Is an Object-Centric Video Representation Beneficial for Transfer?

Chuhan Zhang[1], Ankush Gupta[2], and Andrew Zisserman[1]

[1] Visual Geometry Group, Department of Engineering Science
University of Oxford
{czhang,az}@robots.ox.ac.uk
[2] DeepMind, London
ankushgupta@google.com

Abstract. The objective of this work is to learn an *object-centric* video representation, with the aim of improving transferability to novel tasks, i.e., tasks different from the pre-training task of action classification. To this end, we introduce a new object-centric video recognition model based on a transformer architecture. The model learns a set of object-centric summary vectors for the video, and uses these vectors to fuse the visual and spatio-temporal trajectory 'modalities' of the video clip. We also introduce a novel trajectory contrast loss to further enhance objectness in these summary vectors.
With experiments on four datasets—SomethingSomething-V2, SomethingElse, Action Genome and EpicKitchens—we show that the object-centric model outperforms prior video representations (both object-agnostic and object-aware), when: (1) classifying actions on unseen objects and unseen environments; (2) low-shot learning of novel classes; (3) linear probe to other downstream tasks; as well as (4) for standard action classification.

1 Introduction

Visual data is complicated—a seemingly infinite stream of events emerges from the interactions of a finite number of constituent *objects*. Abstraction and reasoning in terms of these entities and their inter-relationships—*object-centric reasoning*—has long been argued by developmental psychologists to be a *core* building block of infant cognition [1], and key for human-level common sense [2]. This object-centric understanding posits that objects exist [3], have permanence over time, and carry along physical properties such as mass and shape that govern their interactions with each other. Factorizing the environment in terms of these objects as recurrent entities allows for combinatorial generalization in novel settings [2]. Consequently, there has been a gradual growth in video models that embed object-centric inductive biases, e.g., augmenting the visual stream with actor or object bounding-box trajectories [4,5,6,7], graph algorithms on object nodes [8,9], or novel architectures for efficient discovery, planning and interaction [10,11,12].

The promise of object-centric representations is transfer across *tasks*. Due to the shared underlying physics across different settings, knowledge of object

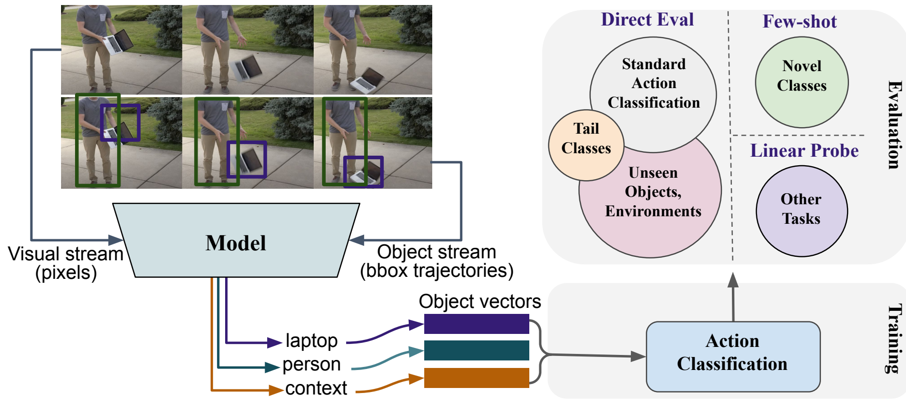

Fig. 1. Do Object-Centric video representations transfer better? To induce objectness into visual representations of videos, we learn a set of object-centric vectors—which are tied to specific objects present in the video, as well as context vectors—which reason about relations and context. The representation is built by fusing the two modality streams of the video—the visual stream, and the spatio-temporal object bounding-box stream. We train the model on the standard action recognition task, and show that using the object and context vectors can lead to SOTA results when evaluated for transfer to unseen objects, unseen environments, novel classes, other tasks and also standard action classification.

properties like shape, texture, and position can be repurposed with little or no modification for new settings [13], much like infants who learn to manipulate objects and understand their properties, and then apply these skills to new objects or new tasks [14,15].

In this paper we investigate this promise by developing an *object-centric* video model and determining if it has superior task generalization performance compared to object-agnostic and other recent object-centric methods. In a similar manner to pre-training a classification model on ImageNet, and then using the backbone network for other tasks, we pre-train our object-centric model on the action recognition classification task, and then determine its performance on downstream tasks using a linear probe.

We consider a model to be object-centric if it learns a set of object summary vectors, that *explicitly* distil information about a specific object into a particular latent variable. In contrast, in object-agnostic [16,17,18,19] or previous state-of-the-art object-centric video models [4,5,20], the object information is de-localized throughout the representation.

To this end, we introduce a novel architecture based on a transformer [21] that achieves an object-centric representation through its design and its training. First, a bottleneck representation is learned, where a set of object query vectors [22] tied to specific constituent objects, cross-attend in the manner of DETR [23] to visual features and corresponding bounding-box trajectories. We demonstrate this cross-attention based fusion is an effective method for merging the two modality streams [24,4,5]—visual and geometric—complementing the individual streams. We call this 'modality' fusion module an *Object Learner*. Second, a novel *trajectory contrast* loss is introduced to further enhance object-awareness in the object summaries. Once learnt, this explicit set of object summary vectors are repurposed and refined for downstream tasks.

We evaluate the task generalization ability of the object-centric video representation using a number of challenging transfer tasks and settings:

1. **Unseen data:** Action classification with known actions (verbs), but novel objects (nouns) in the SomethingElse dataset [20]; Action classification with known actions (verbs and nouns), but unseen kitchens in EpicKitchens [25].
2. **Low-shot data:** Few-shot action classification in SomethingElse; Tail-class classification in EpicKitchens.
3. **Other downstream tasks:** Hand contact state estimation in SomethingElse, and human-object predicate prediction in ActionGenome [26].

