Chop & Learn: Recognizing and Generating Object-State Compositions

Nirat Saini* Hanyu Wang* Archana Swaminathan Vinoj Jayasundara Bo He
Kamal Gupta Abhinav Shrivastava

University of Maryland, College Park

Abstract

Recognizing and generating object-state compositions has been a challenging task, especially when generalizing to unseen compositions. In this paper, we study the task of cutting objects in different styles and the resulting object state changes. We propose a new benchmark suite Chop & Learn, to accommodate the needs of learning objects and different cut styles using multiple viewpoints. We also propose a new task of Compositional Image Generation, which can transfer learned cut styles to different objects, by generating novel object-state images. Moreover, we also use the videos for Compositional Action Recognition, and show valuable uses of this dataset for multiple video tasks. Project website: https://chopnlearn.github.io.

1. Introduction

Objects often exist in different shapes, colors, and textures in the real-world. These visually discernible properties of objects, also known as states or attributes, can be inherent to an object (*e.g.*, color) or be a result of an action (*e.g.*, chopped). Generalization to unseen properties of objects remains an Achilles heel of current data-driven recognition models (*e.g.*, deep networks) that assume robust training data available for exhaustive object properties. However, humans (and even animals) [3, 6] can innately imagine and recognize a large number of objects with varying properties, by composing a few known objects and their states. This ability to synthesize and recognize new combinations from finite concepts, called *compositional generalization* is often absent in modern deep learning models [30].

Several recent works have been proposed to study composition in terms of the disentanglement of objects and the states in images [24, 33, 54, 69] as well as videos [2, 4, 11, 15, 16, 19, 53, 57, 58]. A few works have attempted to improve open-world text-to-image generation models [12, 51] for the task of compositional generation. However, current suite of datasets lacks either granular annotations for object states or enough data to study how object states evolve under different conditions. Therefore, measuring the compositional generalizability of these models on different tasks remains an open challenge.

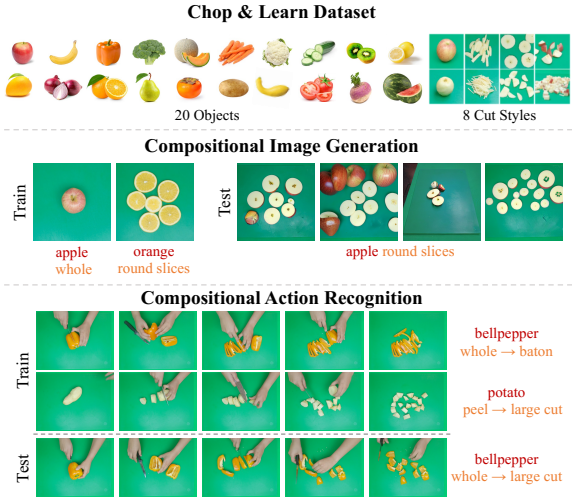


Figure 1. We present **Chop & Learn** (**ChopNLearn**), a new dataset and benchmark suite for the tasks of Compositional Image Generation and Compositional Action Recognition. It consists of 1260 video clips and 112 object state combinations captured from multiple viewpoints for 20 objects and 8 cut styles. We also propose two new compositional tasks and benchmarks - (1) Image Generation: given training images of various objects in various states, the goal is to generate images of unseen combinations of objects and states. (2) Action Recognition: training videos are used to recognize objects along with transition from state1 → state2, to generalize on recognizing unseen object-state transitions.

In this paper, we propose a new dataset, **Chop & Learn** (**ChopNLearn**) collected to support studying compositional generalization, the ability to recognize and generate unseen compositions of objects in different states. To focus on the compositional aspect, we limit our study to a common task in our daily lives – cutting fruits and vegetables. When using different styles of cutting, these objects undergo different transformations and the resulting states are easily recognizable by humans. Our goal is to study how these different styles can be applied to a variety of

*First two authors contributed equally.

Table 1. **Comparison with other video datasets.** This table highlights the distribution of the objects, states and compositions in different datasets. Obj. refers to objects, Comp. is compositions of objects and styles, N refers to the number of compositions that have more than 10 samples, and Styles* refers to grouping of styles: instead of generic names like cut, chop, etc., we use 3 distinct styles (chop/dice, peel, grate) as styles. MIT-States† is the only image-based dataset, the rest are video-based datasets. All these data numbers are for edible objects and cutting style actions from respective datasets. Our dataset has uniform distribution for each metric in the table, which makes it suitable for learning objects and their states.

Datasets	Total # of				Avg. # of Samples			N	# of Views
	Samples	Obj.	Comp.	Styles*	/Obj.	/Comp.	/Style		
MIT-States† [25]	1676	27	52	4	62.07	32.23	419	48	1
Youcook2 [72]	714	160	313	3	7.3	2.2	166.7	26	1
VISOR [8]	301	58	122	3	5.2	2.5	42.9	3	1
COIN [61]	390	6	7	2	65	55	195	6	1
Ego4D [13]	216	12	12	3	18.2	18	54.5	8	1
50Salads [59]	904	5	6	2	182	152	457	6	1
ChangeIt [57]	264	8	14	4	46.3	26.4	96	14	1
CrossTask [73]	1150	7	8	2	164.3	143.7	575	8	1
Breakfast [29]	1055	3	4	2	351.7	263.8	527.5	4	1
ChopNLearn	**1260**	**20**	**112**	**8**	**74.2**	**11.8**	**185.5**	**112**	**4**

objects for recognizing unseen object states. More specifically, we select *twenty* objects and *seven* commonly used styles of cuts (plus whole object) which results in object-state pairs with different granularity and sizes (Figure 1). We collect videos of these objects being from *four* different viewpoints, and label different object states in each video. Each style of cut changes the visual appearance of different objects in different ways. To study and understand object appearance changes, we propose two new benchmark tasks of Compositional Image Generation and Compositional Action Recognition, with a focus on unseen compositions.

The objective of the first task is to generate an image based on an (object, state) composition that was not seen during training. As shown in Figure 1, during training, a generative model is provided with images of an (`apple`, `whole`) as well as an (`orange`, `round slices`). At the test time, the model has to synthesize a new unseen composition (`apple`, `round slices`). We propose to adapt large-scale text-to-image generative models for this task. Specifically, by using text prompts to represent the object-state composition, we benchmark several existing methods such as Textual Inversion [12] and DreamBooth [51]. We also propose a new method by introducing new tokens for objects and states and simultaneously fine-tuning language and diffusion models. Lastly, we discuss the challenges and limitations of prior works as well as the proposed generative model with an extensive evaluation.

In the second task, we extend an existing task of Compositional Action Recognition [35]. While the focus of prior work [35] is on long-term activity tracking in videos, we aim to recognize subtle changes in object states which is a crucial first step for activity recognition. By detecting the initial state and final object state compositions, our task allows the model to learn unseen object state changes robustly. We benchmark multiple recent baselines for video tasks on the ChopNLearn dataset.

