Learning Object State Changes in Videos: An Open-World Perspective

Zihui Xue[1,2] Kumar Ashutosh[1,2] Kristen Grauman[1,2]
[1]The University of Texas at Austin [2]FAIR, Meta

Abstract

Object State Changes (OSCs) are pivotal for video understanding. While humans can effortlessly generalize OSC understanding from familiar to unknown objects, current approaches are confined to a closed vocabulary. Addressing this gap, we introduce a novel open-world formulation for the video OSC problem. The goal is to temporally localize the three stages of an OSC—the object's initial state, its transitioning state, and its end state—whether or not the object has been observed during training. Towards this end, we develop VIDOSC, a holistic learning approach that: (1) leverages text and vision-language models for supervisory signals to obviate manually labeling OSC training data, and (2) abstracts fine-grained shared state representations from objects to enhance generalization. Furthermore, we present HowToChange, the first open-world benchmark for video OSC localization, which offers an order of magnitude increase in the label space and annotation volume compared to the best existing benchmark. Experimental results demonstrate the efficacy of our approach, in both traditional closed-world and open-world scenarios.[1]

1. Introduction

In video understanding, the study of objects primarily revolves around tasks like recognition [16], detection [24], and tracking [8, 62], with the assumption that objects maintain a consistent visual appearance throughout the video. Yet, objects in video are often dynamic. They can undergo transformations that change their appearance, shape, and even topology. For example, a pineapple goes from whole to peeled to sliced, or wood is carved into a new shape.

Object State Changes (OSCs) [2, 15, 17, 22, 47, 50–52, 60, 64] add a critical dimension to video understanding. On the one hand, they provide insights into human actions—observing a piece of metal being shaped into a hook, for instance, implies the action of bending; observing an egg shell go from whole to broken implies the action of cracking. On the other hand, OSCs are essential for assessing goal completion, in a way that is invariant to the specific procedure used. For instance, the readiness of cake batter signifies the completion of a mixing task, regardless of whether it was stirred by hand or a mechanical mixer. These core functionalities are instrumental for various real-world applications [5, 6, 12, 13, 20, 21, 66], ranging from AR/VR assistants [42] that guide users through complex tasks by monitoring object states, to robotic manipulation [9, 55] where understanding the state of objects is critical for task planning and failure recovery.

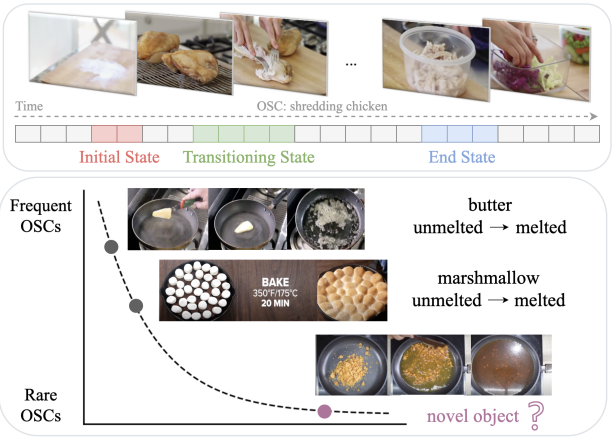

Figure 1. Top: The video OSC objective is to temporally localize an object's three states (i.e., initial, transitioning, end). Bottom: OSCs naturally exhibit a long tail. Certain OSCs, such as melting butter or marshmallow, are frequently showcased in instructional videos while others like melting jaggery might be rarely seen. We introduce an innovative open-world formulation that requires extrapolating to novel objects never encountered during training.

However, the development of video OSC is still at a primitive stage. Existing approaches [1, 2, 15, 46, 50, 51, 58] assume a **closed vocabulary**, where each state transformation is associated with a limited set of objects—often just 1 or 2. For example, the concept of "melting" might be limited to familiar items like butter and marshmallow. Consequently, the learned models are only capable of identifying state changes for objects observed during training and stumble when presented with novel objects. In contrast, in real-world situations a single state transition like "melting" can be linked with a plethora of objects—some ubiquitous and

[1]Project webpage: https://vision.cs.utexas.edu/projects/VidOSC/.

others rare. Importantly, there is an intrinsic connection that threads these OSC processes together. As humans, even if we have never encountered an unusual substance (e.g., jaggery) before, we can still understand that it is experiencing a "melting" process based on its visual transformation cues (see Fig. 1, bottom).

In light of these limitations, we propose a a novel open-world formulation of the video OSC problem. We formally characterize OSCs in videos in terms of three states that must be temporally localized: initial, transitioning, and end (see Fig. 1, top). In our **open-world setting**, there are *known* objects and *novel* objects. The model is only presented with known objects during training (e.g., frying chicken, frying onions). Then, it is evaluated on both known and novel objects (e.g., frying cauliflower). In addition to this fundamental open-world generalization, we also tackle a more comprehensive notion of the transitioning state. Specifically, our transitioning states encapsulate not only action-induced modifications to the object (e.g., peeling), but also passive transformations the object undergoes without human interference (e.g., melting) and edited-out transformations (e.g., the cake goes from raw to baked even if baked off camera). Though largely overlooked in existing approaches [1, 15, 46, 58], these cases are both common and technically interesting, since they force a model to reason about the *effects* of the core state change rather than search for evidence of a human actor carrying out the action.

Armed with this new problem formulation, we propose a holistic video OSC learning approach, anchored by two innovative ideas. First, we explore text and vision-language models (VLMs) for supervisory signals during training. We pioneer the use of textual state descriptions to generate a long tail of object state pseudo-labels, by leveraging the remarkable capabilities of VLMs [23, 44, 63] and large language models (LLMs) [40, 53, 54]. This strategy eliminates the need for exhaustive label collection for training data, facilitating large-scale model training.[2] Second, to confront the open-world challenge, we propose object-agnostic state prediction, consisting of three key techniques: a shared state vocabulary to unify the label space across objects sharing the same state transition (e.g., melting butter and melting marshmallow), temporal modeling to grasp the progression of state changes over time, and object-centric features that better represent the objects during an OSC. These designs equip our model with the ability to generalize state understanding from known objects to novel ones. We term our approach VIDOSC.

Complementing this, we present a large-scale real-world dataset HowToChange. Sourced from the HowTo100M collection [36], it sets a new benchmark in the field of unprecedented scale and an authentic long-tail distribution, setting it apart from the constrained scope and closed-world setting of earlier benchmarks (see Table 1). Finally, experimental results demonstrate the efficacy of VIDOSC, surpassing the state-of-the-art in both traditional closed-world and novel open-world scenarios by a great margin.

[2]Note that at test time, our model requires only the video, with no need for additional text, ensuring utmost flexibility and applicability.

2. Related Work

Object State Changes Image-based methods explore the compositional nature of objects and their attributes, including zero-shot recognition of unseen combinations [22, 29, 33, 37–39, 41, 43], but do not consider the temporal progression of OSCs in video, which brings new challenges. Video-based methods develop human-centric models that leverage state change cues to facilitate action recognition [1, 15, 46, 58], or explore joint discovery of object states and actions [2, 50, 51]. Notably, all the existing methods assume a closed world, recognizing only the OSC categories ("known" objects) seen during training. Our approach distinguishes itself in three crucial ways: (1) we introduce a novel open-world[3] formulation, where the objects encountered during evaluation can be entirely unseen during training; (2) we adopt an object-centric perspective, allowing for scenarios where OSCs occur with no observable human actions in the video; and (3) we propose to utilize the text modality and VLMs as supervisory signals, greatly scaling up model training and boosting performance.

Early datasets for video OSC capture a limited array of OSCs in fewer than 1,000 videos [2, 31]. The more recent ChangeIt dataset [50] marks an advance in dataset scale (34K videos), yet is still restricted to a small OSC vocabulary of 44 total object-state transitions. It lacks the scope to adequately test unseen objects, since most state changes coincide with only 1 or 2 objects across all the videos (see Table 1). Other datasets explore different aspects of video OSCs, including object state recognition [47], point-of-no-return (PNR) frame localization [17], and object segmentation [52, 64], but they lack any temporal annotations of fine-grained OSC states needed for our task's evaluation. Our HowToChange dataset increases the OSC vocabulary space by an order of magnitude and allows for the first time rigorous study of the open-world temporal OSC challenge.

Vision and Language LLMs [40, 53, 54] have revolutionized various research fields with their exceptional performance across diverse applications. Building on this momentum, the use of web-scale image-text data has emerged as a powerful paradigm in computer vision, with powerful VLMs now advancing an array of image tasks, including zero-shot classification [44], detection [18], segmentation [32] and visual question answering [27, 28]. Similarly, joint video-text representation learning [3, 30, 36, 67] has been advanced by multimodal datasets like

[3]also called "unseen compositions" in object-attribute learning [34, 37].

HowTo100M [36] and Ego4D [17], which offer large-scale collections of videos paired with their corresponding narrations. These datasets have also facilitated tasks like step discovery [11], localization in procedural activities [35] and long egocentric videos [45]. The vision-language multimodal cycle consistency loss proposed in [14] helps discover long-term temporal dynamics in video. In line with these developments, we propose to leverage the text accompanying instructional videos as well as existing high-performing VLMs to provide supervisory signals that guide the training of our video OSC model and allow learning for the open world.

Learning in an Open World The open world setting has received increasing attention, predominantly within the image domain for object recognition [4, 48], detection [10, 18, 25, 65] and object-attribute compositionality [29, 33, 37–39, 41, 43]. In the video domain, compositionality is advanced by the Something-Else dataset [34], where the training combinations of verbs and nouns do not overlap with the test set, sparking work on the dynamics of subject-object interactions [7, 34, 47]. More recent efforts leverage VLMs for zero-shot action recognition [7, 26, 57]. Concurrent work explores video object segmentation with state-changing objects [64] and recognition of novel object-state compositions for food chopped in different styles [47]. Despite these advances, temporal video OSC understanding in the open world remains unexplored. Our open-world formulation requires generalizing the temporal localization of fine-grained object states from known to novel objects.

3. Approach

We present the VIDOSC framework for learning video OSCs, detailing the open-world formulation (Sec. 3.1), model design for object-agnostic state prediction (Sec. 3.2), and text-guided training scheme (Sec. 3.3).

3.1. Video OSC in the Open World

We begin by formally defining an object state change (OSC) in videos as a visually detectable transformation, where an object experiences a change that is not easily reversible, in line with [17]. We characterize an *OSC category* as an object combined with a specific state transition, such as "chicken + shredding", and delineate an OSC process through three distinct states: initial (precondition state), transitioning, and end (postcondition state).[4] It is essential to highlight that we take an object-centric perspective: the "transitioning state" accounts for instances where the video depicts an active action applied to the object, as well as scenarios where the object undergoes a passive transformation absent of human intervention (e.g., melting, drying).

[4]Due to real-world data variation, some videos may lack one of the OSC states, and there may be multiple segments in a video corresponding to the same state. Our formulation accounts for all these scenarios.

Given a video in which an object may be changing state, the objective is to temporally localize each of the three OSC states. Consistent with prior work [50, 51], we formulate the task as a frame-wise classification problem. Formally, a video sequence is represented by a temporal sequence of T feature vectors $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_T\}$, where T denotes the video duration. The goal is to predict the OSC state label for each timepoint, denoted by $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_T\}$, $\mathbf{y}_t \in \{1, \cdots, K+1\}$, where K is the total number of OSC states, and there is one additional category representing the background class not associated with any OSC state.[5]

Next, we propose an open-world problem formulation, capturing the intrinsic long-tail distribution observed in real-world scenarios. Consider N state transitions, each paired with a set of associated objects, denoted by $\mathbf{O}^n = \{o_1^n, o_2^n, \cdots, o_{m(n)}^n\}$, for $n = \{1, 2, \cdots, N\}$, where $m(n)$ denotes the number of objects linked with the n-th state transition. Within a specific state transition (like frying), certain objects in the set (such as chicken or onions) are frequently observed, while the combination of the same state transition with other objects (such as cauliflower) appear less often. Motivated by this inherent long-tail, we propose to split $\mathbf{O}^n$ into two distinct subsets: $\mathbf{O}^n_{\text{known}}$ covering the common combinations observed during training, and $\mathbf{O}^n_{\text{novel}}$ comprising the infrequent combinations, which are unseen during training due to their rarity. During inference, the model is evaluated on the entire object set $\mathbf{O}^n$ for a comprehensive reflection of the open-world setting.