Note, task 3 uses a linear probe on pre-trained representations for rigorously quantifying the transferability. In addition to evaluating the transferability as above, we also benchmark the learned object-centric representations on the standard task of action classification. In summary, our key contributions are:

1. A new object-centric video recognition model with explicit object representations. The object-centric representations are learned by using a novel cross-attention based module which fuses the visual and geometric streams, complementing the two individually.
2. The object-centric model sets a new record on a comprehensive set of tasks which evaluate transfer efficiency and accuracy on unseen objects, novel classes and new tasks on: SomethingElse, Action Genome and EpicKitchens.
3. Significant gains over the previous best results on standard action recognition benchmarks: 74.0%(+6.1%) on SomethingSomething-V2, 66.6%(+6.3%) on Action Genome, and 46.3%(+0.6%) top-1 accuracy on EpicKitchens.

2 Related Work

Object-centric video models. Merging spatio-temporal object-level information and visual appearance for video recognition models has been explored extensively. These methods either focus solely on the human actors in the videos [6,27,7], or more generally model human-object interactions [28,29,30,31]. The dominant approach involves RoI-pooling [32,33] features extracted from a visual backbone using object/human bounding-boxes generated either from object detectors [34], or more generally using a region proposal network (RPN) [6,35,27,36,8,37] on each frame independently, followed by global aggregation using recurrent models [38]. The input to these methods is assumed to just be RGB pixels, and the object boxes are obtained downstream. A set of object-centric video models [4,20,5] assume object boxes as *input*, and focus on efficient fusion of the two streams; we follow this setting. Specifically, ORViT [4] is an object-aware vision transformer [39] which incorporates object information in two ways: (1) by attending to RoI-pooled object features, and (2) by merging encoded trajectories at several intermediate layers. STIN [20] encodes the object boxes and identity independently of the visual stream, and merges the two through concatenation before feeding into a classifier. STLT [5] uses a transformer encoder on object boxes, first across all objects in a given frame, and then across frames, before fusing with appearance features. We adopt STLT's hierarchical trajectory encoder, and develop a more performant cross-attention based fusion method.

Multi-modal fusion. Neural network architectures which fuse multiple modalities, both within the visual domain, i.e., images and videos [40] with optical flow [24,41], bounding-boxes [6,35,27,36,8,37], as well as across other modalities, e.g., sound [42,43] and language [44,45,46,47], have been developed. The dominant approach was introduced in the classic two-stream fusion method [24] which processes the visual and optical flow streams through independent encoders before summing the final softmax predictions. Alternative methods [41] explore fusing at intermediate layers with different operations, e.g., sum, max, concatenation, and attention-based non-local operation [48]. We also process the visual and geometric streams independently, but fuse using a more recent cross-attention based transformer decoder [49] acting on object-queries [22]. An alternative to learning a single embedding representing all the input modalities, is to learn modality encoders which all map into the same joint vector space [50,51,44]; such embeddings are primarily employed for retrieval.

Benchmarks with object annotations. Reasoning at the *object* level lies at the heart of computer vision, where standard benchmarks for recognition [52], detection and segmentation [53,54], and tracking [55,56,57,58] are defined for categories of objects. Traditionally, bounding-box tracking of single [55,56] or multiple objects [57,58], or more spatially-precise video object segmentation [59,60,61,62] were the dominant benchmarks for object-level reasoning in videos. More recently, a number of benchmarks probe objects in videos in other ways, e.g., ActionGenome [26] augments the standard action recognition with human/object based scene-graphs, SomethingElse [20] tests for transfer of action recognition on novel objects, CATER [63] evaluates compositional reasoning over synthetic objects, and CLEVERER [64] for object-based counterfactual reasoning.

Object-oriented reasoning. There is a large body of work on building in and reasoning with object-level inductive biases across multiple domains and tasks. Visual recognition is typically *defined* at the object-level both in images [54,65,66,67] and videos [34,68,26,69]. Learning relations, expressed as edges, between entities/particles, expressed as nodes in a graph has been employed for amortizing inference in simulators and modelling dynamics [70,11]. Such factorized dynamics models conditioned on structured object representations have been employed for future prediction and forecasting [71,72,73]. Object-conditional image and video decomposition [74,75,10,76] such as Monet [77] and

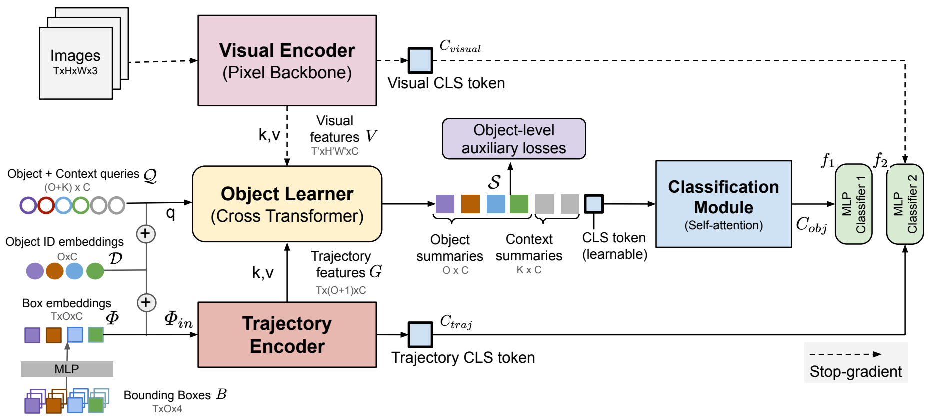

Fig. 2. The object-centric video transformer model architecture. The *Visual Encoder* module ingests the RGB video and produces a set of object agnostic spatio-temporal tokens. The *Trajectory Encoder* module ingests the bounding boxes, and labels them with object ID embedddings $\mathcal{D}$, to produce object-aware trajectory spatio-temporal tokens. The *Object Learner* module fuses the visual and trajectory streams by querying with the object IDs $\mathcal{D}$, and outputs object summaries which contains both visual and trajectory information. An object-level auxiliary loss is used to encourage each object summary vector to be tied to the object in the query. Finally, the *Classification* module ingests the outputs from the Object Learner to predict the class. The model is trained with cross-entropy losses applied to class predictions from the dual encoders and Object Learner, together with an auxiliary loss.