Finally, we discuss various other applications and tasks that can use our dataset in image and video domains. To summarize, our contributions are threefold:

- We propose a new dataset ChopNLearn, consisting of a large number of images and videos of diverse object-state compositions with multiple camera views.
- We introduce the task of Compositional Image Generation, which goes beyond the common conditional image generation benchmarks, and focuses on generating images for unseen object and state compositions.
- We introduce a new benchmark for the task of Compositional Action Recognition, which aims at understanding and learning changes in object states over time and across different viewpoints.

2. Related Work

Object states or attributes have recently received significant attention for recognition tasks, in images and videos. Some of the common works and their dissimilarities with the proposed dataset are mentioned here.

Attributes of Objects. In the image domain, states are often referred to as attributes for Compositional Learning of attribute-object pairs. Attributes describe the visual properties of objects, such as shape, color, structure and texture. The common datasets used are MIT-states [24], UT-Zappos [69], COCO-attributes [41], CGQA [34] and VAW [43]. All of these datasets consist of web-scraped images of various types of objects (from furniture to shoes and clothes to food items), which makes the variety of states very diverse. Most of the prior works [31, 33, 34, 39, 42, 44, 54, 56, 66, 68] focus on attribute-object recognition tasks using compositional learning but do not expand to image generation tasks due to the diversity in background and attributes. Some works in compositional zero-shot learning of attributes show visual disentanglement of attributes from objects [54, 64], however, they only hallucinate compositions of unseen attribute-object pairs in the feature space, rather than the image space. Moreover, even newer large vision-language models such as CLIP [46], DALL-E [48] fail to capture the subtle attributes of objects which are visually discernible [36, 70]. Therefore, the image generation task for objects with different attributes is still unexplored, which is a major focus of our work.

States for Action Recognition. Detecting object states and corresponding actions from videos is explored in supervised [2, 4, 11, 53] and self-supervised manners [10, 57,

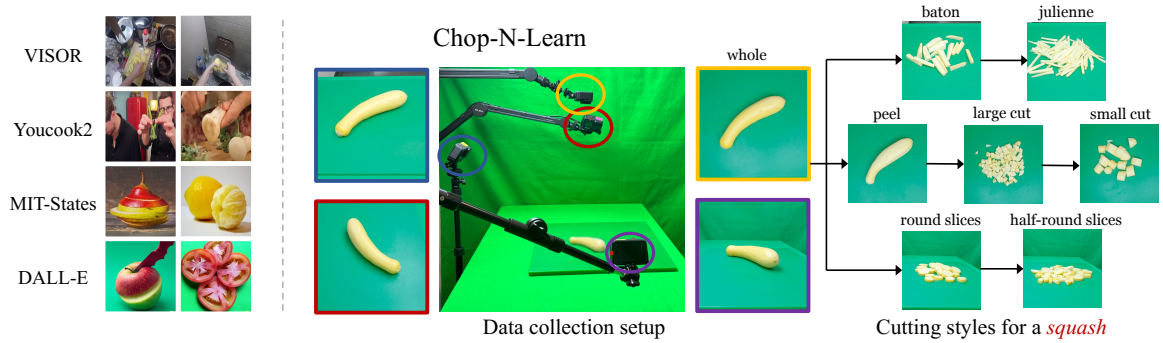


Figure 2. Left: We show examples of cutting styles from popular video datasets (VISOR [8]: chop and peel potato, Youcook2 [72]: chop broccoli, peel radish), image dataset (MIT-states [24]:slice pear, peel orange) and generation pipelines (DALL-E [48]:baton cut apple, half round slices tomato). Most of these are either too noisy to capture subtle differences in objects or do not have the granularity of specific cutting styles. Center: Our 4 camera setup captures videos of one object in 4 different views. Right: We capture 8 styles of object states, which can be derived in a hierarchical manner from larger to small cuts. Each style is of different shape and granularity.

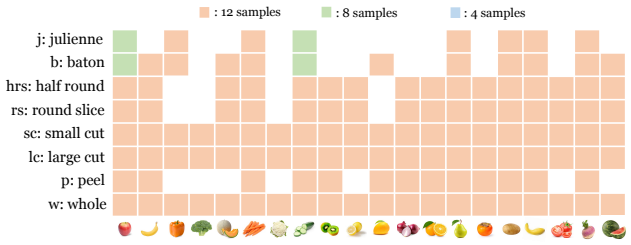
58]. While some works focus on recognizing actions using states [2, 4, 11, 53], others discover states as the future frames in the videos in [10, 26]. Some works [57, 58] also detect the exact frames of state 1, state 2 and the action that causes transition from state 1 $\rightarrow$ 2. Another recent work (Ego4D [13]) also proposes new tasks like point-of-return state-change prediction for object state transition detection. Hence, object states so far have been used as a signal for detecting and localizing actions. We focus on extending this understanding of states to generalize across different objects with limited seen object-state transition videos.

Compositional Action Recognition. In contrast to randomly assigning samples for training and testing, [35] presented a new task of Compositional Action Recognition. The premise of this task is: actions are split based on objects they apply on. During training, only a set of objects are seen corresponding to set of objects, while during testing, unseen object appear for seen action labels. Following studies [28, 32, 45, 63, 67] used relationship between objects and states bounding boxes to model the compositional aspect, where the evaluation is performed on how well the composition of unseen object and state is recognized. We propose a similar task, where videos are trained on seen compositions and tested on unseen compositions.

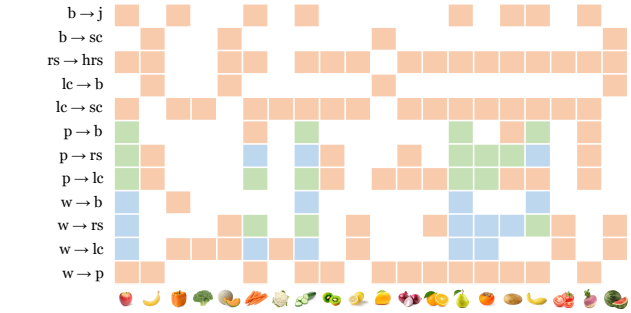
Comparison with existing Datasets. The existing image datasets such as MIT-states [24], UT-Zappos [69], COCO-attributes [41], CGQA [34] and VAW [43], are not suitable for image generation tasks for two reasons: 1) there are very few transferable objects and attributes, 2) the images are web-scraped and very diverse with varied background. Due to this, generative models latch on background details rather than understanding subtle changes in objects. In video domain, there have been various video datasets with procedural and kitchen activities that capture object and state transformations, such as Epic-Kitchens [7] with object and hand bounding box annotation version VISOR [8], Youcook2 [72], Ego4D [13], COIN [61], HowTo100M [38], Breakfast [29], 50Salads [59], CrossTask [73] and ChangeIt [57]. There are a few common problems across these datasets: (1) Most of these datasets lack annotations for the granularity of cutting styles. The styles labeled are `cut`, `chop`, `slice`, `dice`, `peel`, `grate`, `julienne`, which only comprises of three broader styles of transformations, *i.e.* `chop/dice`, `peel` and `grate`. (2) The compositions of different objects and states are highly skewed and similar to image datasets. Some datasets have a long-tail distribution of objects, which can make it challenging for models to learn per-object-based states when there is only one sample available in the dataset. And lastly (3), the frames are noisy with lots of objects and attributes that object states changes are harder to capture (as shown in left side of Figure 2). For most datasets, the ground truth is also not annotated for object detection, which makes it even harder to look for object of interest. Using an object detector to remove the background is an option, however with deformable objects, most Faster-RCNN [49] based object detectors fail to capture the object itself, and latch onto smaller pieces instead. In Table 1, we show statistics of data available in different datasets. The # of clips from other datasets that has granular annotations of object-state pairs and can be used for compositional tasks. For instance, COIN [61] has 180 categories with 10000 videos, but clips that have cutting styles as labels were only 390. Further, these clips only cover cut/peel actions, and cannot be categorized further based on granularity and shape of pieces. Our proposed dataset ChopN-Learn is designed to capture various objects and their cut styles, with uniformly distributed samples for 20 objects and 8 styles (including whole, 7 other cut styles Figure 2).