3.2. Object-Agnostic State Prediction

To address the open-world challenges and ensure robust generalization to novel objects, VIDOSC integrates three key techniques: (1) a shared state vocabulary that consolidates the state label space; (2) temporal modeling within the video OSC model design; and (3) object-centric features.

Shared State Vocabulary The inherent link among different OSC categories is overlooked in prior work. Common practice involves developing separate models for each OSC category [2, 50] (e.g., one model exclusively for melting butter and another for melting chocolate), or using one single model that still treats every OSC as a distinct entity in the label space [51] (e.g., considering melted butter and melted chocolate as two separate end states). Such a formulation results in an extensive label set with $K = 3 \times \sum_{n=1}^{N} m(n)$, where 3 denotes the number of OSC states (i.e., initial, transitioning and end) and $\sum_{n=1}^{N} m(n)$ is the total number of OSC categories. This compartmentalized view can inadvertently hinder the model's generalization ability. Yet, on closer inspection, states across varied objects are intrinsically related. For instance, the "melting" principle remains consistent, even when applied to visually

[5]Not all training videos may feature an OSC due to data collection noise, yet all evaluation videos are manually verified to include one.

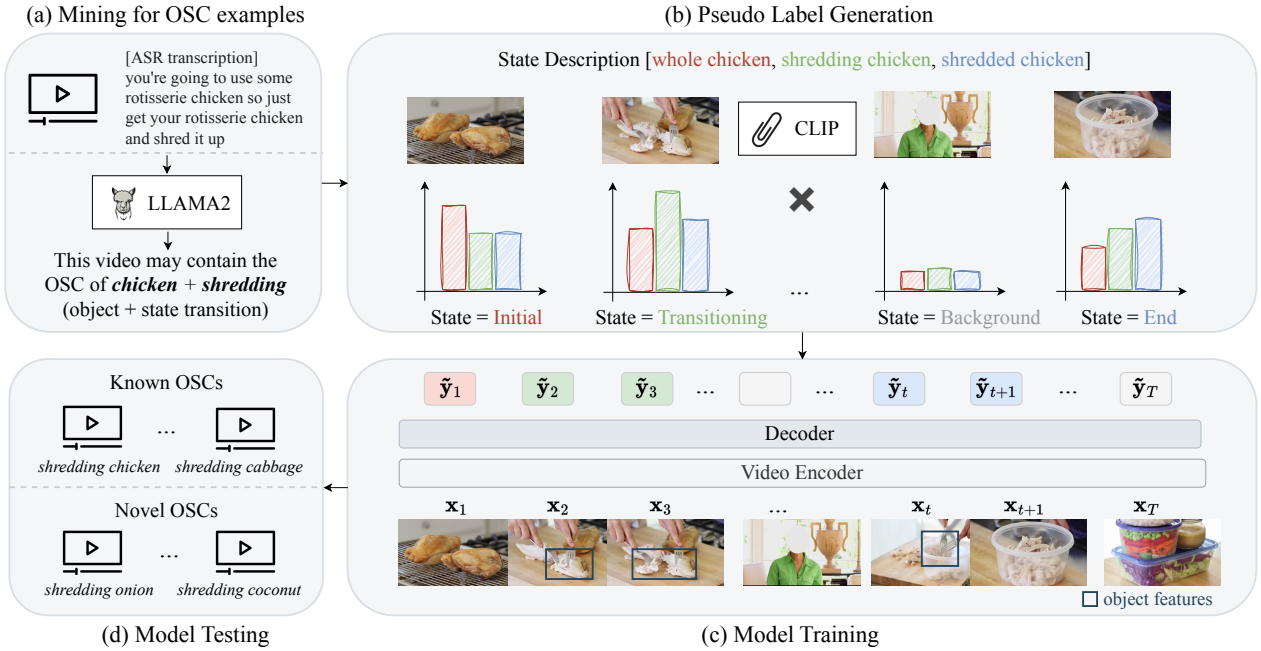

Figure 2. Our proposed VIDOSC framework: (a) Mining for OSC examples (Sec. 3.3): We leverage ASR transcriptions paired with videos and the capabilities of LLM to automatically mine OSC examples. (b) Pseudo Label Generation (Sec. 3.3): We utilize textual state descriptions and a VLM for supervisory signals during training; (c) Model Training (Sec. 3.2): We develop a video model for object-agnostic state prediction. (d) Model Testing (Sec. 3.1): We propose an open-world formulation, evaluating on both known and novel OSCs. Notably, while we employ the text modality to guide model training, our model is purely video-based and requires no text input at the test phase, ensuring maximum flexibility and applicability. Ground truth for the test set is manually annotated.

distinct objects like butter and chocolate. This motivates us to group OSC labels that share the same state transition as one, and train a single model per state transition.[6] By adopting object-agnostic OSC labels, we encourage the model to learn state representations shared by visually different objects, and thus facilitate the transfer from known to novel objects.

Temporal Modeling Recognizing that state changes unfold over time, it is important to capture the temporal dynamics in videos. An object's state at any frame, be it initial, transitioning, or end, is often best understood in the context of preceding and succeeding frames. Contrary to prior works [2, 50, 51] that rely on isolated frame-wise modeling without considering the temporal progression of video OSCs, we address this gap by proposing a temporally-aware model design. As illustrated in Fig. 2(c), we adopt an encoder-decoder architecture. For the encoding phase, input video features $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_T\}$ are first projected using an MLP f_{project}, and augmented with sinusoidal positional embeddings, $\mathbf{Z}_{\text{pos}}$. Subsequently, a transformer encoder [56] $f_{\text{transformer}}$ is adopted to capture the temporal dynamics among these features, yielding $\mathbf{Z} = f_{\text{transformer}}(f_{\text{project}}(\mathbf{X}) + \mathbf{Z}_{\text{pos}})$. For the decoding phase, a MLP decoder g, maps these temporally-aware hidden representations to OSC state predictions: $\tilde{\mathbf{Y}} = g(\mathbf{Z})$. See Sec. 5 for architecture and training details. This design ensures that when predicting the state for a frame, the model has assimilated temporal context from the entire sequence. Essentially, our model exploits the fact that the *dynamics* of the state change has greater invariance to the object category than how the object looks, emphasizing temporal object transformations over objects' static appearances, and thereby enhancing generalization to novel OSCs.

Object-Centric Features Finally, we discuss how to better represent "object" features in the problem. In many scenarios, due to camera placement and framing, the object going through a state transition might only occupy a small portion of the frame, surrounded by other visual elements such as background, people, or bystander objects. Recognizing this challenge, we introduce an enhancement to our model's input features $\mathbf{X}$ to emphasize the object of interest. To be specific, we leverage an off-the-shelf detector [49] to identify the active object (i.e., the object being manipulated) region at each timepoint t, yielding feature $\mathbf{x}_t^{obj}$ that centers on the object (see bounding boxes in Fig. 2 (c)). The input feature is then constructed as a concatenation of the original global feature and the localized object-centric feature,

[6] See Supp. for the multi-task model variant, where we develop one unified model for all state transitions.

i.e., $\mathbf{X} = \{[\mathbf{x}_t, \mathbf{x}_t^{obj}]\}_{t=1}^{T}$. By emphasizing the object in this manner, our model is better positioned to discern the intricate state changes and provide more informed predictions.

3.3. Text and VLMs as Supervision

To scale up training and ensure broad generalization, we propose a novel training pipeline that leverages LLMs and VLMs for OSC mining and pseudo-labeling.

Mining for OSC examples Utilizing a vast collection of "how-to" instructional videos as the training source, we develop an automated mining process to capture the rich, real-world variability of OSCs. The motivation is that instructional videos usually have accompanying Automatic Speech Recognition (ASR) transcriptions that offer valuable OSC cues. For example, a speaker mentioning, "so just get your rotisserie chicken and shred it up" suggests that the chicken may undergo a state transition of shredding in the video. Leveraging this fact, we employ LLAMA2 [54] to analyze ASR transcriptions for identifying candidate videos and their associated OSC categories. This text mining stage allows us to corral a long tail of OSCs, discovering likely state change terms—even if rare—from the free-form natural language narrations given by how-to instructors in the training videos. See Fig. 2 (a).

Pseudo Label Generation Next, to negate the need of manually labeling large-scale training data, we propose a novel pseudo-labeling pipeline facilitated by VLMs. From the identified OSC category (e.g., shredding chicken), we form three *textual state descriptions* for its initial, transitioning, and end states (e.g., whole chicken, shredding chicken, and shredded chicken). We then adopt both the vision and language encoder from a well-trained VLM (we experiment with CLIP [44] and VideoClip [61]) to compute the cross-modal similarity between every frame in training video and the three state descriptions, producing a score matrix $\mathbf{S} \in \mathbb{R}^{T\times 3}$. The pseudo label $\hat{y}_t$ at timepoint t is then assigned based on this score matrix:

$$\hat{y}_t = \begin{cases} \text{Background} & \text{if } \sum(\mathbf{S}[t,:]) < \tau \\ \text{Initial} & \text{elif } \mathbf{S}[t,0] - \mathbf{S}[t,1] > \delta \text{ and } \mathbf{S}[t,0] - \mathbf{S}[t,2] > \delta \\ \text{Transitioning} & \text{elif } \mathbf{S}[t,1] - \mathbf{S}[t,0] > \delta \text{ and } \mathbf{S}[t,1] - \mathbf{S}[t,2] > \delta \\ \text{End} & \text{elif } \mathbf{S}[t,2] - \mathbf{S}[t,0] > \delta \text{ and } \mathbf{S}[t,2] - \mathbf{S}[t,1] > \delta \\ \text{Ambiguous} & \text{otherwise} \end{cases}$$

where δ is the threshold that separates states and τ is the threshold that differentiates between states and background. Essentially if the VLM scores some state more strongly than the other two—and the cumulative confidence score of all states is high—then it is adopted as the pseudo label. Otherwise it is omitted as ambiguous. See Fig. 2(b).

To further refine these labels, we enforce a causal ordering constraint. Given the inherent progression of OSCs, the anticipated order is initial states followed by transitioning states, and finally, the end states. Any frame whose pseudo label does not respect this natural progression is re-assigned to the ambiguous category. Our training employs a cross-entropy loss between pseudo label $\hat{y}_t$ and the corresponding model prediction $\tilde{y}_t$, with ambiguous frames excluded to maintain clarity and distinction among the state labels. See Section 4.2 in Supp. for detailed pseudo label analysis.

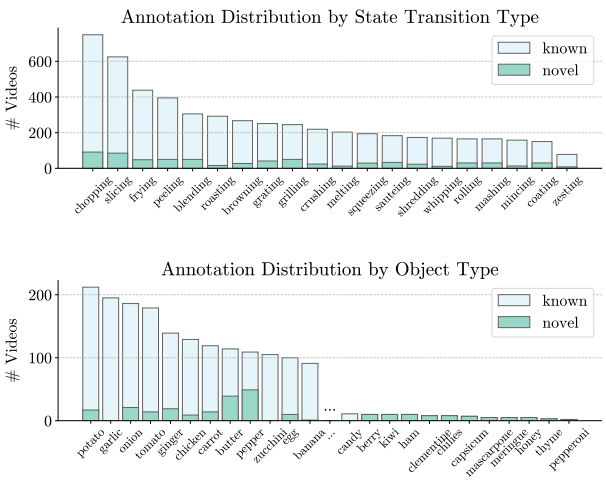

Figure 3. Ground truth annotation distribution across 20 state transitions (top) and 134 objects (bottom) in HowToChange (Evaluation). In line with our open-world formulation, annotations cover a diverse range of object-state transition combinations, categorized into known and novel OSCs.

4. The HowToChange Dataset

Existing OSC datasets fall short in capturing the open-world's diversity and long-tail distribution of OSCs. To address this gap, we introduce HowToChange. It encompasses varied state transitions coupled with a diverse range of objects, providing an authentic reflection of their real-world frequency—from the commonplace to the rare.