Genesis [78] and generation [79,80,81,82,83,84] methods benefit from compositional generalization. Finally, object-level world-models have been used to constrain action-spaces, and states in robotics [85,86] and control domains [87,12,88].

3 An Object-Centric Video Action Transformer

We first describe the architecture of the object-centric video action recognition model for fusing visual and trajectory streams. We then describe the training objectives for action classification and for learning the object representations. Finally, we discuss our design choices, and the difference between our model and previous fusion methods, and explain its advantages.

3.1 Architecture

The model is illustrated in Figure 2, and consists of four transformer-based modules. We briefly describe each module, with implementation details in section 4.

Video Encoder. The encoder ingests a video clip F of RGB frames $F = (f_1, f_2, \ldots, f_t)$, where $f_i \in \mathbb{R}^{H\times W\times 3}$. The clip F is encoded by a Video Transformer [49] which tokenizes the frames by 3D patches to produce downsampled feature maps. These feature maps appended with a learnable `CLS` token are processed by self-attention layers to obtain the spatio-temporal visual representations $V \in \mathbb{R}^{T \times H'W' \times C}$ and video-level visual embedding C_{visual}. We take the representation V from the 6th self-attention layer to be the visual input of the Object Learner, and C_{visual} from the last layer to compute the final loss in eq. (2).

Trajectory Encoder. The encoder ingests the bounding box coordinates $B^t = (b_1^t, b_2^t, \ldots, b_o^t)$ of $\mathcal{O}$ number of annotated objects in the t^{th} frame, where each box b_i^t is in format $[x_1, y_1, x_2, y_2]$. These boxes are encoded into corresponding box embeddings $\Phi^t = (\Phi_1^t, \Phi_2^t, \ldots, \Phi_o^t)$ through an MLP. Object ID embeddings $\mathcal{D} = \{d_i\}_{i=1}^{O}$ are added to Φ to produce the sum Φ_{in}, This is to keep the ID information persistent throughout the video length T; Φ_{in} serves as the input to the Trajectory Encoder, which is a Spatial Temporal Layout Transformer (STLT) of [5]. STLT consists of two self-attention Transformers in sequence – a Spatial Transformer and a Temporal Transformer. First, the Spatial Transformer encodes boxes in every frame separately. It takes a learnable `CLS` token and box embeddings $\Phi_{in}^t \in \mathbb{R}^{O \times C}$ from frame t as the input into the self-attention layers, and output a frame-level representation $l^t \in \mathbb{R}^{1 \times C}$ and spatial-context-aware box embeddings $\Phi_{out}^t \in \mathbb{R}^{O \times C}$ respectively. The Temporal Transformer models trajectory information over frames, it applies self-attention on the frame-level embeddings $L = (l^1, l^2, l^3, \ldots, l^T)$ from the Spatial Transformer with another learnable `CLS` token. Its output are temporal-context-aware frame embeddings $L_{out} \in \mathbb{R}^{T \times C}$ and a video-level trajectory representation $C_{traj} \in \mathbb{R}^{1 \times C}$. C_{traj} is used to compute the final loss in eq. (2), while $L_{out} \in \mathbb{R}^{T \times 1 \times C}$ is concatenated with the $\Phi_{out} \in \mathbb{R}^{T \times O \times C}$ from the Spatial Transformer to be the spatio-temporal trajectory embeddings $G \in \mathbb{R}^{T \times (O+1) \times C}$, which are used as trajectory input to the Object Learner. (See Supp. for detailed architecture.)

Object Learner. The Object Learner module is a cross-attention Transformer [21] which has a query set $\mathcal{Q} = \{q_i\}_{i=1}^{O+K}$ made up of O learnable object queries and K learnable context queries. The same ID embeddings $\mathcal{D}$ from the Trajectory Encoder are added to the first O queries to provide object-specific identification, while the remaining K context queries can be learnt freely. We concatenate the visual feature maps $V \in \mathbb{R}^{T \times H'W' \times C}$ and trajectory embeddings $G \in \mathbb{R}^{T \times (O+1) \times C}$ as keys and values in the cross-attention layers. Note the query latents are video level (i.e., common across all frames), and attend to the features from the visual and trajectory encoders using cross-attention. The Object Learner outputs summary vectors $\mathcal{S} = \{s_i\}_{i=1}^{O+K}$, O of which are object centric, and the remaining K carry context information. The output is independent of the number of video frames, with the visual and trajectory information distilled into the summary vectors. Figure 3 presents a schematic of the module.

Classification Module. This is a light-weight cross-attention transformer that ingests the summary vectors output from the the Object Learner, together with a learnable query vector C_{obj}. The vector output of this module is used for a linear classifier for the actionss prediction.

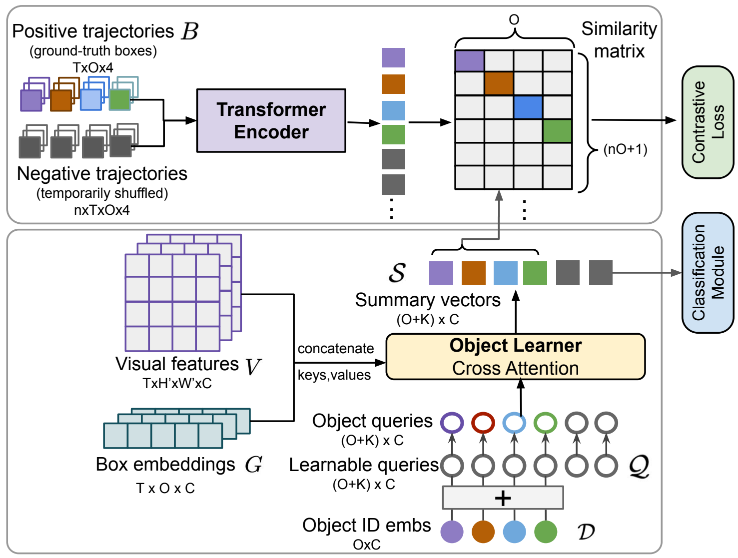

Fig. 3. Object Learner and auxiliary loss. The Object Learner is a cross-attention transformer, that outputs object-level summary vectors by attending to tokens from both the visual and trajectory encoders. These summary vectors are used for downstream tasks like classification. An auxiliary loss is added where the object summary vectors are tasked with distinguishing GT and shuffled trajectory embeddings.