3. Chop & Learn

Our main objective with Chop & Learn (ChopNLearn) is to understand and learn granular object states, specifically styles of cuts which can be applied to diverse variety of objects. With this in focus, we collect object state transition videos, as well as images of object in various states, with



(a) Image Statistics



(b) Video Clips Statistics

Figure 3. **Statistics for ChopNLearn:** We show the number of samples for each object-style composition in a color-coded manner: orange represents 12 samples, green represents 8 samples and blue represents 4 samples.

4 different camera views (Figure 2). We discuss the design choices and motivation below.

3.1. Design Choices

Selection of States (styles of cuts). Fruits and vegetables are commonly cut in specific styles based on the need of the recipes. For instance, for eating an apple, we slice it in relatively large pieces while for using it in a pie, we might cut smaller or round slices of it. We select 8 common styles of cuts, *i.e.*, `large cut`, `small cut`, `baton`, `julienne`, `round slices`, `half round slices`, `peel`, and `whole` for our study. These are the most common styles of cuts for vegetables and fruits, which do not require any additional training to learn apart from common kitchen operation and knife handling skills. These styles of cuts can also have similarities with respect to shapes, yet are different in granularity. For example, `baton` (french-fries style cut) and `julienne` are similar in shape (long pieces), but `julienne` is more finely cut than `baton`. Similarly, `large cut` is a coarser version of `small cut`, and `half round slice` is one step from `round slices` (as shown in Figure 2). We also have annotated the states `whole` and `peel`, which are the base states of objects.

Selection of Objects. We want to learn to transfer styles of cuts to different objects. To ensure consistency in transfer, we also consider the base state, *i.e.*, `whole` state of objects. For instance, it is hard to visualize `large cut` of carrots, if the seen data only includes rounder objects like `oranges`. Hence, we consider some fruits and vegetables with similar colors, textures and shapes to include consistency across visual similarities after chopping. In this study, we used seasonal fruits and vegetables categorised on the basis on their shapes, colors and textures: round small objects: [`apple`, `pear`, `mango`, `potato`, `turnip`, `onion`, `kiwi`], citrus fruits [`lemon`, `orange`], flower-like textured objects: [`cauliflower`, `broccoli`] , larger round objects: [`cantaloupe`, `watermelon`], textured from inside objects: [`bellpepper`, `tomato`, `persimmon`], and long objects: [`cucumber`, `carrot`, `squash`, `banana`]. This consists of 10 fruits and 10 vegetable items, with at least one pair of similar objects presents in the dataset.

Related Groups. One of the key aspects of this dataset is transferability of cut styles to a variety of objects. We set up some constraints and create related groups for objects and styles. These related group enable us with structural and visual style transfer abilities. If an object is seen from related group A with a particular style, we should be able to transfer that style to another object from the same related group A and vice-versa. In other words, we group sets of objects and cut styles which are visually similar (based on color, shape and texture) together to create related groups for objects and states separately. For states, we combine [`baton`, `julienne`], [`round slices`, `half-round slices`], and [`large cut`, `small cut`] together as related groups. For objects, we define seven groups with related objects: [`apple`, `pear`, `mango`], [`lemon`, `orange`], [`cauliflower`, `broccoli`], [`cantaloupe`, `watermelon`, `kiwi`], [`bellpepper`, `tomato`, `persimmon`], [`potato`, `turnip`, `onion`], and [`cucumber`, `carrot`, `squash`, `banana`].

3.2. Data Collection Setup

We collect data using four GoPro cameras [1] positioned at different angles, with three participants (Figure 2). We use a green screen and green chopping board for minimum distraction in the background, such that the objects and their cut pieces are easily segmented for each view.

Granularity of styles. For ease and consistency across participants, the size of cut pieces can be defined as the shape and ratio of one piece with respect to the whole object. For more details, please refer to the appendix. Given a set of n states and m objects, we can have at most $m \times n$ compositions. However, our dataset does not include some compositions which are not commonly found in real world. For instance, due to the texture of `onions`, it is not feasible to cut `onions` in `baton` or `julienne` style, since the layers of the `onion` do not stay intact, so we do not have a sample of [`baton`, `onion`].

Video Recording. We primarily collect video data, and derive state change frames from long videos. Each video consists of 2-3 object states, which are annotated while data collection process using the highlight features of GoPros. For

Figure 4. **(a)** The clip length distribution for one camera (315 unique clips). **(b)** Preliminary results of using green screen to augment the dataset with different backgrounds. We continue to improve the transfer results by adding shadows and background matting.

Table 2. **Compositional generation evaluation.** FID, user scores, and classifier scores of various generative models. User Realism is on a scale of 1-5. ($\star$) denotes that accuracies are evaluated on a seen data split. **Bold** represents the best result.

Method	Patch FID ↓	User Realism ↑	Classifier Acc. (%) Object ↑	Classifier Acc. (%) State ↑	User Acc. (%) Object ↑	User Acc. (%) State ↑
Real Images	-	4.65	87.5*	92.0*	73.6	84.0
SD	178.0	3.41	**73.1**	27.9	**81.6**	28.8
SD+TI	145.0	2.58	23.6	37.7	21.6	43.2
DreamBooth	139.9	3.56	53.5	74.2	61.6	72.8
SD+FT	88.9	**3.78**	70.5	67.7	72.0	65.6
SD+FT+TI	**82.2**	3.47	67.8	**81.4**	67.2	**79.2**

synchronizing across different cameras, we initially start with a clapper to make a clap sound for indicating the beginning of the video. Then, we highlight the frames in one of the GoPro as the first/initial state. The participant then walks up the object and starts cutting the object. After the object is cut in one style, the participant steps back and we highlight another frame as the next state. The participant performs at least 2 styles of cut in each video, which can be done consecutively. For instance, we can first cut an object with `large cuts`, and then do `small cuts` subsequently. The video ends with another clap for the end of video detection and synchronization across different cameras. Henceforth, we collect video data along with annotated states for each participant, without extra effort of annotations. More details and statistics of dataset are shown in Figure 3. Average video clip length (one state change for an object) is 1m40s. The distribution is shown in Fig. 4(a).