Data Collection The HowTo100M collection [36] contains a wealth of instructional videos that often feature OSCs and is particularly suitable for this task. We specifically focus on the HowTo100M Food & Entertaining category because (1) it constitutes a third of the entire HowTo100M videos, (2) cooking tasks offer a wealth of objects, tools, and state changes, providing an excellent test bed for open-world OSC, and (3) in cooking activities a single state transition can often be associated with a varied range of objects, opening the door to learning compositionality. (Note, we also experiment with non-cooking domains below.) We process a total of 498,475 videos and 11,390,287 ASR transcriptions with LLAMA2. From the responses, we identify the most frequently seen state transitions and objects associated with them to establish an OSC vocabulary, resulting in 134 objects, 20 state transitions, and 409 unique OSCs. The number of objects associated with each state transition ($m(n)$)

Datasets	# Obj	# ST	# OSC	ObjPer	# Videos	GT Label?
Alayrac et al. [2]	5	6	7	1.2	630	✓
Task-Fluent [31]	25	14	32	2.3	809	✓
ChangeIt (Training) [50]	42	27	44	1.6	34,428	✗
ChangeIt (Evaluation) [50]	42	27	44	1.6	667	✓
HowToChange (Training)	122	20	318	15.9	36,075	✗
HowToChange (Evaluation)	134	20	409	20.5	5,424	✓

Table 1. Comparison with existing video datasets focusing on object states. 'Obj' and 'ST' represent objects and state transitions, respectively. 'ObjPer' denotes the average number of objects associated with each state transition; higher values indicate more need to generalization across objects. We present the first open-world benchmark for temporal video OSC, with an order of magnitude increase in OSC vocabulary and annotation volume.

spans from 6 for "zesting" to 55 for "chopping". The state transitions applied to each object vary from 1 to 15, with onions being the most versatile. In total, we identify 36,075 videos for the training set of HowToChange, with an average duration of 41.2 seconds.

Data Splits In line with our open-world formulation, we divide the 409 identified OSCs into two disjoint subsets based on their frequency of occurrence. Within each state transition, we categorize the top 75% frequent objects as known and the bottom 25% as novel. This yields a total of 318 known OSCs that are seen during training and testing, spanning 20 state transitions associated with 122 objects, and 91 novel OSCs that are only seen during testing, encompassing the same transitions across 58 objects.

Ground Truth Label Collection (Evaluation Set) To facilitate thorough evaluation, we obtain manual annotations for a subset of 5,424 videos from the collected dataset.[7] The annotation workflow is as follows: each annotator is presented with a video segment along with an OSC category that was previously identified by LLAMA2. The annotator has the option to reject the video segment if it does not contain the specified OSC. Otherwise, they label the time ranges corresponding to the initial, transitioning, and end states of the OSC. Adhering to our object-centric emphasis, annotators are instructed to label based on the visual changes of the object, rather than human-centric actions, and exclude time ranges where the object of interest is not visible, ensuring clean and focused temporal labels. Fig. 3 provides the distribution of annotated videos. On average, we collect 271 annotations per state transition, with a video duration of 41.5 seconds, and 12.9% of videos belong to the novel category. The entire annotation process required around 1,507 hours by 30 professional annotators.

Dataset Comparison Table 1 compares HowToChange with existing video datasets on temporal OSC understanding. HowToChange offers an unprecedented scale—with 9.3x more OSC categories and 8.1x more annotated video instances compared to the previous largest collection [50]. Furthermore, notably, in prior datasets, each state transition is typically coupled with 1 or 2 objects, preventing subsequent models from generalizing to new objects, as we will see in results. In contrast, HowToChange pioneers the open-world formulation, presenting a much broader range of objects associated with each state transition—from 6 to 55 objects per state transition, and averaging 20. This facilitates the development of models with generalized OSC understanding. Please see Supp. for full data collection and annotation details.

5. Experiments

Datasets In addition to our new HowToChange dataset, we also evaluate on ChangeIt [50] due to its expansive data scale (34K videos spanning many activities) and high relevance to our task. Beyond the conventional split of ChangeIt, we propose a new split tailored to our open-world formulation. Specifically, from the 44 available OSC categories in ChangeIt, we concentrate on the 25 categories where each state transition is paired with more than one object. With those, we form the "ChangeIt (open-world)" subset that comprises 8 state transitions and 25 objects. Within each state transition, objects are randomly divided into known and novel categories, yielding 13 known OSC categories for training and 12 novel OSC categories exclusively reserved for evaluation. A detailed breakdown of this split can be found in Supp. To sum up, our evaluation encompasses: ChangeIt; ChangeIt (open-world); and HowToChange, offering a comprehensive setup in both closed-world and open-world scenarios.

Evaluation For ChangeIt and ChangeIt (open-world), we adhere to the original dataset's evaluation protocol [50], reporting action and state precision@1 as the evaluation metrics. For our dataset, besides precision@1, which evaluates a single frame for each state within a video, we advocate for the use of F1 score and precision over all frames to ensure a more holistic evaluation.

Baselines We compare our approach with four baselines across two categories: (a) self-supervised approaches on identifying object states enforced by the causal ordering constraint: LookForTheChange [50] trains a dedicated model for each OSC category, while MultiTaskChange [51] evolves this into a multi-task approach[8], catering to several OSCs concurrently; (b) zero-shot VLMs: image-based CLIP [44], video-based VideoCLIP [61] and InternVideo [59]. All baselines in (a) use the same training data as our model, whereas the zero-shot models (b) are directly

[7] Our training is purely guided by a VLM and requires no ground truth labels. The annotations are reserved exclusively for evaluation.

[8] To ensure a thorough evaluation, we train both single-task and multi-task variants of our approach. See Supp. for a detailed discussion.

Method	ChangeIt		ChangeIt (open-world)				HowToChange					
	State Prec.@1	Action Prec.@1	State Prec.@1		Action Prec.@1		F1 (%)		Prec (%)		Prec.@1 (%)	
			known	novel	known	novel	known	novel	known	novel	known	novel
CLIP [44]	0.30	0.63	0.29	0.29	0.71	0.70	26.9	25.4	27.3	26.6	47.5	47.5
VideoCLIP [61]	0.33	0.59	0.25	0.24	0.62	0.55	36.6	34.3	39.7	38.5	48.3	44.8
InternVideo [59]	0.27	0.57	0.29	0.25	0.60	0.61	29.9	29.5	31.4	30.8	46.9	46.3
LookForTheChange [50]	0.35	0.68	0.36	0.25	0.77	0.68	30.3	28.7	32.5	30.0	37.2	36.1
MultiTaskChange [51]	0.49	0.80	0.41	0.22	0.72	0.62	33.9	29.9	38.5	34.1	43.1	38.8
VIDOSC (ours)	**0.57**	**0.84**	**0.56**	**0.48**	**0.89**	**0.82**	**46.4**	**43.1**	**46.6**	**43.7**	**60.7**	**58.2**

Table 2. Results on ChangeIt, ChangeIt (open-world), and HowToChange. VIDOSC outperforms all approaches in both closed-world and open-world scenarios, across known and novel OSCs.

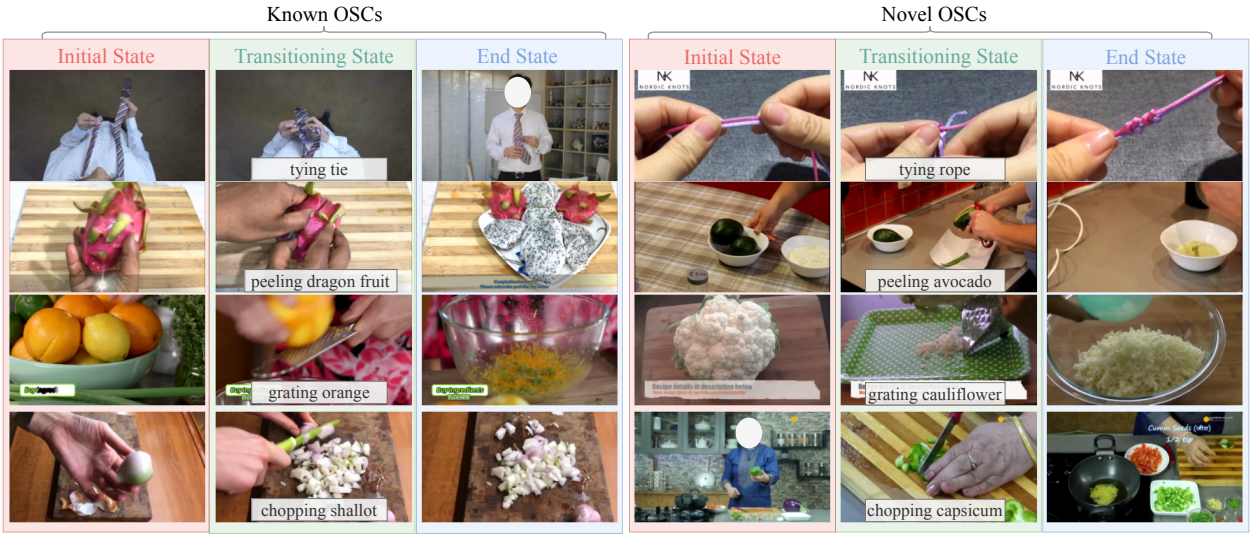

Figure 4. Top-1 frame predictions given by VIDOSC for the initial, transitioning, and end states, on ChangeIt (open-world) (first 2 rows) and HowToChange (last 2 rows). VIDOSC not only accurately localizes the three fine-grained states for known OSCs, but also generalizes this understanding to novel objects, such as cauliflower and capsicum, which are not observed during training.

evaluated on the test set with no training.

Implementation Videos are sampled at one frame per second. Each one-second video segment gets assigned to an OSC state label, and is encoded by InternVideo [59], a general video foundation model (which is kept frozen for training efficiency). Our video model consists of a 3-layer transformer with a hidden dimension of 512 and 4 attention heads as the encoder and a 1-layer MLP as the decoder. Consistent with prior work [50, 51], our model predicts the OSC state label (i.e., initial, transitioning, end state, or background) for a video, assuming the video OSC category is known (say from a recognition model's output or a user's specific query). While the standard output does not include the state transition name, our multi-task version detailed in Supp. is capable of this.

Main Results Table 2 presents results on all three datasets. VIDOSC outperforms all approaches in both traditional closed-world and our newly introduced open-world settings. The large performance gains—as much as 9.6% jumps in precision vs. the next best method—underscore the effectiveness of two pivotal components in VIDOSC. First, its use of text and VLMs for supervisory signals during training: particularly on novel OSCs , VIDOSC extracts meaningful cues from the video modality to refine pseudo labels from VLMs, and ultimately surpasses the VLM baselines. Second, our model for object-agnostic state prediction, designed for the open-world context, effectively narrows the gap between known and novel OSCs. For instance, on ChangeIt (open-world), MultiTaskChange [51] experiences a 19% decline in state precision@1 while VIDOSC has only a 8% drop. Note that we observe a more pronounced performance drop of all approaches on ChangeIt (open-world) than on HowToChange due to its limited number of objects available per state transition. These results also point to the potential for future development in bridging the known and novel OSC performance gap.

Qualitative Results Fig. 4 presents VIDOSC's top-1 frame predictions on ChangeIt (open-world) and HowToChange,

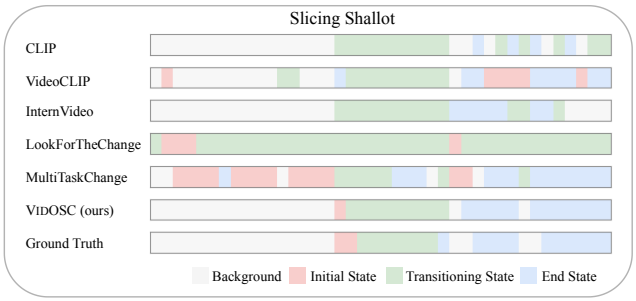

Figure 5. Comparison of model predictions across a test video depicting the OSC of "slicing shallot" on HowToChange. The x-axis represents temporal progression through the video. VIDOSC gives temporally smooth and coherent predictions that best align with the ground truth, significantly outperforming baselines in capturing the video's global temporal context.

Shared State	Temporal Modeling	Object Centric	Prec.@1 (%) known	novel	Δ
✗	✓	✓	58.5	53.3	5.2
✓	✗	✓	52.9	48.2	4.7
✓	✓	✗	59.8	56.7	3.1
✓	✓	✓	**60.7**	**58.2**	**2.5**

Table 3. Ablation Study. Δ denotes the performance gap between known OSCs and novel OSCs.

across both known and novel OSCs. Notably, despite never seeing objects such as cauliflower and capsicum during training, VIDOSC effectively leverages visual state change cues and correctly localizes the three states of these objects going through OSCs, demonstrating its strong generalization capability. For a more holistic view, Fig. 5 compares VIDOSC's frame-by-frame predictions with all baselines for a given test video. VIDOSC provides temporally coherent predictions, smoothly progressing through the OSC states in the natural order (i.e., from initial to transitioning then end). In contrast, baseline approaches often yield fragmented and inconsistent predictions, indicating a lack of understanding of the video's global temporal context, primarily due to their reliance on frame-wise modeling. See Supp. and Supp. video for more qualitative examples and VIDOSC's interpretability on object relations.