3.2 Objectives

We apply two types of losses to train the model. One is an object-level auxiliary loss on the object summary vectors $S = (s_1, s_2, \ldots, s_o)$ to ensure object-centric information is learned in these vectors. The other is a standard cross-entropy loss on action category prediction.

Object-level Trajectory Contrast loss. The aim of this loss is to encourage the object specific latent queries in the Object Learner to attend to both the trajectory and visual tokens, and thereby fuse the information from the two input modalities. The key idea is to ensure that the video-level object queries do not ignore the identity, position encodings, and trajectory tokens. The loss is a contrastive loss which encourages discrimination between the correct trajectories and others that are randomly perturbed or from other video clips in the batch. This is implemented using an InfoNCE [89] loss, with a small transformer encoder used to produce vectors for each of the trajectories. This encoder only consists of two self-attention layers that encode the object trajectories into vectors.

In more detail, for an object j, the transformer takes its ground-truth trajectory $B_j \in \mathbb{R}^{T\times 4}$ and outputs the embedding $z_j \in \mathbb{R}^C$ as the positive to be matched against the summary vector $\hat{s}_j \in \mathbb{R}^C$. For negatives, other trajectories in the same batch as well as n new ones generated by temporally shuffling B_j

encoded into $z_j^{shuffle}$ are used.

$$\mathcal{L}_{aux} = -\sum_j \left[\log \frac{\exp(\hat{s}_j^\top \cdot z_j)}{\sum_k \exp(\hat{s}_j^\top \cdot z_k) + \sum_k \exp(\hat{s}_j^\top \cdot z_k^{shuffle})} \right] \tag{1}$$

Final objective. We use two MLPs as classifiers for the CLS vectors from visual and trajectory backbones and the Object Learner. The first classifier $f_1(.)$ is applied to concatenated C_{visual} and C_{traj} CLS vectors, and the second, $f_2(.)$, is applied to the CLS vector C_{obj} from the Object Learner. The total loss is the sum of the cross-entropy loss for the two classifiers and the auxiliary loss:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{CE}(f_1(C_{visual}; C_{traj}), gt) + \mathcal{L}_{CE}(f_2(C_{obj}), gt) + \mathcal{L}_{aux}, \tag{2}$$

The final class prediction is obtained by averaging the class probabilities from the two classifiers.

Discussion: Object Learner and other fusion modules. Prior fusion methods can be categorized into three main types: (a) RoI-Pooling based methods like STRG [8], where visual features are pooled using boxes for the downstream tasks; (b) Joint training methods like ORViT [4] where the two modalities are encoded jointly from early stages; and (c) Two stream methods [5,20] with dual encoders for the visual and trajectory modalities, where fusion is in the last layer. The RoI-pooling based methods explicitly pool features inside boxes for downstream operations, omitting context outside the boxes. In contrast, our model allows the queries to attend to the visual feature maps freely. Joint training benefits from fine-grained communication between modalities, but this may not be as robust as the two-stream models under domain shift. Our method combines the two, by keeping the dual encoders for independence and having a bridging module to link the information from their intermediate layers. Quantitative comparisons are done in Section 5.

4 Implementation details

Model architecture. We use Motionformer [49] as the visual encoder, operating on 16 frames of size 224×224 pixels uniformly sampled from a video; the 3D patch size for tokenization is 2×16×16. We use STLT [5] as the trajectory encoder which takes normalized bounding boxes from 16 frames as input. Our Object Learner is a Cross-Transformer with 6 layers and 8 heads. We adopt the trajectory attention introduced in [49] instead of the conventional joint spatio-temporal attention in the layers. The Classification Module has 4 self-attention layers with 6 heads. We set the number of context queries as 2 in all the datasets, and number of object queries as 6 in SomethingElse, SomethingSomething and EpicKitchens, 37 in ActionGenome.
Training We train our models on 2 Nvidia RTX6k GPUs with the AdamW [90] optimizer. Due to the large model size and limited compute resources, we are

not able to train the full model end-to-end with a large batch size. Instead, we first train the visual backbone for action classification with batch size 8, and then keep it frozen while we train the rest of the model with batch size 72. More details on architecture and training are in the supplementary material.

5 Experiments

We conduct experiments on four datasets, namely SomethingSomething-V2, SomethingElse, Action Genome and EpicKitchens. We first train and evaluate our model on the standard task of action recognition on these datasets, and then test its transferability on novel tasks/settings including action recognition on unseen objects, few-shot action recognition, hand state classification, and scene-graph predicate prediction. We first introduce the datasets and the metrics. Followed by a comparison of our method with other fusion methods, and then ablations on the design choices in the proposed Object Learner. Finally, we compare with SOTA models on different tasks and analyze the results.

5.1 Datasets and Metrics

SomethingSomething-V2 [91] is a collection of 220k labeled video clips of humans performing basic actions with objects, with 168k training videos and 24k validation videos. It contains 174 classes, these classes are object agnostic and named after the interaction, e.g, 'moving something from left to the right'. We use the ground-truth boxes provided in the dataset as input to our networks.

Something-Else [20] is built on the videos in SomethingSomething-V2 [91] and proposes new training/test splits for two new tasks testing for generalizability: compositional action recognition, and few-shot action recognition. The compositional action recognition is designed to ensure there is no overlap in object categories between 55k training videos and 58k validation videos. In the few-shot setting, there are 88 base actions (112k videos) for pre-training, 86 novel classes for fine-tuning. We use the ground-truth boxes provided in the dataset.