4. Compositional Image Generation

Large-scale deep generative models [47, 50, 52] trained on open-world big datasets have made significant breakthroughs in image generation in the last couple of years. These models, are typically conditioned using a text encoder and also support tasks such as zero-shot image generation, inpainting, image editing, and super-resolution without explicit training on these tasks. However, the performance of these models significantly degrades when it comes to compositional generation [9]. Our dataset, consisting of 112 real-world object and state combinations, is well-suited to test the compositional capabilities of generative models.

Task Description. The goal of the task is to either train from scratch or fine-tune an existing generative model using the (object, state) pairs provided in the training, and generate images from unseen compositions. We consider all 20 objects, each object captured in up to 7 different states, *i.e.*, all the states excluding `peel`. We split the (object, state) combinations into a training set consisting of 87 combinations and a test set consisting of 25 combinations. The training set covers all objects and states used in our dataset, but it does not overlap with the test set in terms of (object, state) combinations. In other words, for each combination of object and state present in the test set, the training set includes exactly one of either the object, or the state, but not both. We also ensure that for each (object, state) combination (o, s_i) in the test set, there exists a combination (o, s_j) in the training set, where s_i and s_j belong to the same state related group defined in Section 3.1. This setting ensures that all object and state information are available in the training set. Each combination in our dataset has 8-12 images, resulting in a total of 1032 images in the training set and 296 images in the test set. The exact split is provided in the appendix along with some examples.

4.1. Methods

Stable Diffusion. (SD) We evaluate a popular open-source text-to-image generative model Stable Diffusion (SD) [50]. For details on the SD, refer to the original work [50]. Here we briefly describe the sampling process. Diffusion models generate an image from Gaussian noise via an iterative denoising process. SD uses classifier-free guidance [21] for sampling. This means given a text prompt $\mathbf{c}$, we encode the prompt using CLIP's text classifier [46] and recursively update a Gaussian noise sample with

$$\omega\epsilon_\theta(\mathbf{x}_t, \mathbf{c}) + (1-\omega)\epsilon_\theta(\mathbf{x}_t) \quad (1)$$

where $\mathbf{x}_t$ is the denoised sample at the time step t and ϵ_θ is SD. With each time step, we try to move the denoised sample using the guidance provided by the text prompt. The strength of the guidance is defined by ω.

As our first baseline approach, we sample zero-shot images from SD with a text prompt "An image of o_i cut in s_j style", where o_i is the i^{th} object and s_j is the j^{th} state of the object. Zero-shot generation with a pre-trained SD model doesn't work as intended as shown in Figure 5, and the generated images often perform poorly in capturing the

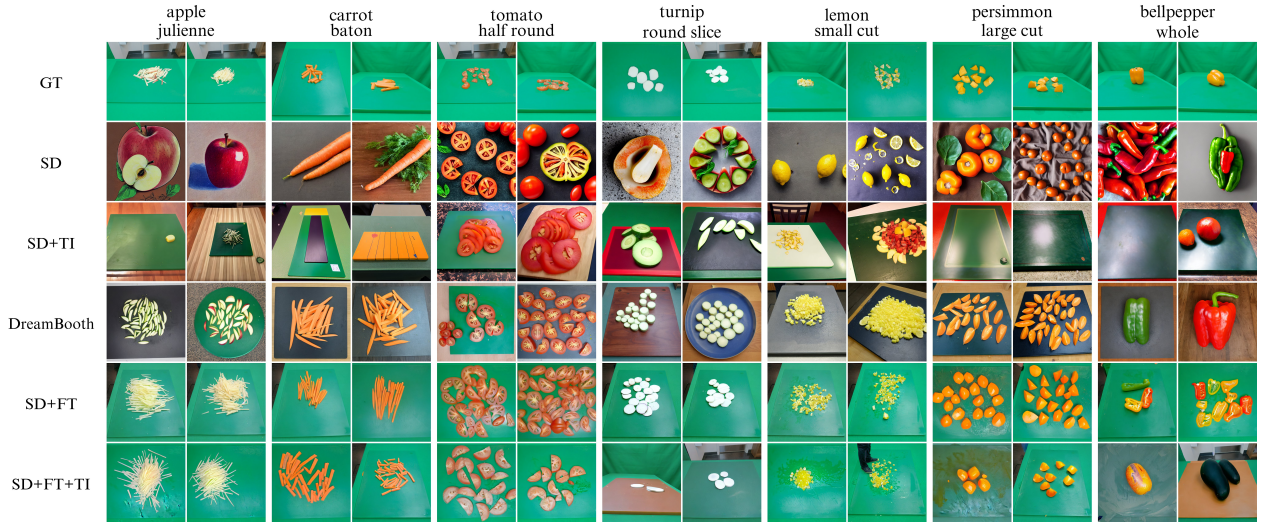


Figure 5. **Compositional Generation Samples.** Ground Truth (GT) real images are shown in the first row for reference. Seven object-state combinations in the test set are displayed, each with two generated samples for each method. Please zoom in to see details.

object state. Several recent works have shown that it is possible to extend models such as SD to achieve high-quality customized generations [12, 51, 71]. We evaluate several methods that have been proposed for compositional generation in the recent literature. We also propose a simple yet strong baseline by fine-tuning a Stable Diffusion (SD) model [50] along with textual inversion.

SD + Textual Inversion (TI). Textual Inversion [12] introduces new tokens in the vocabulary and optimizes their embedding from the given images keeping SD frozen. We adapt the method for our task by introducing new tokens for the objects $\{o_i\}$ and the states $\{s_j\}$, and jointly optimize the embeddings of $\{o_i\} \cup \{s_j\}$ by providing (image, prompt) pairs from our training data. As before, the prompt is simply constructed as "An image of o_i cut in s_j style".

DreamBooth. Next, we adapt DreamBooth [51], which fine-tunes the diffusion model along with the state-specific tokens. In our experiments, we fine-tune one model for each state in the dataset, where only the state token is learned. Original DreamBooth optimizes the diffusion loss as well as a prior preservation loss [51]. We observed that the latter significantly deteriorates the performance thus we skip it.

SD + Fine-tuning (FT). We also fine-tune SD. In this baseline, only the parameters in the UNet of the diffusion model are optimized while keeping the text encoder fixed. **SD + TI + FT.** Finally, we combine SD fine-tuning and Textual Inversion [12]. Specifically, on top of our SD + Fine-tuning baseline, we also adapt Textual Inversion by introducing new object tokens and state tokens and optimizing their embeddings along with the UNet parameters.

4.2. Evaluation

We use both qualitative and quantitative measures to evaluate the capabilities of different methods. This section explains the details of different evaluation metrics we used:

Patch FID. Fréchet Inception Distance (FID) [20] is a commonly used metric to assess the quality of generative models. Given a set of real images and a set of generated images, FID compares the mean and std of Inception-v3 features of the two sets. For each composition and generative model, we compute patch FID using all real and 16000 generated patches, and report the average number for the test pairs. We hypothesize that using patch FID gives more weight to the object-state patches, rather than the whole image, which includes almost 50% background pixels. We further calculate the lower bound for patch FID score by computing it between two sets of real images. Any score lower than that for this dataset can be disregarded as irrelevant. The determined lower bound for the patch FID score is 37.2.