Ablation To further dissect the performance gains brought by our three model design techniques (i.e., shared state vocabulary, temporal modeling, and object-centric features), we conduct an ablation study, removing one component at a time. Table 3 confirms the essential role of each element. A shared state vocabulary is particularly crucial in the open-world context, as its absence increases the gap between known and novel OSCs from 2.5% to 5.2%. Furthermore, temporal modeling provides a substantial performance boost, and object-centric features offer further gains. See Supp. for an additional analysis of VIDOSC's performance with different pseudo labels.

Figure 6. Frame retrieval on the HowToChange test set. Given a query frame showcasing a *novel* OSC at its *initial* state, VIDOSC retrieves the nearest and furthest frame among all *known* OSCs at their *end* states. The closest post-OSC frame follows the transition trajectory of the query while the furthest post-OSC frame depicts a substantially different object, demonstrating VIDOSC's capability to generalize the OSC progression for novel objects.

Frame Retrieval VIDOSC learns features that accurately characterize the evolution of an OSC process. To illustrate, we consider a novel frame retrieval setting. Within the HowToChange test set, when presented with a query video featuring a novel OSC at its initial state, VIDOSC seeks the most similar and most contrasting video from a pool of candidates at their end states. The frame triplets with the smallest and largest feature distance are shown in Fig. 6. Remarkably, the retrieved nearest post-OSC frames correspond to the query's anticipated state transition trajectory, despite the object and state gap. Conversely, the furthest frames exhibit end states of markedly different objects. These results further lend support to VIDOSC's capability to understand the evolution of an OSC process, even for novel objects it has never encountered during training.

6. Conclusion

This work aims at a comprehensive exploration of video OSCs, with a novel open-world formulation. To address the challenges, we leverage text and VLMs to assist the training of a video OSC model at scale and design three modeling techniques to achieve object-agnostic state prediction for better generalization to novel OSCs. Furthermore, we present the most expansive video OSC dataset collection HowToChange, which echoes the natural long-tail of state transitions coupled with varied objects, fostering a realistic representation of real-world scenarios. As for future work, we will consider extending VIDOSC to video sequences featuring concurrent OSC processes, and integrating spatial understanding of OSC within our open-world framework.

Acknowledgements: UT Austin is supported in part by the IFML NSF AI Institute. KG is paid as a research scientist by Meta.

References

[1] Nachwa Aboubakr, James L Crowley, and Rémi Ronfard. Recognizing manipulation actions from state-transformations. *arXiv preprint arXiv:1906.05147*, 2019. 1, 2

[2] Jean-Baptiste Alayrac, Ivan Laptev, Josef Sivic, and Simon Lacoste-Julien. Joint discovery of object states and manipulation actions. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2127–2136, 2017. 1, 2, 3, 4, 6

[3] Kumar Ashutosh, Rohit Girdhar, Lorenzo Torresani, and Kristen Grauman. Hiervl: Learning hierarchical video-language embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23066–23078, 2023. 2

[4] Abhijit Bendale and Terrance Boult. Towards open world recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1893–1902, 2015. 3

[5] Jing Bi, Jiebo Luo, and Chenliang Xu. Procedure planning in instructional videos via contextual modeling and model-based policy learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15611–15620, 2021. 1

[6] Chien-Yi Chang, De-An Huang, Danfei Xu, Ehsan Adeli, Li Fei-Fei, and Juan Carlos Niebles. Procedure planning in instructional videos. In *European Conference on Computer Vision*, pages 334–350. Springer, 2020. 1

[7] Dibyadip Chatterjee, Fadime Sener, Shugao Ma, and Angela Yao. Opening the vocabulary of egocentric actions. *arXiv preprint arXiv:2308.11488*, 2023. 3

[8] Gioele Ciaparrone, Francisco Luque Sánchez, Siham Tabik, Luigi Troiano, Roberto Tagliaferri, and Francisco Herrera. Deep learning in video multi-object tracking: A survey. *Neurocomputing*, 381:61–88, 2020. 1

[9] Xinke Deng, Yu Xiang, Arsalan Mousavian, Clemens Eppner, Timothy Bretl, and Dieter Fox. Self-supervised 6d object pose estimation for robot manipulation. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3665–3671. IEEE, 2020. 1

[10] Akshay Dhamija, Manuel Gunther, Jonathan Ventura, and Terrance Boult. The overlooked elephant of object detection: Open set. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1021–1030, 2020. 3

[11] Nikita Dvornik, Isma Hadji, Ran Zhang, Konstantinos G Derpanis, Richard P Wildes, and Allan D Jepson. Stepformer: Self-supervised step discovery and localization in instructional videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18952–18961, 2023. 3

[12] Dave Epstein and Carl Vondrick. Learning goals from failure. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11194–11204, 2021. 1

[13] Dave Epstein, Boyuan Chen, and Carl Vondrick. Oops! predicting unintentional action in video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 919–929, 2020. 1

[14] Dave Epstein, Jiajun Wu, Cordelia Schmid, and Chen Sun. Learning temporal dynamics from cycles in narrated video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1480–1489, 2021. 3

[15] Alireza Fathi and James M Rehg. Modeling actions through state changes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2579–2586, 2013. 1, 2

[16] Tao Gong, Kai Chen, Xinjiang Wang, Qi Chu, Feng Zhu, Dahua Lin, Nenghai Yu, and Huamin Feng. Temporal roi align for video object recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1442–1450, 2021. 1

[17] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18995–19012, 2022. 1, 2, 3

[18] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. *arXiv preprint arXiv:2104.13921*, 2021. 2, 3

[19] Tengda Han, Weidi Xie, and Andrew Zisserman. Temporal alignment networks for long-term video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2906–2916, 2022. 1

[20] Farnoosh Heidarivincheh, Majid Mirmehdi, and Dima Damen. Detecting the moment of completion: Temporal models for localising action completion. *arXiv preprint arXiv:1710.02310*, 2017. 1

[21] Farnoosh Heidarivincheh, Majid Mirmehdi, and Dima Damen. Action completion: A temporal model for moment detection. *arXiv preprint arXiv:1805.06749*, 2018. 1

[22] Phillip Isola, Joseph J. Lim, and Edward H. Adelson. Discovering states and transformations in image collections. In *CVPR*, 2015. 1, 2

[23] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021. 2

[24] Licheng Jiao, Ruohan Zhang, Fang Liu, Shuyuan Yang, Biao Hou, Lingling Li, and Xu Tang. New generation deep learning for video object detection: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 33(8): 3195–3215, 2021. 1

[25] KJ Joseph, Salman Khan, Fahad Shahbaz Khan, and Vineeth N Balasubramanian. Towards open world object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5830–5840, 2021. 3

[26] Chen Ju, Tengda Han, Kunhao Zheng, Ya Zhang, and Weidi Xie. Prompting visual-language models for efficient video

understanding. In *European Conference on Computer Vision*, pages 105–124. Springer, 2022. 3

[27] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pages 12888–12900. PMLR, 2022. 2

[28] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. 2

[29] Xiangyu Li, Xu Yang, Kun Wei, Cheng Deng, and Muli Yang. Siamese contrastive embedding network for compositional zero-shot learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9326–9335, 2022. 2, 3

[30] Kevin Qinghong Lin, Jinpeng Wang, Mattia Soldan, Michael Wray, Rui Yan, Eric Z XU, Difei Gao, Rong-Cheng Tu, Wenzhe Zhao, Weijie Kong, et al. Egocentric video-language pretraining. *Advances in Neural Information Processing Systems*, 35:7575–7586, 2022. 2

[31] Yang Liu, Ping Wei, and Song-Chun Zhu. Jointly recognizing object fluents and tasks in egocentric videos. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2924–2932, 2017. 2, 6, 1

[32] Timo Lüddecke and Alexander Ecker. Image segmentation using text and image prompts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7086–7096, 2022. 2

[33] Massimiliano Mancini, Muhammad Ferjad Naeem, Yongqin Xian, and Zeynep Akata. Open world compositional zero-shot learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5222–5230, 2021. 2, 3

[34] Joanna Materzynska, Tete Xiao, Roei Herzig, Huijuan Xu, Xiaolong Wang, and Trevor Darrell. Something-else: Compositional action recognition with spatial-temporal interaction networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1049–1059, 2020. 2, 3

[35] Effrosyni Mavroudi, Triantafyllos Afouras, and Lorenzo Torresani. Learning to ground instructional articles in videos through narrations. *arXiv preprint arXiv:2306.03802*, 2023. 3

[36] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2630–2640, 2019. 2, 3, 5, 1, 6

[37] Ishan Misra, Abhinav Gupta, and Martial Hebert. From red wine to red tomato: Composition with context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1792–1801, 2017. 2, 3

[38] Muhammad Ferjad Naeem, Yongqin Xian, Federico Tombari, and Zeynep Akata. Learning graph embeddings for compositional zero-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 953–962, 2021.

[39] Tushar Nagarajan and Kristen Grauman. Attributes as operators: factorizing unseen attribute-object compositions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 169–185, 2018. 2, 3

[40] OpenAI. Gpt-4 technical report. Technical report, OpenAI, 2023. Submitted on 15 Mar 2023, last revised 27 Mar 2023. 2, 6, 9

[41] Khoi Pham, Kushal Kafle, Zhe Lin, Zhihong Ding, Scott Cohen, Quan Tran, and Abhinav Shrivastava. Learning to predict visual attributes in the wild. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13018–13028, 2021. 2, 3

[42] Chiara Plizzari, Gabriele Goletto, Antonino Furnari, Siddhant Bansal, Francesco Ragusa, Giovanni Maria Farinella, Dima Damen, and Tatiana Tommasi. An outlook into the future of egocentric vision. *arXiv preprint arXiv:2308.07123*, 2023. 1

[43] Senthil Purushwalkam, Maximilian Nickel, Abhinav Gupta, and Marc'Aurelio Ranzato. Task-driven modular networks for zero-shot compositional learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3593–3602, 2019. 2, 3

[44] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2, 5, 6, 7, 9

[45] Santhosh Kumar Ramakrishnan, Ziad Al-Halah, and Kristen Grauman. Naq: Leveraging narrations as queries to supervise episodic memory. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6694–6703, 2023. 3

[46] Nirat Saini, Bo He, Gaurav Shrivastava, Sai Saketh Rambhatla, and Abhinav Shrivastava. Recognizing actions using object states. In *ICLR2022 Workshop on the Elements of Reasoning: Objects, Structure and Causality*, 2022. 1, 2

[47] Nirat Saini, Hanyu Wang, Archana Swaminathan, Vinoj Jayasundara, Bo He, Kamal Gupta, and Abhinav Shrivastava. Chop & learn: Recognizing and generating object-state compositions. *ICCV*, 2023. 1, 2, 3

[48] Walter J Scheirer, Anderson de Rezende Rocha, Archana Sapkota, and Terrance E Boult. Toward open set recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(7):1757–1772, 2012. 3

[49] Dandan Shan, Jiaqi Geng, Michelle Shu, and David F Fouhey. Understanding human hands in contact at internet scale. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9869–9878, 2020. 4

[50] Tomáš Souček, Jean-Baptiste Alayrac, Antoine Miech, Ivan Laptev, and Josef Sivic. Look for the change: Learning object states and state-modifying actions from untrimmed web videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13956–13966, 2022. 1, 2, 3, 4, 6, 7, 5

[51] Tomáš Souček, Jean-Baptiste Alayrac, Antoine Miech, Ivan Laptev, and Josef Sivic. Multi-task learning of object state changes from uncurated videos. *arXiv preprint arXiv:2211.13500*, 2022. 1, 2, 3, 4, 6, 7, 5, 8

[52] Pavel Tokmakov, Jie Li, and Adrien Gaidon. Breaking the" object" in video object segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22836–22845, 2023. 1, 2

[53] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 2

[54] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023. 2, 5, 1

[55] Jonathan Tremblay, Thang To, Balakumar Sundaralingam, Yu Xiang, Dieter Fox, and Stan Birchfield. Deep object pose estimation for semantic robotic grasping of household objects. *arXiv preprint arXiv:1809.10790*, 2018. 1

[56] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 4

[57] Mengmeng Wang, Jiazheng Xing, and Yong Liu. Actionclip: A new paradigm for video action recognition. *arXiv preprint arXiv:2109.08472*, 2021. 3