Action Genome [26] is a dataset which uses videos and action labels from Charades [92], and decomposes actions into spatio-temporal scene graphs by annotating human, objects and their relationship in them. It contains 10K videos (7K train/3k val) with 0.4M objects. We use the raw frames and ground-truth boxes provided for action classification over 157 classes on this dataset.

Epic-Kitchens [25] is a large-scale egocentric dataset with 100-hour activities recorded in kitchens. It provides 496 videos for training and 138 videos for val, each video has detected boudning boxes from [93]. We use an offline tracker [94] to build association between the boxes and use them as input. We use the detected boxes provided in the dataset as input to our networks.

5.2 Ablations

Comparison with other fusion methods. For a fair comparison, we implement other fusion methods using our (i.e., the same) visual and trajectory backbones. We choose LCF (Late Concatenation Fusion) and CACNF (Cross-Attention CentralNet Fusion) as they are the two best methods among seven in the latest work [5]. We also compare against a baseline method where the class probabilities from the encoders trained independently are averaged.

We implement CACNF with the same number of cross-attention layers and attention heads as in our Object Learner. Table 1 summarizes the results. The results show that the above fusion methods work better than the baseline, and our model achieves better results than other parametric methods. Performance for Motionformer with other trajectory encoders, or STLT with other visual encoders, has been explored in previous works [4,5,20]—more comparisons are in Table 2.

Visual Enc.	Traj. Enc.	Fusion Method	Fusion Parametric	SthSth-V2 Top1	SthSth-V2 Top5
Motionformer	-	-	-	66.5	90.1
-	STLT	-	-	57	85.2
Motionformer	STLT	avg prob	✗	67.5	90.9
		CACNF [5]	✓	69.7	93.4
		LCF [5]	✓	73.1	**94.2**
		OL(ours)	✓	**74.0**	**94.2**

Table 1. Different fusion methods with the same visual backbone. We show the performance of Motionformer and STLT alone on SthSth-V2, and compare the classification performance with different fusion methods on them, namely averaged class probabilities, LCF and CACNF and our Object Learner (OL).

Ablating trajectory contrast loss. We compare the performance of training with and without the trajectory contrast loss on different transfer tasks in Table 3. Having the object-level auxiliary loss (Equation (1)) brings improvement in performance in 3 out of 4 tasks. The improvement is 1% in hand state classification in SomethingElse, 2% in scene-graph prediction and 1.7% on standard classification in Action Genome. The results show the auxiliary loss helps in both task transfer as well as standard action recognition. Figure 4 also shows the visualization of attention scores in the Object Learner – object queries trained with auxiliary loss are more object-centric when attending to the visual frames.

5.3 Results

We present the experiment results on a wide range of tasks organized into sections by the dataset used. In each dataset, we first compare the results from different models on standard action recognition, and then introduce the transfer tasks and discuss the performance.

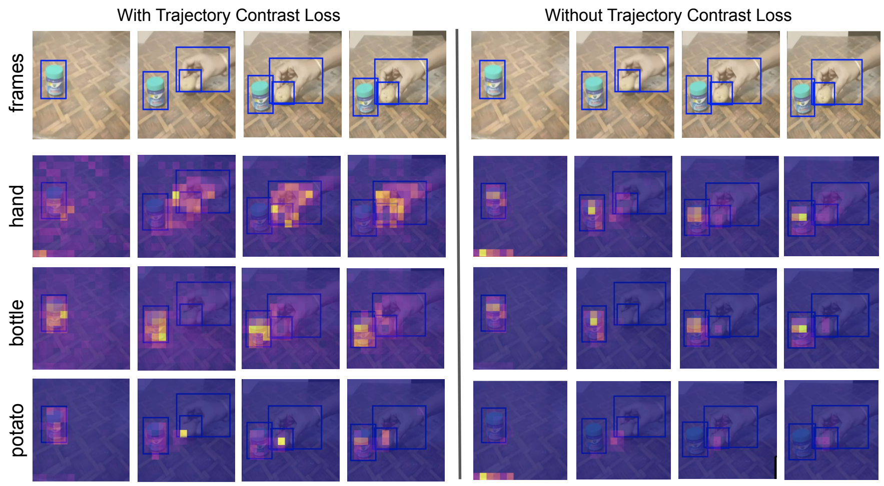

Fig. 4. Trajectory contrast loss induces object-centric attention. Object Learners's attention from various heads when trained with (left) and without (right) the trajectory contrast loss Equation (1). In both cases, attention is on individual objects, indicating objectness in the model. However, there is a notable difference: without the trajectory contrast loss, there is no attention on the hands(first row). Hence, the trajectory contrast loss induces enhanced objectness in the object summary vectors.

5.3.1 SomethingSomething-V2 and SomethingElse

Standard Action Recognition. We evaluate on the regular train/val split on SomethingSomething-V2 for the performance on (seen) action recognition. The accuracy of our model is 5.9% higher than ORViT and 7.1% higher than STLT [5], showing the advantage is not only on transfer tasks but also on standard action classification.

Transfer to Unseen Objects. Compositional action recognition is a task in Something-Else where the actions are classified given unseen objects (i.e., objects not present in the training set). Thus it requires the model to learn appearance-agnostic information on the actions. Our object-centric model improves the visual Motionfomer by 10.9%, and outperforms the joint encoding ORViT model by a margin of 3.9 %, showing that keeping the trajectory encoding independent from the visual encoder can make the representations more generalizable.