Object/State Accuracy using a Classifier. To evaluate the correctness of objects and states in the generated images, we train a classifier on real images for classifying objects and states independently. This classifier is built on top of CLIP-ViT-B/32 [46]. Classification logits are obtained by computing the cosine similarity between the image embedding and text embeddings of all possible state labels or object labels. To ensure the reliability of the classifier's results, we train it on the training set from a different dataset split, where all (object, state) combinations are present.

User Study. We conducted a user study to evaluate the generated images. We took images from the test set as well as samples from our generative models and present them to 30 users. Each user was presented with 25 distinct images, randomly sampled with an even distribution from our models and the test set. After giving a tutorial to the users about the different objects and states present in our experiments, the users were asked to choose an appropriate object name and state label, as well as rate the image for realism on a scale of 1-5. We report the object and state accuracies as well as realism score in Table 2. The details of our user study design can be found in the appendix.

Table 3. **Compositional action recognition results.** "Start/End" denote the prediction results for the initial and the final state composition with the corrected object type. **Bold** and underline represent the top-1 and top-2 results.

		Split 1				Split 2				Split 3			
		Start		End		Start		End		Start		End	
Model	Features	acc@1	acc@3	acc@1	acc@3	acc@1	acc@3	acc@1	acc@3	acc@1	acc@3	acc@1	acc@3
AvgPool	I3D [5]	9.5	23.7	4.7	14.2	8.3	21.9	5.2	19.8	15.9	28.5	4.8	22.3
LSTM [22]	I3D [5]	14.2	36.2	5.7	29.8	12.5	29.2	6.2	26.0	17.5	34.9	6.3	23.7
Transformer [62]	I3D [5]	23.7	49.0	10.9	44.3	27.5	46.2	14.6	44.2	20.6	42.9	11.1	44.4
AvgPool	MIL-NCE [37]	11.1	31.6	4.8	28.4	9.4	17.7	5.2	13.5	14.2	41.4	12.8	41.4
LSTM [22]	MIL-NCE [37]	15.9	36.5	6.4	36.6	11.9	36.7	9.8	36.7	18.9	39.6	8.0	25.4
Transformer [62]	MIL-NCE [37]	50.9	85.7	47.7	76.2	**56.2**	82.3	52.7	88.5	41.1	74.6	42.9	77.7
STLT [45]	–	2.8	15.5	1.4	8.4	1.4	13	1.4	11.6	4.2	14.1	1.4	11.3
Transformer [62]	R3D [14]	45.1	85.9	52.1	85.9	55.1	94.2	**58.0**	92.8	59.1	85.9	56.3	85.9
CAF [45]	R3D [14]	**53.5**	**88.7**	**57.8**	**88.7**	55.1	**95.7**	**58.0**	**95.7**	**62.0**	**93.0**	**63.4**	**93.0**

4.3. Results and Discussion

Qualitative Results. Fig. 5 displays the generated images from various methods for seven (object, state) combinations in the test set. The first row of the figure exhibits the ground truth real images for reference. We observe that vanilla SD often generates correct objects in random states, while SD+TI frequently synthesizes images without displaying the object. DreamBooth performs better than SD+TI, but worse than a simple finetuning of SD. SD+FT and SD+FT+TI perform well in terms of state generation.

Quantitative Results. Table 2 displays the performance of all baseline methods evaluated according to the metrics outlined in Section 4.2. Assessing image realism is a crucial evaluation metric for generative models; however, defining and measuring it can be challenging. Note that the patch FID values and user realism ratings do not align well. This is due to the disparity between the distribution of images in our dataset and that of typical occurrence of those objects in the real world. The patch FID metric measures the similarity between the generated images with those in our dataset, instead of the ones most typical in real world. In particular, our results indicate that SD achieves the worst patch FID score since it has not encountered our dataset before, whereas its user realism rating is more satisfactory. SD+TI has the lowest user realism rating and a poor patch FID score, which suggests that only training object/state embeddings is inadequate for generating high-quality images. DreamBooth receives a good user realism rating but a poor patch FID, indicating that the images it generates are realistic but not very similar to those in our proposed dataset. Finally, fine-tuning via both SD+FT and SD+FT+TI achieve better results for patch FID and user realism.

We next evaluate the accuracy of objects and states in generated images. It is worth noting that the classification task on our dataset is intrinsically difficult, which leads to imperfect user accuracy on real images. In general, the accuracy scores from classifier closely align with one from users, indicating that the proposed classifier is suited for evaluating compositional generation.

Our results show that SD achieves the best object accuracy but the worst state accuracy. This is possibly due to the lack of state variations in most existing large image datasets. SD+TI is the worst performer due to its limited learning capacity. On the other hand, DreamBooth, SD+FT, and SD+FT+TI attain better state accuracy. Among them, DreamBooth's object accuracy is slightly worse as it is particularly trained for states. SD+FT achieves high object accuracy, and SD+FT+TI attains the best state accuracy with the help of fine-tuning and textual inversion together.

Green Screen Removal. One of the main challenges for understanding fine-grained object-state pairs with existing datasets such as MIT-states [25] is diverse backgrounds. Using them for training often leads to the model latching on to unwanted background details and missing out on the state understanding. Hence, we collected ChopNLearn with a clean green screen background for the benchmark tasks. While we acknowledge the limitations it poses to our trained models, we highlight that the green screen can potentially enhance our ability to generalize to diverse scenes. This can be achieved by segmenting out images and placing various backgrounds, along with scaled and rotated object-state images (Figure 4). As a proof-of-concept, we train a SD+FT+TI model on background-augmented images, and report the Patch FID, classifier object accuracy and state accuracy in Tab. 4. Note that here we employ a newly trained classifier that uses background-augmented images, and the patch FID scores are also computed based on these images. We further reference the lower bound of the patch FID as defined in Section 4.2. Due to the complex backgrounds introduced, the object accuracy and the patch FID of the new model are slightly compromised. However, it maintains a high and even improved state accuracy. This demonstrates the potential of the background-augmented ChopNLearn in enhancing fine-grained compositional image generation.

5. Compositional Action Recognition

Human actions often change object states and different objects can have diverse visual transitions even when sub-

Table 4. **Green screen removal evaluation.** Both rows employ the SD+FT+TI but are trained using images with varying backgrounds. Classifiers specific to each dataset are trained to assess Classifier Acc. Validation images used to calculate Patch FID differ between the two rows. Patch FID Lower Bound is computed by evaluating the patch FID on one-half of the validation images relative to the other half. For further details, refer to Section 4.3.