[58] Xiaolong Wang, Ali Farhadi, and Abhinav Gupta. Actions˜ transformations. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2658–2667, 2016. 1, 2

[59] Yi Wang, Kunchang Li, Yizhuo Li, Yinan He, Bingkun Huang, Zhiyu Zhao, Hongjie Zhang, Jilan Xu, Yi Liu, Zun Wang, et al. Internvideo: General video foundation models via generative and discriminative learning. *arXiv preprint arXiv:2212.03191*, 2022. 6, 7, 9

[60] Te-Lin Wu, Yu Zhou, and Nanyun Peng. Localizing active objects from egocentric vision with symbolic world knowledge. *arXiv preprint arXiv:2310.15066*, 2023. 1

[61] Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. Videoclip: Contrastive pre-training for zero-shot video-text understanding. *arXiv preprint arXiv:2109.14084*, 2021. 5, 6, 7, 9

[62] Rui Yao, Guosheng Lin, Shixiong Xia, Jiaqi Zhao, and Yong Zhou. Video object segmentation and tracking: A survey. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11(4):1–47, 2020. 1

[63] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022. 2

[64] Jiangwei Yu, Xiang Li, Xinran Zhao, Hongming Zhang, and Yu-Xiong Wang. Video state-changing object segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20439–20448, 2023. 1, 2, 3

[65] Alireza Zareian, Kevin Dela Rosa, Derek Hao Hu, and Shih-Fu Chang. Open-vocabulary object detection using captions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14393–14402, 2021. 3

[66] He Zhao, Isma Hadji, Nikita Dvornik, Konstantinos G Derpanis, Richard P Wildes, and Allan D Jepson. P3iv: Probabilistic procedure planning from instructional videos with weak supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2938–2948, 2022. 1

[67] Yue Zhao, Ishan Misra, Philipp Krähenbühl, and Rohit Girdhar. Learning video representations from large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6586–6597, 2023. 2

1. Video Containing Qualitative Results

We invite the reader to view the video available at https://vision.cs.utexas.edu/projects/VidOSC/, where we provide: (1) a comprehensive overview of VIDOSC, (2) video examples from HowToChange, and (3) qualitative examples of VIDOSC's predictions. These examples highlight VIDOSC's ability in identifying non-OSC moments as background and effectively distinguishing among the three OSC states. It delivers temporally smooth and coherent predictions that follow the natural OSC progression (from initial to transitioning, and then to the end state), and shows strong performance even with novel OSCs not seen in training. All these underscore the efficacy of VIDOSC.

2. Video OSC

Expanding on Sec. 3.1, we clarify three aspects of our definition of video OSC. First, we focus on OSCs that lead to a visible change in an object's appearance. Processes that are non-visual or involve mere spatial movements (such as moving an apple from the sink to the cutting board) do not qualify as OSCs. Second, in line with previous works [2, 31, 50, 51], we operate under the assumption that each input video predominantly features a single OSC. The challenge of handling videos with multiple concurrent OSCs remains an intriguing avenue for future research. Lastly, during training, the input video and its OSC category name (e.g., shredding chicken) are available (as provided in both ChangeIt and our HowToChange, although not always accurate due to data collection noise). For evaluation, every test video (in both ChangeIt and HowToChange) is accompanied by a manually verified OSC category.

2.1. Data Collection

We streamline our dataset collection via an automated process. First we apply LLAMA2 [54] to ASR transcriptions in HowTo100M [36] with the following text prompt, one sentence at a time:

> *[Text Prompt to LLAMA2]* You will receive descriptions corresponding to a how-to instructional video. Your task is to identify any instances of Object State Change (OSC) based on the provided text. An OSC is a visually detectable transformation where an object undergoes a change that is difficult to reverse. Examples include apple peeling/cutting, bacon frying, milk boiling, butter melting, cake frosting, eggs whisking, cream whipping, etc.
>
> - Note 1: OSCs must be visually detectable. General actions like food preparing, or non-visual processes like onions sweetening, are not included.
> - Note 2: Simple spatial transitions, resulting from actions like add, mix, put, or place, are not considered OSCs.
> - Note 3: To qualify as an OSC, there must be a transition from one state to another, which should be indicated by an active action in the text description. The mere presence of a state (e.g., sliced pineapples, peeled apples) does not count unless there is explicit text describing the change (e.g., we slice the pineapples).
> - Note 4: Generally, sentences without OSCs are far more common than those with OSCs.
>
> To report identified OSCs, please use the following format: [object] + [state transition of the OSC]. Ensure the first word in each identified OSC is the object, and the subsequent words describe a state transition. If multiple OSCs are identified, separate them with semicolons (;). If no OSCs are detected, simply reply with None.

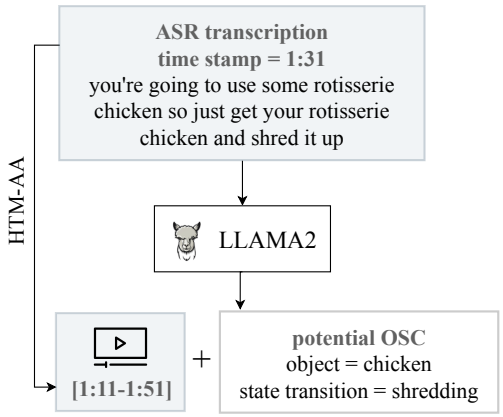

Figure 7. Our proposed data collection process for HowToChange. HTM-AA [19] denotes the auto-aligned version of HowTo100M.

From the responses given by LLAMA2, we identify object and state transitions corresponding to the ASR transcription. Utilizing the HTM-AA dataset [19], where each ASR sentence corresponds to a time stamp in the video, we extract a clip centered around the identified time stamp with a $\pm$ 20 second window. The result is a cropped video segment of 40 seconds, paired with an OSC text (in the form of object + state transition). Finally, when multiple clips from the same video illustrate the same OSC with overlapping start and end times, we combine them into a single, extended clip. The whole process is illustrated in Fig. 7.

3. The HowToChangeDataset

To establish the OSC taxonomy, we identify 20 most frequent state transitions and the objects associated with these state transitions that appear more than 200 times. Utilizing a 0.25 quantile threshold for each state transition, we categorize the top 75% frequent OSCs as known and the bottom 25% as novel, resulting in 318 known and 91 novel OSC categories in total. See Table 4 for the complete OSC taxonomy. With these 318 known OSCs, we compose the training set of HowToChange, encompassing 36,076 video segments from HowTo100M. Fig. 9 provides the detailed distribution of HowToChange (Training).

3.1. Ground Truth Label Collection

For evaluation, we collect annotations from 30 trained professional human annotators for a subset of 5,423 video clips from HowToChange, amounting to 62.5 hours of video. See Fig. 8 for the annotation user interface. We collect an average of 13.3 annotated videos per OSC category. The annotations for known and novel

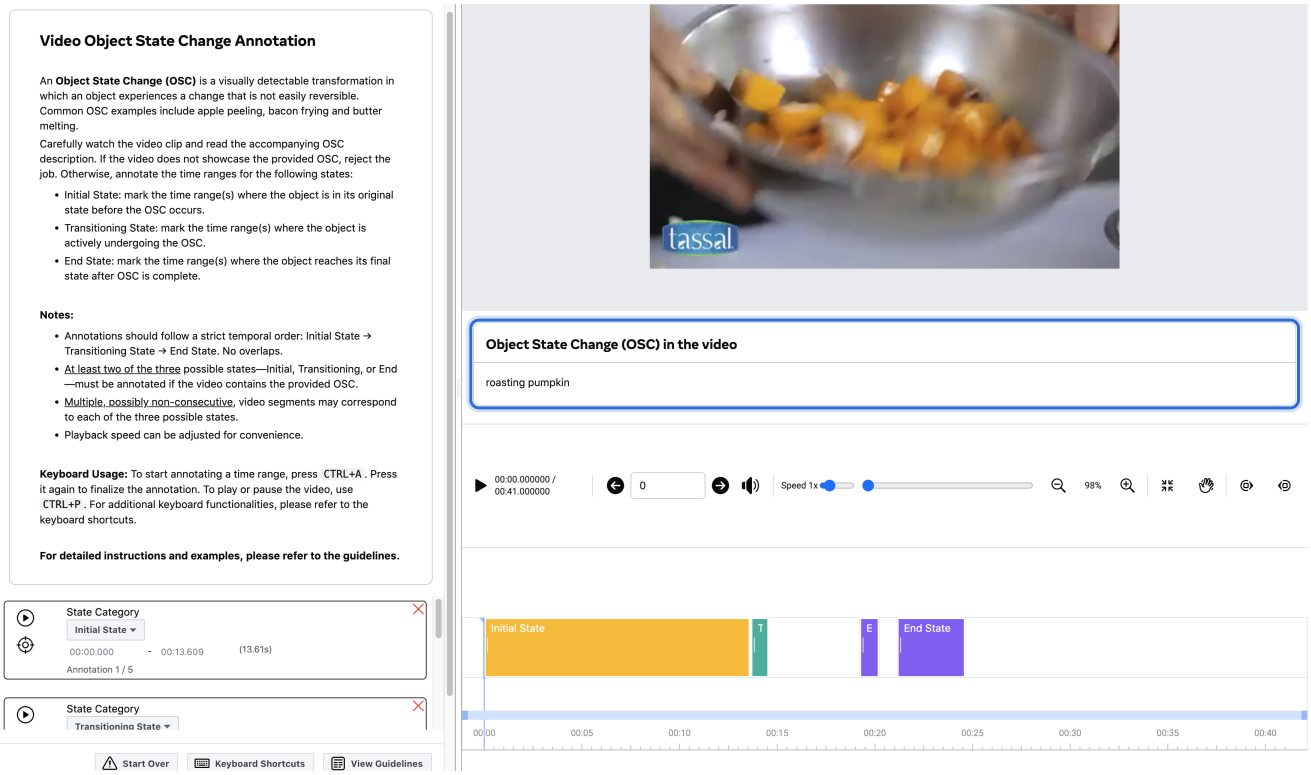

Figure 8. The annotation user interface. Annotators view a video paired with an OSC category, identified from the OSC mining process (outlined in Sec. 3.3). They are instructed to either reject the video if it does not demonstrate the specified OSC, or to annotate the time ranges corresponding to the initial, transitioning, and end states of the OSC shown in the video.

OSCs cap at 15 and 10, respectively. Fig. 10 provides the detailed distribution of HowToChange (Evaluation).

The breakdown of annotated time ranges within these videos is as follows: 19.8% for the initial state, 25.9% for the transitioning state, and 16.9% for the end state, with the remaining categorized as non-OSC-related background. Fig. 11 shows the distribution of video duration (seconds) and the number of annotated time ranges per state in the HowToChange (Evaluation) set. Importantly, the distribution demonstrates the granularity of our annotations, with a varied number of annotated time ranges per video. This is a result of our guidelines that instruct the annotators to exclude any time ranges where the OSC of interest is not observable, thereby ensuring precise labeling.

4. Experiments

4.1. Experimental Setup

Datasets We conduct experiments on ChangeIt [50] and our proposed HowToChange. Other related datasets are not included due to their small scale and differing OSC definition [2], unavailability for public access [31], or the absence of OSC category / temporal labels necessary for our problem [17, 47]. Beyond the conventional split of ChangeIt, we introduce a novel split designed specifically for our open-world framework. The OSC taxonomy is detailed in Table 5. Note the inherent challenge of this setting: each state transition is associated with fewer than five objects. This scarcity of objects per state transition can significantly impede the model's ability to generalize the concept of states and state transitions without becoming overly dependent on specific objects, and reinforces the motivation for creating our new HowToChange dataset to augment this existing resource.

Evaluation For ChangeIt and ChangeIt (open-world), we adhere to the original dataset's evaluation protocol, reporting action and state precision@1 as the evaluation metrics. For our collected HowToChange, while we also adopt state precision@1, due to our definition that positions the midpoint between the initial and end states as "transitioning states" rather than "action", our evaluation takes into account three distinct states, unlike the two states in ChangeIt. In addition, since precision@1 solely evaluates a single frame for each state within a video, we advocate for the use of F1 score and precision over all frames to ensure a more holistic evaluation. For each video, we compute the state precision@1, F1 score and precision for the present states (considering that a video might not always contain all three states: initial, transitioning, and end) and then compute their average over states. Subsequently, we average these values across videos within a state transition category and report the overall average for all state transitions. Lastly, for the two open-world datasets, we present these metrics on both known and novel OSCs.