Data-efficiency: Few-shot Action Recognition. We follow the experiment settings in [20] to freeze all the parameters except the classifiers in 5-shot and 10-shot experiments on SomethingElse. Again, models that are using both visual and trajectory modalities have an obvious advantage over visual only ones. The performance boost is more obvious in a low data regime, with a 4.7% and 0.3% improvement over R3D, STLT in 5-shot and 10-shot respectively. It's worth noting that while the raw classification results from the backbone and Object

Model	Video Input	Box Input	GFLOP	SthSth-V2 Action Recognition		SomethingElse Compositional Action (Unseen Objects)		SomethingElse Few Shot	
				Top 1	Top5	Top 1	Top 5	5 shot	10 shot
I3D [16]	✓	✗	28	61.7	83.5	46.8	72.2	21.8	26.7
SlowFast,R101 [40]	✓	✗	213	63.1	87.6	45.2	73.4	22.4	29.2
Motionformer [49]	✓	✗	369	66.5	90.1	62.8	86.7	28.9	33.8
STIN [20]	✗	✓	5.5	54.0	79.6	51.4	79.3	24.5	30.3
SFI [37]	✗	✓	-	-	-	44.1	74	24.3	29.8
STLT [5]	✗	✓	4	57.0	85.2	59.0	86	31.4	38.6
STIN+I3D [20]	✓	✓	33.5	-	-	54.6	79.4	28.1	33.6
STIN,I3D [20]	✓	✓	33.5	-	-	58.1	83.2	34.0	40.6
SFI [37]	✓	✓	-	-	-	61.0	86.5	35.3	41.7
R3D,STLT(CACNF) [5]	✓	✓	48	66.8	90.6	67.1	90.4	37.1	45.5
ORViT [4]	✓	✓	405	73.8	93.6	69.7	91	33.3	40.2
Motionformer+STLT(baseline)	✓	✓	373	72.8	94.1	72.0	92.3	38.9	44.6
Motionformer+STLT+OL(Ours)	✓	✓	383.3	**74.0**	**94.2**	**73.6**	**93.5**	**40.0**	**45.7**

Table 2. Comparison with SOTA models on Something-Else and SomethingSomething-V2. We report top1 and top5 accuracy on three action classification tasks, including compositional and few-shot action recognition on Somehing-Else, and action recognition on SomethingSomething-V2. From top to bottom: we show the performance of SOTA visual models, trajectory models, and the models which takes both modalities as input. In the last section we list the classification performance from the backbone baseline without an Object Learner(OL) and our model with an Object Learner(OL), Our model outperforms other methods by a clear margin on all the tasks.

Method	Aux Loss	Something-Else Compositional Action		Something-Else Hand Contact State		Action Genome Action	Action Genome Predicate	
		Top1	Top5	Per-video	Per-class	mAP	R@10	R@20
ORViT	-	69.7	91.0	70.2	66.0	-	-	-
MFormer+STLT(baseline)	-	72	93.2	66.8	43.3	66.0	78.3	83.5
MFormer+STLT+OL(ours)	✗	73.5	**93.5**	77.5	68.5	64.9	78.9	83.8
MFormer+STLT+OL(ours)	✓	**73.6**	**93.5**	**78.2**	**69.7**	**66.6**	**80.9**	**85.4**

Table 3. Ablate auxiliary loss and Object Learner (OL) on compositional action, hand state classification and predicate prediction. We show linear probe results on the backbone CLS token.

Learner classifiers only have a 0.1% difference, averaging the two together gives more than 1% improvement. It suggests that our Object Learner has captured complementary information through combining the two streams.

Transfer to Hand State Classification. We further evaluate the object-level representations (pre-trained with standard action recognition) on hand contact state classification using a linear probe. We extract hand state labels using a pre-trained object-hand detector from [93] as ground truth, and design a 3-way classification task on SomethingElse. Specifically, the three classes are 'no hand contact', 'one hand contact' (one hand contacts with object) or 'two hands contact' (both hands contact with object). In our experiments, we average-pool the object summary vectors, train a linear classifier on the training set, and test on the validation set. We conduct the linear probe on summaries trained with and without the auxiliary loss in Equation (1), and also on the baseline backbone classifier. Video-level top-1 accuracy and class-level top-1 accuracy are reported in Table 3. Our model is better than the baseline by 11.4 % in per-video accuracy and 26.4 % in per-class accuracy. Object summaries trained with auxiliary losses on trajectories outperform the one without by about 1%.

5.3.2 Epic-Kitchens

Standard Action Recognition. Table 4 shows the results of action recognition in Epic-Kitchens, With the Object-Learner, our model is 0.6-4.0% more accurate in action prediction than other methods that use both visual stream and trajectory stream as input, and 1.7% more accurate than the Late Concatenation Fusion (LCF) method without an Object-Learner.

Classification on Tail Classes and Unseen Kitchens. In Table 4 we also present the classification results on tail classes and videos from unseen kitchens. In average, Object-centric models are better than visual-only models by 4.8% on tail actions, and by 0.4% on unseen kitchens. Among all the models with objectness, our model with Object Learner achieves the best action classification accuracy on both tail classes and unseen kitchens.

5.3.3 Action Genome Standard Action Recognition. In Action Genome, each action clip is labelled with object bounding boxes and their categories. We follow the experiment settings in [5], train and evaluate our model with RGB frames and ground truth trajectory as input. Table 5 shows the classification results. By using our Object Learner trained with auxiliary loss, we achieve the best result 66.6% mAP, outperforming other fusion methods using the same backbone. We also compare to SGFB [26], which uses scene graphs as input, our model is better by 6.3% without access to the relationship between objects.