Data Background	Classifier Acc. (%) Object ↑	State ↑	Patch FID ↓	Patch FID Lower Bound
Green Screen	67.8	81.4	82.2	37.2
Various	46.3	82.3	133.6	46.4

jected to the same action type. To investigate this problem in a more intuitive manner, [35] introduced a new task of compositional action recognition, which targets at improving the robustness of models to handle the same actions with different objects involved. For example, given an action of 'taking something out from something', the model is trained on a limited set of objects and is tested on unseen types of objects to access its generalizability. Hence, despite the same underlying action, the object and visual features can be quite diverse. Similarly, the composition of the same action with different object types can look very distinctive. For instance, although cutting an `carrot` and a `apple` require similar knife movements, the resulting visual changes are distinct, with the former changing from a `whole apple` to a `peeled apple`, and the latter changing from a `whole carrot` to a `peeled carrot`. Therefore, we propose to use our dataset for the task of compositional action recognition, which can also be referred to as Compositional Zero-Shot Action Recognition, as the compositions of objects and states are unseen during training.

Task Description. For this task, we consider each clip of a video as containing a single object with a single state transition. From the raw videos, which typically contain 2-3 transitions of object states per video, we segment the clips into isolated ones with only one transition. Examples of transitions include changing from a `whole` object to a `peeled` object or from a `peeled` object to a `baton` cut object. Similar to [35], we divide all object-final state compositions into two sets: seen compositions, which are used for training, and unseen compositions, which are used for testing. Following the approach used in the Compositional Image Generation task, we ensure that each object and state are seen at least once individually in the training set, but not together as a composition. The objective of the task is to predict the correct labels for the initial object-state composition (o_i, s_j) and the final composition (o_i, s_k), given a clip containing an object o_i transitioning from an initial state s_j to a final state s_k. Note that the clip is considered correctly classified only if both the object and state labels are correct for both the initial and final compositions.

5.1. Dataset Splits

We create 3 different dataset splits as follows (more details are in the Appendix). All splits have disjoint train, test and validation samples, and are created with different constraint combinations:

- **Split 1:** This split is a random selection of object-final state compositions with cross-view condition. We do not use any information from related groups.
- **Split 2:** In this split, we use related group information for states, along with cross-view. based on related groups, if `baton carrots` is seen in training set, then `julienne carrots` can be part of test set. Since `baton` and `julienne` are part of the same related group, we can learn an object in one style and can generalize to another style from the same group in Section 3.1.
- **Split 3:** This split includes information from both related groups for states and objects. We want to ensure that even if an object is not seen in its related group, a similar object is seen in the related group. For example, if `broccoli` is seen with `large cuts`, then `cauliflower` with large or small cuts can be in the test set.

Hence different splits represent different complexity levels for compositional action recognition.

Evaluation. We evaluate the accuracy of predicting both the initial and final compositions of objects and states in the test set. Only when the object and state are both correct, it is counted as a correct prediction. Specifically, we use two separate prediction heads for objects and states. We emphasize the need to evaluate composition as a whole, rather than just predicting the state, as the way an `apple` is cut can differ significantly from the way a `bellpepper` is cut. Therefore, accurately recognizing both the object and state is crucial for tasks related to understanding and generating videos of object states. We also recognize the importance of top@3 accuracy, since object states can sometimes be visually similar, leading to confusion in detecting the correct composition. For example, `julienne apple` can be visually very similar to `julienne potato`.

5.2. Results

To evaluate our proposed method, we establish baselines using both traditional architectures and features for video action classification, as well as comparing with recent works in compositional action recognition. As shown in Table 3, in the first section, we use pre-extracted I3D[5] features and conduct experiments by comparing simple average pooling, LSTM, and multi-layer Transformer [62] model. It shows that the Transformer model performs the best among these variants due to the great capacity of temporal modeling ability. In the second section, we also experiment with more recent pre-trained features MIL-NCE [37] along with transformer models, which outperforms I3D features. MIL-NCE [37] features are pre-trained on HowTo100M [38] with multimodal (RGB+narrations) setup, which is more robust for video downstream tasks.

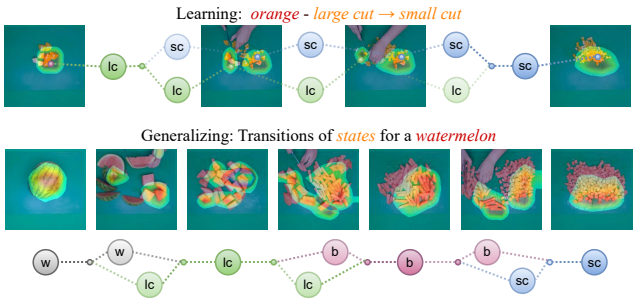


Figure 6. **Video parsing graph:** For a given video, we use Grad-CAM[55] on the intermediate frames to identify and visualize the class activation maps corresponding to the most salient states. Top: A training video clip has one transition of `orange` from `large cut` → `small cut`. Bottom: We can learn single transitions from training data, to generalize transitions in a long video with multiple state changes and parse the video as a graph.

In the final section of Table 3, we employ the state-of-the-art compositional video recognition model proposed in [45] and use pseudo labels of bounding boxes for each hand and object, as there are no ground-truth hand and object trajectories available. Specifically, the Spatial-Temporal Layout Transformer (STLT) [45] takes in the spatio-temporal locations and class labels for each bounding box as input, uses positional embeddings to project coordinates and class labels into features, and adds transformer layers to model spatial-temporal relationships. However, without any appearance information, STLT achieves low performance on all metrics. On the other hand, with the appearance features, which are extracted by inflated 3D ResNet50 [27] (R3D), it can achieve much higher performances than STLT. Finally, Cross-Attention Fusion (CAF) applies cross-attention [60] to fuse the layout (STLT) and appearance (R3D) branch embeddings, achieving the best results. It demonstrates that combining the layout and appearance information together can help predict object and state types more accurately.

6. Discussion

We discuss the potential future use of ChopNLearn, while addressing the limitations and scope as well.

Long-term Video Parsing. We use compositional state recognition to further understand the temporal dynamics [17, 18] with the aid of a video parsing graph construction as previously explored in Ego-Topo [40] and VideoGraph [23]. Each clip in the training set has one state transformation (top example in Figure 6). We visualize the class activation maps corresponding to the most salient intermediate state transitions with Grad-CAM [55], to learn the transition in each frame of the video for training data. This is illustrated as a graph for a training video. Having learned multiple single transformations, we can now extend this knowledge to understand long activities, with multiple transitions. As shown in Fig. 6, we can learn state changes for `orange` from `large cut` → `small cut` using our training clip. Given a long unseen video with multiple clips, we can construct a state-transition graph to represent changes in state for a `watermelon`. Hence, by using an extensive array of videos, the process of learning transitions between individual states can be extended to encompass transitions between multiple states. This enables the creation of a self-supervised transition knowledge graph for comprehensive long-term video comprehension, as demonstrated in [10, 65].

Limitations. With advent of foundation models, few-shot generalization is an increasingly important task. In this work, we explore the potential of ChopNLearn for the research in compositional generation and recognition for highly complex and interdependent concepts. Admittedly, ChopNLearn is a small scale dataset with green screen background, which restricts the models trained on it to have specific biases. Nonetheless, this is the first attempt to understand how fine-grained states (cut styles) can be transferred to diverse objects. We explore this by using ChopNLearn as a test set for larger models, fine-tuning these models using ChopNLearn and trying them with or without a green screen background. We further see the potential of using ChopNLearn for benefiting the community in even more challenging tasks such as 3D reconstruction, video frame interpolation, state change generation, *etc*.