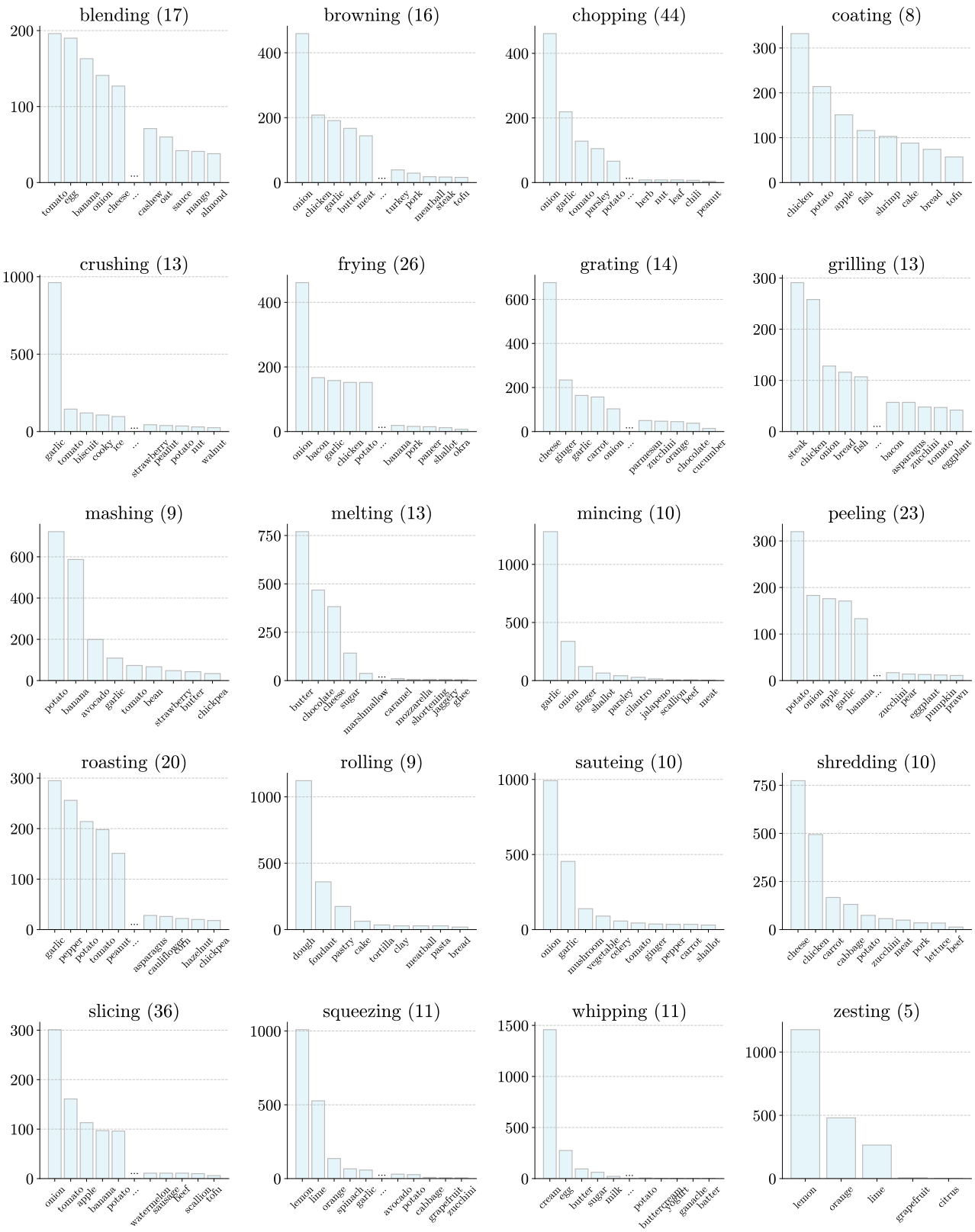

Figure 9. Data distribution of HowToChange (Training). The y-axis denotes the number of annotated videos, and numbers in parentheses represent the count of unique objects associated with each state transition. Our data collection process mines video OSCs that authentically reflects the real-world's long-tail.

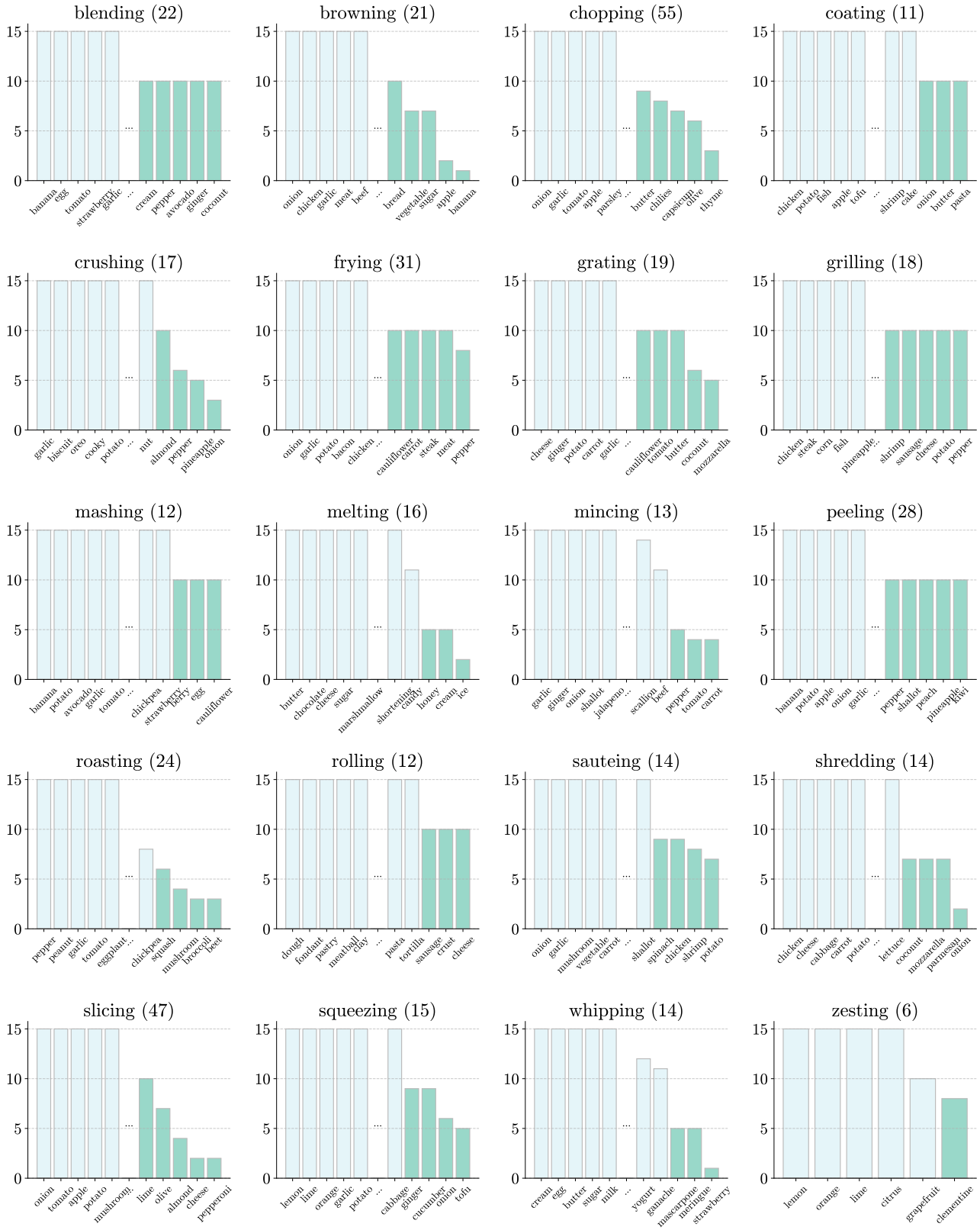

Figure 10. Data distribution of HowToChange (Evaluation). Known and novel OSCs are shown in light blue and dark green, respectively. HowToChange (Evaluation) presents a comprehensive evaluation benchmark, encompassing a diverse array of objects and state transitions.

State Transition	Objects (known)	Objects (novel)
blending	banana, egg, tomato, strawberry, garlic, butter, oat, sugar, milk, ice, onion, date, cashew, sauce, almond, cheese, mango	cream, pepper, avocado, ginger, coconut
browning	onion, chicken, garlic, meat, beef, sausage, butter, crust, bacon, pork, meatball, mushroom, tofu, turkey, steak, potato	bread, vegetable, sugar, apple, banana
chopping	onion, garlic, tomato, apple, parsley, carrot, pepper, mushroom, bacon, cilantro, spinach, cabbage, nut, banana, strawberry, cucumber, chocolate, rosemary, chive, shallot, peanut, vegetable, herb, kale, celery, mint, dill, mango, chicken, walnut, leaf, potato, jalapeno, zucchini, chili, egg, pecan, ginger, coriander, basil, avocado, broccoli, scallion, lettuce	cauliflower, almond, sausage, pineapple, date, leek, butter, chilies, capsicum, olive, thyme
coating	chicken, potato, fish, apple, tofu, bread, shrimp, cake	onion, butter, pasta
crushing	garlic, biscuit, oreo, cooky, potato, ice, walnut, ginger, strawberry, cracker, tomato, peanut, nut	almond, pepper, pineapple, onion
frying	onion, garlic, potato, bacon, chicken, tortilla, egg, fish, plantain, tofu, bread, mushroom, tomato, rice, sausage, batter, paneer, eggplant, shallot, beef, vegetable, shrimp, ginger, okra, pork, banana	cauliflower, carrot, steak, meat, pepper
grating	cheese, ginger, potato, carrot, garlic, nutmeg, orange, zucchini, cucumber, parmesan, chocolate, apple, lemon, onion	cauliflower, tomato, butter, mozzarella, coconut
grilling	chicken, steak, corn, fish, pineapple, salmon, onion, bread, tomato, zucchini, asparagus, bacon, eggplant	shrimp, sausage, cheese, potato, pepper
mashing	banana, potato, avocado, garlic, tomato, bean, butter, chickpea, strawberry	berry, egg, cauliflower
melting	butter, chocolate, cheese, sugar, marshmallow, ghee, caramel, jaggery, gelatin, margarine, mozzarella, shortening, candy	honey, cream, ice
mincing	garlic, ginger, onion, shallot, jalapeno, cilantro, parsley, beef, meat, scallion	pepper, tomato, carrot
peeling	banana, potato, apple, onion, garlic, plantain, egg, orange, ginger, carrot, cucumber, lemon, tomato, squash, avocado, mango, eggplant, pumpkin, shrimp, pear, beet, zucchini, prawn	pepper, shallot, peach, pineapple, kiwi
roasting	pepper, peanut, garlic, tomato, eggplant, potato, coconut, onion, nut, pumpkin, almond, chicken, vegetable, cauliflower, carrot, hazelnut, turkey, chickpea, corn, asparagus	broccoli, squash, beet, mushroom
rolling	dough, fondant, pastry, meatball, clay, cake, bread, pasta, tortilla	sausage, crust, cheese
sauteing	onion, garlic, mushroom, vegetable, carrot, ginger, celery, pepper, tomato, shallot	spinach, shrimp, chicken, potato
shredding	chicken, cheese, cabbage, carrot, potato, zucchini, beef, pork, meat, lettuce	coconut, mozzarella, parmesan, onion
slicing	onion, tomato, apple, potato, mushroom, garlic, lemon, banana, strawberry, cabbage, meat, zucchini, chicken, mango, pepper, cake, shallot, egg, sausage, watermelon, carrot, tofu, ginger, leek, beef, cucumber, scallion, eggplant, avocado, bread, pear, steak, pineapple, radish, peach, bacon	jalapeno, celery, butter, olive, mozzarella, orange, ham, lime, almond, cheese, pepperoni
squeezing	lemon, lime, orange, garlic, potato, spinach, avocado, zucchini, tomato, grapefruit, cabbage	ginger, cucumber, onion, tofu
whipping	cream, egg, butter, sugar, milk, potato, ganache, buttercream, batter, yogurt, frosting	mascarpone, strawberry, meringue
zesting	lemon, orange, lime, citrus, grapefruit	clementine

Table 4. The OSC taxonomy for HowToChange encompasses 134 objects undergoing 20 distinct state transitions, resulting in 409 unique OSCs (318 known and 91 novel).

