figure 26.