7. Conclusion

In this paper, we propose ChopNLearn, a new dataset for measuring the ability of models to recognize and generate unseen compositions of objects in different states, a skill known as compositional generalization. We also introduce two tasks, Compositional Image Generation and Compositional Action Recognition, and benchmark the performance of state-of-the-art generative models and video recognition methods on these tasks. We show the challenges with the existing approaches and their failure in some cases in their ability to generalize to new compositions. However, these two tasks are just the tip of the iceberg. Understanding object states is important for multiple image and video tasks such as 3D reconstruction, future frame prediction, video generation, summarization, and parsing of long-term video. We hope that our dataset will help the computer vision community to propose and learn new compositional tasks for images, videos, 3D, and beyond.

Acknowledgements. The authors would like to dedicate this paper to the memory of Vinoj Jayasundara. His creativity, contributions and enthusiasm for the field of Computer Vision will continue to inspire us. We would also like to thank Snehesh, Chahat, Kanishka, and Pulkit for their valuable conversations during data collection. This work was partially funded by DARPA SAIL-ON (W911NF2020009) program and NSF CAREER Award (#2238769) to AS.

References

[1] Hero9 black 5k video 20mp photo streaming camera and bundles. https://gopro.com/en/us/shop/cameras/hero9-black/CHDHX-901-master.html. 4

[2] Jean-Baptiste Alayrac, Josef Sivic, Ivan Laptev, and Simon Lacoste-Julien. Joint discovery of object states and manipulation actions. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2146–2155, 2017. 1, 2, 3

[3] Kate Arnold and Klaus Zuberbühler. Semantic combinations in primate calls. *Nature*, 441(7091):303–303, 2006. 1

[4] Nachwa Abou Bakr, James L. Crowley, and Rémi Ronfard. Recognizing manipulation actions from state-transformations. *ArXiv*, abs/1906.05147, 2019. 1, 2, 3

[5] João Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4724–4733, 2017. 7, 8

[6] Noam Chomsky. A minimalist program for linguistic theory. 1993. 1

[7] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Scaling egocentric vision: The epic-kitchens dataset. *ArXiv*, abs/1804.02748, 2018. 3

[8] Ahmad Darkhalil, Dandan Shan, Bin Zhu, Jian Ma, Amlan Kar, Richard E. L. Higgins, Sanja Fidler, David F. Fouhey, and Dima Damen. Epic-kitchens visor benchmark: Video segmentations and object relations. *ArXiv*, abs/2209.13064, 2022. 2, 3

[9] Miranda Dixon-Luinenburg. What dall-e 2 can and cannot do, 2023. 5

[10] Dave Epstein, Jiajun Wu, Cordelia Schmid, and Chen Sun. Learning temporal dynamics from cycles in narrated video. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1460–1469, 2021. 2, 3, 9

[11] 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, 3

[12] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022. 1, 2, 6

[13] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Q. Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, Miguel Martin, Tushar Nagarajan, Ilija Radosavovic, Santhosh K. Ramakrishnan, Fiona Ryan, Jayant Sharma, Michael Wray, Mengmeng Xu, Eric Z. Xu, Chen Zhao, Siddhant Bansal, Dhruv Batra, Vincent Cartillier, Sean Crane, Tien Do, Morrie Doulaty, Akshay Erapalli, Christoph Feichtenhofer, Adriano Fragomeni, Qichen Fu, Christian Fuegen, Abrham Gebreselasie, Cristina González, James M. Hillis, Xuhua Huang, Yifei Huang, Wenqi Jia, Weslie Yu Heng Khoo, Jáchym Kolár, Satwik Kottur, Anurag Kumar, Federico Landini, Chao Li, Yanghao Li, Zhenqiang Li, Karttikeya Mangalam, Raghava Modhugu, Jonathan Munro, Tullie Murrell, Takumi Nishiyasu, Will Price, Paola Ruiz Puentes, Merey Ramazanova, Leda Sari, Kiran K. Somasundaram, Audrey Southerland, Yusuke Sugano, Ruijie Tao, Minh Vo, Yuchen Wang, Xindi Wu, Takuma Yagi, Yunyi Zhu, Pablo Arbeláez, David J. Crandall, Dima Damen, Giovanni Maria Farinella, Bernard Ghanem, Vamsi Krishna Ithapu, C. V. Jawahar, Hanbyul Joo, Kris Kitani, Haizhou Li, Richard A. Newcombe, Aude Oliva, Hyun Soo Park, James M. Rehg, Yoichi Sato, Jianbo Shi, Mike Zheng Shou, Antonio Torralba, Lorenzo Torresani, Mingfei Yan, and Jitendra Malik. Ego4d: Around the world in 3,000 hours of egocentric video. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18973–18990, 2021. 2, 3

[14] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Learning spatio-temporal features with 3d residual networks for action recognition. *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 3154–3160, 2017. 7

[15] Bo He, Jun Wang, Jielin Qiu, Trung Bui, Abhinav Shrivastava, and Zhaowen Wang. Align and attend: Multimodal summarization with dual contrastive losses. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14867–14878, 2023. 1

[16] Bo He, Xitong Yang, Le Kang, Zhiyu Cheng, Xin Zhou, and Abhinav Shrivastava. Asm-loc: action-aware segment modeling for weakly-supervised temporal action localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13925–13935, 2022. 1

[17] Bo He, Xitong Yang, Le Kang, Zhiyu Cheng, Xin Zhou, and Abhinav Shrivastava. Asm-loc: Action-aware segment modeling for weakly-supervised temporal action localization. 2022. 9

[18] Bo He, Xitong Yang, Hanyu Wang, Zuxuan Wu, Hao Chen, Shuaiyi Huang, Yixuan Ren, Ser-Nam Lim, and Abhinav Shrivastava. Towards scalable neural representation for diverse videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 9

[19] Bo He, Xitong Yang, Zuxuan Wu, Hao Chen, Ser-Nam Lim, and Abhinav Shrivastava. Gta: Global temporal attention for video action understanding. *British Machine Vision Conference (BMVC)*, 2021. 1

[20] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 6

[21] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 5

[22] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 7

[23] Noureldien Hussein, Efstratios Gavves, and Arnold W. M. Smeulders. Videograph: Recognizing minutes-long human activities in videos. *ArXiv*, abs/1905.05143, 2019. 9

[24] Phillip Isola, Joseph J. Lim, and E. Adelson. Discovering states and transformations in image collections. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1383–1391, 2015. 1, 2, 3

[25] Phillip Isola, Joseph J. Lim, and Edward H. Adelson. Dis-

covering states and transformations in image collections. In *CVPR*, 2015. 2, 7

[26] Dinesh Jayaraman, Frederik Ebert, Alexei A. Efros, and Sergey Levine. Time-agnostic prediction: Predicting predictable video frames. *ArXiv*, abs/1808.07784, 2018. 3