State Transition	Objects (known)	Objects (novel)
peeling	apple, dragon fruit, onion, pineapple	avocado, corn, eggs, garlic
frying	bacon	potatoes
pouring	beer, tea	juice, milk
wrapping	tortilla	gift/box
melting	butter	chocolate
cleaning	pan	shoes
tying	tie, ribbon	rope
cutting	tile	tree

Table 5. The OSC taxonomy for ChangeIt (open-world). Aligning with our open-world formulation, we propose a new split of ChangeIt [50] that randomly splits objects associated with the same state transition as known and novel.

Baselines For results on ChangeIt, we reference the metrics as officially reported in their original papers. For results on ChangeIt (open-world), we reimplement the baselines to accommodate our newly introduced data split. As the LookForTheChange baseline [50] requires a model for every OSC category, when evaluating novel OSCs, we apply every model trained on OSCs with the same state transition and report best model performance. For the MultiTaskChange baseline [51], we train one multi-task model across all known OSCs and evaluate on both known and novel OSCs. On HowToChange, baseline results [50, 51] were obtained using InternVideo features, not their originally proposed features, to ensure comparability. Aligning with our own approach, we adopt a shared vocabulary for both baselines. This means grouping OSCs with the same state transition as one category to enhance generalization. On all datasets, following their original papers, we enforce an additional casual ordering constraint during test time as we observe better performance of the baselines in this setting. We adopt adaptive weights for baaselines on open-world ChangeIt using the values from their original papers but not on HowToChange

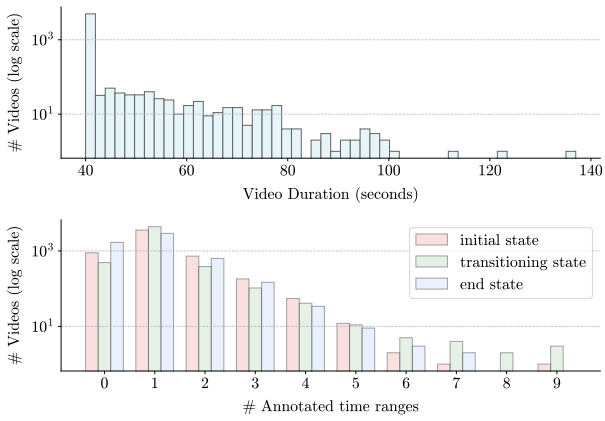

Figure 11. Distribution of video duration (upper) and number of annotated time ranges (lower).

due to no exemplar images.

Regarding the three zero-shot baselines (i.e., CLIP [44], VideoClip [61] and InternVideo [59]), we adopt both the vision and language encoders to compute the similarity score between each (image/video, text) pair. The OSC state description text is same as adopted in our pseudo label generation process (Sec. 3.3). Based on the similarity scores, we then conduct a grid search within each state transition to pinpoint the optimal threshold distinguishing background classes from OSC state categories, and report the best value.

Implementation For training pseudo label generation, we first employ GPT-4 [40] to automatically generate text descriptions for each OSC category in the dataset. We then assign pseudo labels to each video segment based on the similarity scores given by a CLIP [44] model for ChangeIt and a VideoCLIP [61] trained on HowTo100M [36] for HowToChange. We conduct a grid search for the two pseudo label thresholds δ and τ to identify the best value. We employ the AdamW optimizer with a learning rate of 1e-4 and a weight decay of 1e-4. Models are trained using a batch size of 64, over 50 epochs. Training takes a few hours on a NVIDIA A100.

To ensure a thorough evaluation, we train both single-task and multi-task variants of our approach. We note that the approach is designed differently for ChangeIt and HowToChange, due to their distinct characteristics: (1) For ChangeIt and ChangeIt (open-world), the term single-task denotes training a separate model for each OSC category (e.g. peeling apples), whereas multi-task denotes training a unified model for all OSC categories. Regarding baseline methods, LookforTheChange [50] aligns with the single-task paradigm while MultiTaskChange [51] belongs to the multi-task one. (2) For HowToChange, where each state transition is associated with a much broader range of objects, we adopt a shared state vocabulary to enhance model generalization, both for our approach and the baselines. In this context, the single-task model is developed for each state transition (e.g, peeling) rather than each OSC (e.g. peeling apples). Consequently, a single-task model is already capable of identifying states for any OSCs that fall within the same state transition category. The multi-task model extends this concept to accommodate all 20 state transitions in HowToChange. Both baselines, LookforTheChange [50] and MultiTaskChange [51] fall into the single-task implementation as we observe worse performance and prohibitive long training time with the multi-task formulation.

State Transition	F1 (%)		Prec (%)		Prec.@1 (%)	
	known	novel	known	novel	known	novel
chopping	46.5	44.1	43.7	42.4	58.3	58.2
slicing	48.6	45.6	49.7	44.9	68.6	63.7
frying	56.3	53.7	53.5	50.8	61.2	54.5
peeling	49.0	42.4	51.4	45.8	65.6	57.7
blending	42.2	45.2	43.4	50.7	59.1	66.7
roasting	36.5	40.8	40.3	44.4	59.2	64.6
browning	44.9	51.5	46.3	54.4	55.1	60.5
grating	52.5	51.3	51.6	50.4	66.6	65.0
grilling	54.6	53.7	54.0	49.6	67.8	61.0
crushing	39.2	32.2	38.3	28.0	58.9	52.8
melting	34.9	36.8	35.1	38.2	46.6	38.9
squeezing	54.4	54.6	54.0	54.6	61.0	66.1
sauteing	47.7	36.4	47.1	41.4	56.3	46.0
shredding	53.3	41.8	52.8	44.8	66.6	58.7
whipping	45.1	43.1	46.9	44.3	57.8	45.5
rolling	39.7	32.6	43.5	39.3	62.2	60.0
mashing	52.4	52.0	52.2	53.8	66.7	69.4
mincing	45.3	37.2	41.7	32.1	54.7	55.1
coating	35.5	30.0	37.8	28.5	55.1	48.3
zesting	49.6	36.2	49.3	35.3	65.9	70.8
Average	46.4	43.1	46.6	43.7	60.7	58.2

Table 6. Detailed per-state-transition results of VIDOSC on HowToChange.

Our multi-task model follows the same design as the single-task, with the modification of an expanded output label dimension to encapsulate all categories. Essentially, the multi-task variant can be conceptualized as a hierarchical classification problem. During testing, the model first determines the most probable state transition (e.g., peeling) based on prediction scores. Subsequently, it provides a prediction of fine-grained states for each time point (e.g., initial, transitioning and end state of peeling, or background). It's important to note that while the single-task variant only performs the latter prediction step, the multi-task variant adds the ability to name the state transition. The multi-task model thus offers the benefit of a single, unified model that can predict OSC states for all categories of videos, eliminating the need for developing individual specialized models.

Finally, we emphasize that our model, irrespective of the variant, *relies solely on video as input*. This is in contrast to VLM baselines (i.e., CLIP [44], VideoCLIP [61] and InternVideo [59]), where the OSC text is required as input to calculate the cross-modality similarity.

4.2. Results

Detailed per-state-transition results Supplementing Table 2 in the main paper, we provide a detailed breakdown of VIDOSC's performance on HowToChange per state transition in Table 6. We observe superior results in transitions like mashing, squeezing, and grilling, while transitions such as melting and coat-

Method	ChangeIt		ChangeIt (open-world)				HowToChange					
	State Prec.@1	Action Prec.@1	State Prec.@1		Action Prec.@1		F1 (%)		Prec (%)		Prec.@1 (%)	
			known	novel	known	novel	known	novel	known	novel	known	novel
VIDOSC (multi-task)	0.44	0.69	0.43	0.29	0.75	0.63	40.7	37.9	41.8	39.0	56.8	54.8
VIDOSC (single-task)	0.57	0.84	0.56	0.48	0.89	0.82	46.4	43.1	46.6	43.7	60.7	58.2

Table 7. A comparison of the single-task and multi-task variant of VIDOSC. The multi-task variant offers the benefit of a single, unified model capable of predicting fine-grained OSC states for videos of all OSC categories, while the single-task variant is optimized for each individual OSC and demonstrates superior performance.

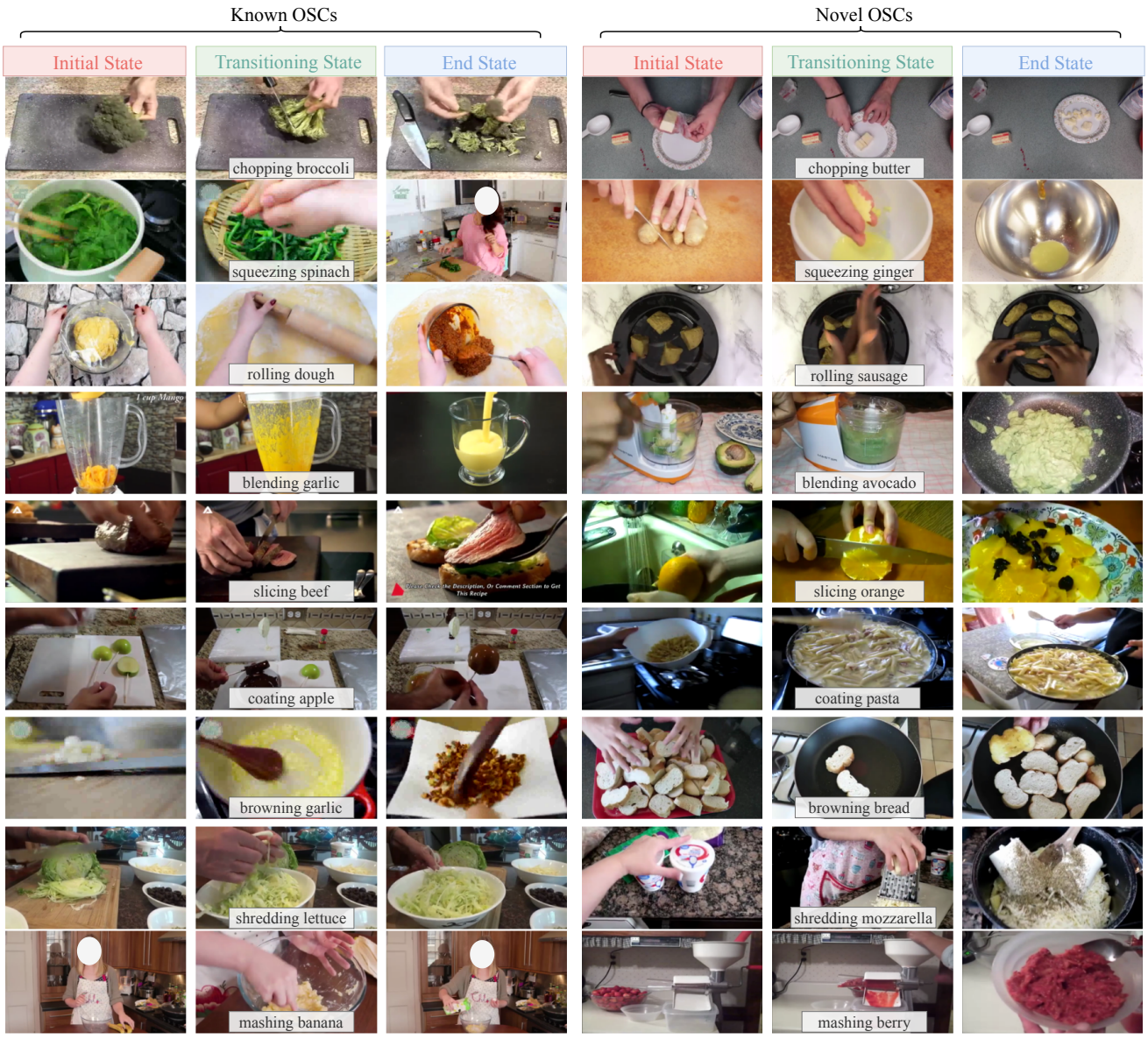

Figure 12. Top-1 frame predictions given by VIDOSC for the initial, transitioning, and end states, on HowToChange(Evaluation). VIDOSC not only accurately localizes the three fine-grained states for known OSCs, but also generalizes this understanding to novel objects, which are not observed during training.

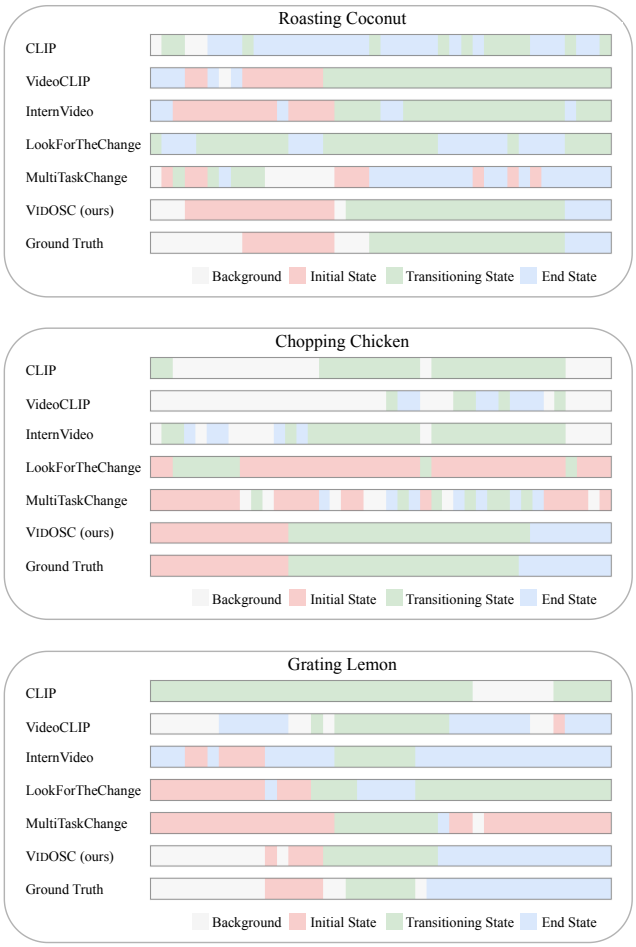

Figure 13. Comparison of model predictions on HowToChange (Evaluation). The x-axis represents temporal progression through the video. VIDOSC gives temporally smooth and coherent predictions that best align with the ground truth, significantly outperforming baselines in capturing the video's global temporal context.

ing show comparatively weaker performance, possibly due to the ambiguity in their OSC states. In addition, the known-novel OSC gap is smallest for chopping, grating and mashing, whereas shredding and sauteing exhibit larger performance discrepancies. Intuitively, state transitions like chopping and mashing share more invariant representations across objects, with objects consistently going from whole to pieces or a mashed state. In contrast, shredding and sauteing may demonstrate less consistent transformation patterns across different objects, leading to greater variability and thus larger performance discrepancies. We hope these results provide insight for further analysis and development in this area.

Single-task vs Multi-task We train both single-task and multi-task variants of VIDOSC, and compare their performance in Table 7. While the multi-task model offers the convenience of a unified framework that can handle various state transitions simultaneously, they generally underperform their single-task counterparts. This performance disparity is a long-standing problem in multi-task learning and could stem from multiple factors such as varied convergence rates and potential competition among different OSCs, suggesting promising areas for future research. In terms of the comparison of VIDOSC (multi-task) with the MultiTaskChange baseline [51], for ChangeIt (open-world), VIDOSC achieves a +0.02 increase in state precision@1 and +0.03 in action precision@1 for known OSCs, and a +0.07 and +0.01 increase for novel OSCs, respectively. For the standard ChangeIt dataset, our reimplementation of MultiTaskChange based on their officially released code achieves a state precision@1 of 0.40 and an action precision@1 of 0.69, lower than the original paper's reported 0.49 and 0.80. VIDOSC (multi-task) surpasses the reproduced numbers with a 0.04 improvement in state precision@1. Lastly, we note that HowToChange features closely related state transitions (such as crushing and mashing, melting and browning) as well as fine-grained variations within a general transition, (such as various cutting ways: chopping, slicing and mincing). These variations present both challenges and opportunities for the advancement of multi-task models, particularly in modeling the similarities and fine distinctions among different state transition categories. We leave it as future work.

Further Qualitative Results We provide more qualitative results supplementing Fig. 4 and Fig. 5 in the main paper. Fig. 12 showcases more examples of VIDOSC's top-1 predictions on HowToChange (Evaluation). VIDOSC gives correct predictions for various state transitions, across both known and novel objects. In addition, Fig. 13 provides more examples of VIDOSC's predictions from a global perspective. Compared with all approaches, VIDOSC consistently delivers temporally coherent predictions that closely align with the ground truth labels. All these results help demonstrate the strong performance of VIDOSC.

Interpretability on Object Relations VIDOSC provides interpretable insights on how object relate to each other during specific state transitions. To illustrate this, we calculate features that belong to the transitioning state of "crushing", averaged over objects across all test videos. We then compute a feature distance matrix from these object features, as depicted in Fig. 14. While VIDOSC is purely video-based and has no access to ground truth object names, the feature embeddings it produces well captures the relations between each object pair during the "crushing" transition. For instance, crushing cracker is more similar to crushing oreo or biscuit than to crushing pineapple, which aligns with our intuition. Furthermore, the heatmap reveals how VIDOSC effectively leverages known object relationships to reason about novel objects. For example, the novel object "almond" is closet in feature space to "walnut" and "peanut" among all known objects.

Pseudo Label Analysis The pseudo label thresholds δ and τ in Section 3.3 are decided via a hyperparameter search. Figure 15 illustrates the process, showing F1 scores for different δ and τ for the state transition of slicing and sauteing. According to this analysis, we set pseudo label thresholds $\tau = 12$ and $\delta = 0$ for the state transition of slicing and $\tau = 6$ and $\delta = 0.05$ for sauteing. We repeat this process for all other state transitions and for zero-shot baselines as well for a fair comparison.

In addition, we experiment with no ordering constraint enforced in pseudo label generation (Section 3.3) on HowToChange.

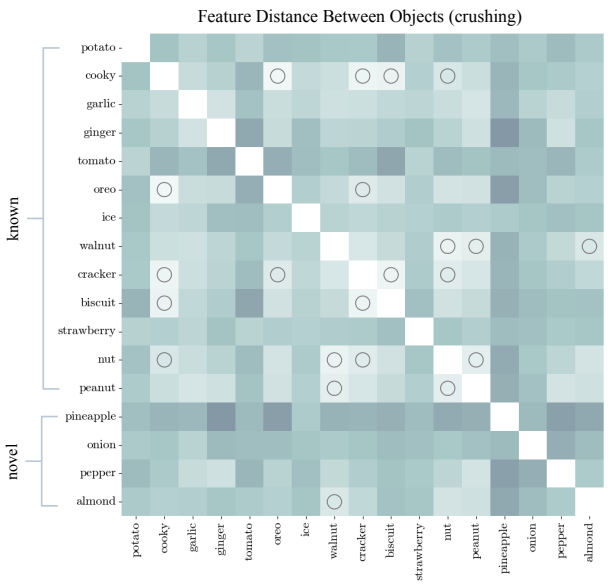

Figure 14. Distance matrix between object features produced by VIDOSC during the "crushing" process. A lighter color indicates smaller feature distance, and object pairs with relative small distance are marked by a circle. The heatmap offers interpretability on object relations during a state transition and provides insight on how the model generalizes from known to novel objects.

Method	F1 (%)		Prec (%)		Prec.@1 (%)	
	known	novel	known	novel	known	novel
CLIP [44]	26.9	25.4	27.3	26.6	47.5	47.5
VIDOSC (CLIP)	35.5	34.1	38.6	36.3	51.1	48.5
Improvement	+8.6	+8.7	+11.3	+9.7	+3.6	+1.0
VideoCLIP [61]	36.6	34.3	39.7	38.5	48.3	44.8
VIDOSC (VideoCLIP)	46.4	43.1	46.6	43.7	60.7	58.2
Improvement	+9.8	+8.8	+6.9	+5.2	+12.4	+13.4

Table 8. A comparison of VIDOSC using pseudo labels provided by CLIP [44] and VideoCLIP [61]. VIDOSC is a flexible framework can be combined with different VLMs. Employing a better VLM (VideoCLIP over CLIP) further enhances VIDOSC's performance.

Method	F1 (%)		Prec (%)		Prec.@1 (%)	
	known	novel	known	novel	known	novel
VIDOSC (no ordering)	41.1	36.6	41.7	37.4	52.1	47.5
VIDOSC	46.4	43.1	46.6	43.7	60.7	58.2

Table 9. A comparison of VIDOSC with and without ordering constraint enforced in pseudo label generation. Enforcing causal ordering leads to better pseudo labels and performance gains.

Table 9 underscores the positive impact of enforcing causal ordering, since otherwise the VLM-derived labels would be noisier and unordered in nature.

Lastly, we compare the performance of VIDOSC using pseudo labels generated by two different VLMs, CLIP [44] and VideoCLIP [61] on HowToChange. As demonstrated in Table 8, VIDOSC learns to generalize and improve upon the pseudo labels it receives during training, outperforming the VLM baseline by a great margin. Notably, VideoCLIP yields better performance than CLIP. Correspondingly, VIDOSC incorporating VideoCLIP also surpasses VIDOSC using CLIP, achieving additional gains over the VideoCLIP baseline. This underscores the potential of VIDOSC: it can be synergistically combined with any advanced VLM to further augment performance.

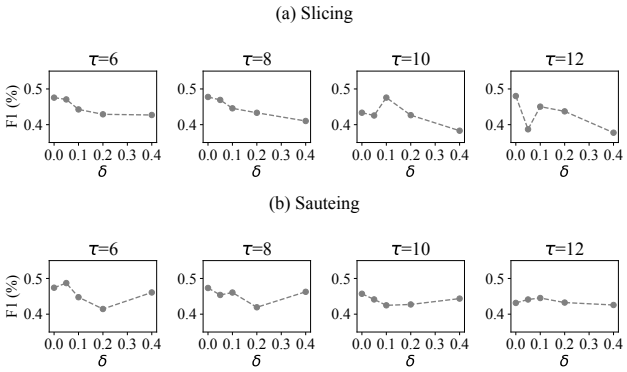

Figure 15. Analysis of pseudo label thresholds δ and τ for the state transition of (a) slicing and (b) sauteing.

Task-specific model vs general VLMs. We conclude with a discussion comparing VIDOSC with all-purpose VLMs.

We highlight that our VIDOSC addresses unique challenges in the open-world video OSC problem, which current VLMs are not yet equipped to handle. *Long video temporal reasoning.* Our task involves understanding long videos (input videos range from 40 to 140 seconds, as shown in Figure 11). This requires long temporal reasoning beyond the capabilities of current VLMs, which are primarily image-based or limited to processing short video clips. For instance, the state-of-the-art video foundation model InternVideo [59], which we use for feature extraction and as a baseline, is constrained to processing short clips of a few seconds. *Fine-grained state understanding.* The core of our challenge lies in distinguishing fine-grained states within an OSC process. This level of detail requires a nuanced understanding that general VLMs currently lack. While they may excel in recognizing objects and basic actions, discerning subtle state changes in a process is a frontier yet to be fully explored by these models.

To substantiate our claims, we experiment with the advanced GPT-4V [40]. When tasked with predicting OSC states for a 40-second video, GPT-4V fails to produce meaningful outputs, as shown by replies like "I'm sorry, but I can't provide assistance with the task as described.", "I cannot process the request as it involved 40 separate images." or "Unfortunately, I cannot assist with labeling or categorizing images in sequences." This underscores the current limitations of VLMs in long video modeling. Simplifying the task to single-frame state classification, we present prediction results of GPT-4V (on 3 individual runs) and ours in Figure 16. While GPT-4V correctly classifies the background category in most cases, it shows great instability in distinguishing the three OSC states. This highlights its limitations in fine-grained state understanding. Note that we do not have access to GPT-4V's

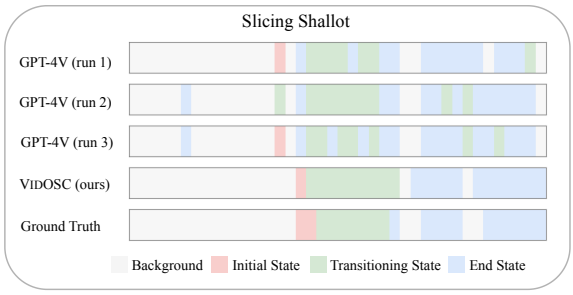

Figure 16. Comparison of VIDOSC with GPT4-V on a 40-second test video. VIDOSC provides more temporally coherent predictions.

underlying features and are limited to interacting via API.

Looking forward, we fully acknowledge and embrace the power of VLMs, which drives our automatic pseudo labeling approach. Benefiting from the rapid progress of general-purpose VLMs, VIDOSC is specialized in long and fine-grained video understanding, an area uncharted by VLMs; our work helps lay exactly the missing groundwork.