[27] Hirokatsu Kataoka, Tenga Wakamiya, Kensho Hara, and Yutaka Satoh. Would mega-scale datasets further enhance spatiotemporal 3d cnns? *arXiv preprint arXiv:2004.04968*, 2020. 9

[28] Tae Soo Kim and Gregory Hager. Safcar: Structured attention fusion for compositional action recognition. *ArXiv*, abs/2012.02109, 2020. 3

[29] H. Kuehne, A. B. Arslan, and T. Serre. The language of actions: Recovering the syntax and semantics of goal-directed human activities. In *Proceedings of Computer Vision and Pattern Recognition Conference (CVPR)*, 2014. 2, 3

[30] Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. Building machines that learn and think like people. *Behavioral and brain sciences*, 40:e253, 2017. 1

[31] Yong-Lu Li, Yue Xu, Xiaohan Mao, and Cewu Lu. Symmetry and group in attribute-object compositions. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11313–11322, 2020. 2

[32] Jian Ma and Dima Damen. Hand-object interaction reasoning. *2022 18th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–8, 2022. 3

[33] Massimiliano Mancini, Muhammad Ferjad Naeem, Yongqin Xian, and Zeynep Akata. Open world compositional zero-shot learning. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2021. 1, 2

[34] Massimiliano Mancini, Muhammad Ferjad Naeem, Yongqin Xian, and Zeynep Akata. Learning graph embeddings for open world compositional zero-shot learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP:1–1, 2022. 2, 3

[35] Joanna Materzynska, Tete Xiao, Roei Herzig, Huijuan Xu, Xiaolong Wang, and Trevor Darrell. Something-else: Compositional action recognition with spatial-temporal interaction networks. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1046–1056, 2019. 2, 3, 8

[36] Moustafa Meshry, Yixuan Ren, Larry S Davis, and Abhinav Shrivastava. Step: Style-based encoder pre-training for multi-modal image synthesis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[37] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9876–9886, 2019. 7, 8

[38] 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. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2630–2640, 2019. 3, 8

[39] Tushar Nagarajan and K. Grauman. Attributes as operators: Factorizing unseen attribute-object compositions. In *ECCV*, 2018. 2

[40] Tushar Nagarajan, Yanghao Li, Christoph Feichtenhofer, and Kristen Grauman. Ego-topo: Environment affordances from egocentric video. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 160–169, 2020. 9

[41] Genevieve Patterson and James Hays. Coco attributes: Attributes for people, animals, and objects. In *ECCV*, 2016. 2, 3

[42] Khoi Pham, Kushal Kafle, Zhe Lin, Zhihong Ding, Scott Cohen, Quan Tran, and Abhinav Shrivastava. Improving closed and open-vocabulary attribute prediction using transformers. In *European Conference on Computer Vision*, pages 201–219. Springer, 2022. 2

[43] Khoi Pham, Kushal Kafle, Zhe Lin, Zhi Ding, Scott D. Cohen, Quan Tran, and Abhinav Shrivastava. Learning to predict visual attributes in the wild. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13013–13023, 2021. 2, 3

[44] Senthil Purushwalkam, Maximilian Nickel, Abhinav Gupta, and Marc'Aurelio Ranzato. Task-driven modular networks for zero-shot compositional learning. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3592–3601, 2019. 2

[45] Gorjan Radevski, Marie-Francine Moens, and Tinne Tuytelaars. Revisiting spatio-temporal layouts for compositional action recognition. *British Machine Vision Conference (BMVC)*, abs/2111.01936, 2021. 3, 7, 9

[46] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 2021. 2, 5, 6

[47] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 5

[48] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. *ArXiv*, abs/2102.12092, 2021. 2, 3

[49] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1137–1149, 2015. 3

[50] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 5, 6

[51] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. *arXiv preprint arXiv:2208.12242*, 2022. 1, 2, 6

[52] Chitwan Saharia, William Chan, Saurabh Saxena, Lala

Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022. 5

[53] 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, 3

[54] Nirat Saini, Khoi Pham, and Abhinav Shrivastava. Disentangling visual embeddings for attributes and objects. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13648–13657, 2022. 1, 2

[55] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 9

[56] Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Constrained semi-supervised learning using attributes and comparative attributes. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part III 12*, pages 369–383. Springer, 2012. 2

[57] 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. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13936–13946, 2022. 1, 2, 3

[58] Tomáš Souček, Jean-Baptiste Alayrac, Antoine Miech, Ivan Laptev, and Josef Sivic. Multi-task learning of object state changes from uncurated videos. *ArXiv*, abs/2211.13500, 2022. 1, 3

[59] Sebastian Stein and Stephen J. McKenna. Combining embedded accelerometers with computer vision for recognizing food preparation activities. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, page 729–738, 2013. 2, 3

[60] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*, 2019. 9

[61] Yansong Tang, Dajun Ding, Yongming Rao, Yu Zheng, Danyang Zhang, Lili Zhao, Jiwen Lu, and Jie Zhou. Coin: A large-scale dataset for comprehensive instructional video analysis. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1207–1216, 2019. 2, 3

[62] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 7, 8

[63] X. Wang and Abhinav Kumar Gupta. Videos as space-time region graphs. *ArXiv*, abs/1806.01810, 2018. 3

[64] Xin Wang, Fisher Yu, Trevor Darrell, and Joseph E Gonzalez. Task-aware feature generation for zero-shot compositional learning. *arXiv preprint arXiv:1906.04854*, 2019. 2

[65] Donglai Wei, Joseph J. Lim, Andrew Zisserman, and William T. Freeman. Learning and using the arrow of time. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 9

[66] Ziwei Xu, Guangzhi Wang, Yongkang Wong, and Mohan S Kankanhalli. Relation-aware compositional zero-shot learning for attribute-object pair recognition. *IEEE Transactions on Multimedia*, 2021. 2

[67] Rui Yan, Peng Huang, Xiangbo Shu, Junhao Zhang, Yonghua Pan, and Jinhui Tang. Look less think more: Rethinking compositional action recognition. *Proceedings of the 30th ACM International Conference on Multimedia*, 2022. 3

[68] Muli Yang, Cheng Deng, Junchi Yan, Xianglong Liu, and Dacheng Tao. Learning unseen concepts via hierarchical decomposition and composition. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10245–10253, 2020. 2

[69] A. Yu and K. Grauman. Fine-grained visual comparisons with local learning. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 192–199, 2014. 1, 2, 3

[70] Tian Yun, Usha Bhalla, Elizabeth-Jane Pavlick, and Chen Sun. Do vision-language pretrained models learn primitive concepts? *ArXiv*, abs/2203.17271, 2022. 2

[71] Lvmin Zhang and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543*, 2023. 6

[72] Luowei Zhou, Nathan Louis, and Jason J. Corso. Weakly-supervised video object grounding from text by loss weighting and object interaction. In *British Machine Vision Conference*, 2018. 2, 3

[73] Dimitri Zhukov, Jean-Baptiste Alayrac, Ramazan Gokberk Cinbis, David Fouhey, Ivan Laptev, and Josef Sivic. Cross-task weakly supervised learning from instructional videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3537–3545, 2019. 2, 3