Methods	Box input	Overall			Tail Classes			Unseen Kitchens		
		Action	Verb	Noun	Action	Verb	Noun	Action	Verb	Noun
SlowFast [40]	N	38.5	65.5	50.0	18.8	36.2	23.3	29.7	56.4	41.5
ViViT-L [95]	N	44.0	66.4	56.8	-	-	-	-	-	-
MFormer [49]	N	43.1	66.7	56.5	-	-	-	-	-	-
MFormer-HR [49]	N	44.5	67.0	58.5	19.7	34.2	28.4	34.8	58.0	46.6
MFormer-HR+STRG	Y	42.5	65.8	55.4	-	-	-	-	-	-
MFormer-HR+STRG+STIN	Y	44.1	66.9	57.8	24.7	**39.9**	34.4	34.8	59.5	48.1
MFormer-HR-ORVIT [4]	Y	45.7	68.5	57.9	-	-	-	-	-	-
MFormer-HR+STLT(baseline)	Y	44.6	67.4	58.8	23.3	38.5	34.1	35.1	**59.7**	**49.6**
MFormer-HR+STLT+OL(ours)	Y	**46.3**	**68.7**	**59.4**	**25.7**	**39.9**	**35.3**	**35.4**	**59.7**	48.3

Table 4. Action Classification results on Epic-Kitchens. Our model achieves the best results compared to other methods using the same backbone. MFormer uses 224x224 resolution input and MFormer-HR uses 336x336 resolution input.

Backbone	Method	Boxes	SG	Aux loss	# Frames	Action CLS.	Predicate Pred.	
						mAP	R@10	R@20
I3D [5]	Avgpool	N	N	-	32	33.5	-	-
MFormer [49]	CLS token	N	N	-	16	36.5	76.4	82.6
STLT [5]	CLS token	Y	N	-	16	56.7	79.0	84.1
STLT [5]	CLS token	Y	N	-	32	60.0	-	-
I3D+STLT [5]	CACNF [5]	Y	Y	-	32	61.6	-	-
MFormer+STLT	CACNF [5]	Y	N	-	16	64.2	-	-
R101-I3D-NL [96]	SGFB [26]	Y	Y	-	32+	60.3	-	-
MFormer+STLT	LCF(baseline)	Y	N	-	16	66.0	78.3	83.5
MFormer+STLT	OL(ours)	Y	N	N	16	64.9	78.9	83.8
MFormer+STLT	OL(ours)	Y	N	Y	16	**66.6**	**80.9**	**85.4**

Table 5. Action recognition and human-object predicate prediction results on Action Genome. In action classification, our model outperforms others with the same frame and boxes input, and even SGFB with scene graph (SG) input. When linear-probing the output features for predicate prediction, our Object Learner fuses the visual and trajectory streams in an efficient way and is 2.6% higher than baseline LCF in recall@10. We also show the object-centric representations learned with the auxiliary loss is better than the ones learned without the auxiliary loss in both tasks.

Transfer to scene graph predicate prediction. We transfer the trained model on action classification to scene graph predicate prediction by linear probing. In this task, the model has to predict the predicate (relationship) between human and object when the bounding boxes and categories are known. Given the object id, we concat one-hot object id vectors with the classification vector from the frozen models, and train a linear classifier to predict the predicate. As shown in Table 5, the result from object summaries trained with the auxiliary loss is 2.6% higher than linear probing the concatenated CLS tokens (LCF) from two backbones, and 2.0% higher than the one trained without auxiliary loss.

6 Conclusion

We set out to evaluate whether objectness in video representations can aid visual task transfer. To this end, we have developed an object-centric video model which fuses the visual stream with object trajectories (bounding-boxes) in a novel transformer based architecture. We indeed find that the object-centric representations learned by our model are more transferrable to novel tasks and settings in video recognition using a simple linear probe, i.e., they outperform both prior object-agnostic and object-centric representations on a comprehensive suite of transfer tasks. This work uses a very coarse geometric representation of objects, i.e., bounding-boxes, for inducing object awareness in visual representations. In the future more spatially precise/physically-grounded representations, e.g., segmentation masks or 3D shape, could further enhance the transferability.

Acknowledgements. This research is funded by a Google-DeepMind Graduate Scholarship, a Royal Society Research Professorship, and EPSRC Programme Grant VisualAI EP/T028572/1.

References

1. Spelke, E.S., Breinlinger, K., Macomber, J., Jacobson, K.: Origins of knowledge. Psychological review **99** (1992) 1
2. Tenenbaum, J.B., Kemp, C., Griffiths, T.L., Goodman, N.D.: How to grow a mind: Statistics, structure, and abstraction. Science (2011) 1
3. Grill-Spector, K., Kanwisher, N.: Visual recognition: As soon as you know it is there, you know what it is. Psychological Science **16** (2005) 152–160 1
4. Herzig, R., Ben-Avraham, E., Mangalam, K., Bar, A., Chechik, G., Rohrbach, A., Darrell, T., Globerson, A.: Object-region video transformers. arXiv preprint arXiv:2110.06915 (2021) 1, 2, 3, 8, 10, 12, 13
5. Radevski, G., Moens, M.F., Tuytelaars, T.: Revisiting spatio-temporal layouts for compositional action recognition. In: Proc. BMVC. (2021) 1, 2, 3, 4, 6, 8, 10, 11, 12, 13, 14
6. Sun, C., Shrivastava, A., Vondrick, C., Murphy, K., Sukthankar, R., Schmid, C.: Actor-centric relation network. In: Proc. ECCV. (2018) 1, 3, 4
7. Zhang, Y., Tokmakov, P., Hebert, M., Schmid, C.: A structured model for action detection. In: Proc. CVPR. (2019) 1, 3
8. Wang, X., Gupta, A.: Videos as space-time region graphs. In: Proc. ECCV. (2018) 1, 3, 4, 8
9. Chen, Y., Rohrbach, M., Yan, Z., Shuicheng, Y., Feng, J., Kalantidis, Y.: Graph-based global reasoning networks. In: Proc. CVPR. (2019) 1
10. Locatello, F., Weissenborn, D., Unterthiner, T., Mahendran, A., Heigold, G., Uszkoreit, J., Dosovitskiy, A., Kipf, T.: Object-centric learning with slot attention. NeurIPS (2020) 1, 4
11. Battaglia, P., Pascanu, R., Lai, M., Jimenez Rezende, D., et al.: Interaction networks for learning about objects, relations and physics. NeurIPS (2016) 1, 4
12. Kulkarni, T.D., Gupta, A., Ionescu, C., Borgeaud, S., Reynolds, M., Zisserman, A., Mnih, V.: Unsupervised learning of object keypoints for perception and control. NeurIPS (2019) 1, 5
13. Dubey, R., Agrawal, P., Pathak, D., Griffiths, T.L., Efros, A.A.: Investigating human priors for playing video games. In: Proc. ICML. (2018) 2
14. Gopnik, A., Meltzoff, A.N., Kuhl, P.K.: The scientist in the crib: What early learning tells us about the mind. William Morrow Paperbacks (2000) 2
15. Smith, L.B., Jayaraman, S., Clerkin, E., Yu, C.: The developing infant creates a curriculum for statistical learning. Trends in cognitive sciences (2018) 2
16. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: Proc. CVPR. (2017) 2, 12
17. Lin, J., Gan, C., Han, S.: Tsm: Temporal shift module for efficient video understanding. In: Proc. ICCV. (2019) 2
18. Xie, S., Sun, C., Huang, J., Tu, Z., Murphy, K.: Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In: Proc. ECCV. (2018) 2
19. Yang, C., Xu, Y., Shi, J., Dai, B., Zhou, B.: Temporal pyramid network for action recognition. In: Proc. CVPR. (2020) 2
20. Materzynska, J., Xiao, T., Herzig, R., Xu, H., Wang, X., Darrell, T.: Something-else: Compositional action recognition with spatial-temporal interaction networks. In: Proc. CVPR. (2020) 2, 3, 4, 8, 9, 10, 11, 12
21. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. NeurIPS (2017) 2, 6

22. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: Proc. ECCV. (2020) 2, 4
23. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European conference on computer vision, Springer (2020) 213–229 2
24. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. NeurIPS (2014) 3, 4
25. Damen, D., Doughty, H., Farinella, G.M., Fidler, S., Furnari, A., Kazakos, E., Moltisanti, D., Munro, J., Perrett, T., Price, W., et al.: Scaling egocentric vision: The epic-kitchens dataset. In: Proc. ECCV. (2018) 3, 9
26. Ji, J., Krishna, R., Fei-Fei, L., Niebles, J.C.: Action genome: Actions as compositions of spatio-temporal scene graphs. In: Proc. CVPR. (2020) 3, 4, 9, 13, 14
27. Girdhar, R., Carreira, J., Doersch, C., Zisserman, A.: Video action transformer network. In: Proc. CVPR. (2019) 3, 4
28. Gao, C., Xu, J., Zou, Y., Huang, J.B.: Drg: Dual relation graph for human-object interaction detection. In: European Conference on Computer Vision. (2020) 3
29. Kato, K., Li, Y., Gupta, A.: Compositional learning for human object interaction. In: Proc. ECCV. (2018) 3
30. Xu, B., Wong, Y., Li, J., Zhao, Q., Kankanhalli, M.S.: Learning to detect human-object interactions with knowledge. In: Proc. CVPR. (2019) 3
31. Gkioxari, G., Girshick, R., Dollár, P., He, K.: Detecting and recognizing human-object interactions. In: Proc. CVPR. (2018) 3
32. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. NeurIPS (2015) 3
33. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Proc. ICCV. (2017) 3
34. Gupta, A., Davis, L.S.: Objects in action: An approach for combining action understanding and object perception. In: Proc. CVPR. (2007) 3, 4
35. Baradel, F., Neverova, N., Wolf, C., Mille, J., Mori, G.: Object level visual reasoning in videos. In: Proc. ECCV. (2018) 3, 4
36. Arnab, A., Sun, C., Schmid, C.: Unified graph structured models for video understanding. In: Proc. ICCV. (2021) 3, 4
37. Yan, R., Xie, L., Shu, X., Tang, J.: Interactive fusion of multi-level features for compositional activity recognition. arXiv preprint arXiv:2012.05689 (2020) 3, 4, 12
38. Ma, C.Y., Kadav, A., Melvin, I., Kira, Z., AlRegib, G., Graf, H.P.: Attend and interact: Higher-order object interactions for video understanding. In: Proc. CVPR. (2018) 3
39. Kolesnikov, A., Dosovitskiy, A., Weissenborn, D., Heigold, G., Uszkoreit, J., Beyer, L., Minderer, M., Dehghani, M., Houlsby, N., Gelly, S., Unterthiner, T., Zhai, X.: An image is worth 16x16 words: Transformers for image recognition at scale. In: Proc. ICLR. (2021) 4
40. Feichtenhofer, C., Fan, H., Malik, J., He, K.: Slowfast networks for video recognition. In: Proc. ICCV. (2019) 4, 12, 13
41. Feichtenhofer, C., Pinz, A., Zisserman, A.: Convolutional two-stream network fusion for video action recognition. In: Proc. CVPR. (2016) 4
42. Arandjelovic, R., Zisserman, A.: Objects that sound. In: Proc. ECCV. (2018) 4
43. Owens, A., Efros, A.A.: Audio-visual scene analysis with self-supervised multisensory features. In: Proc. ECCV. (2018) 4

44. Aytar, Y., Vondrick, C., Torralba, A.: See, hear, and read: Deep aligned representations. arXiv preprint arXiv:1706.00932 (2017) 4
45. Frome, A., Corrado, G.S., Shlens, J., Bengio, S., Dean, J., Ranzato, M., Mikolov, T.: Devise: A deep visual-semantic embedding model. NeurIPS (2013) 4
46. Weston, J., Bengio, S., Usunier, N.: Wsabie: Scaling up to large vocabulary image annotation. In: Proc. IJCAI. (2011) 4
47. Sun, C., Myers, A., Vondrick, C., Murphy, K., Schmid, C.: Videobert: A joint model for video and language representation learning. In: Proc. ICCV. (2019) 4
48. Wang, X., Girshick, R., Gupta, A., He, K.: Non-local neural networks. In: Proc. CVPR. (2018) 4
49. Patrick, M., Campbell, D., Asano, Y., Misra, I., Metze, F., Feichtenhofer, C., Vedaldi, A., Henriques, J.F.: Keeping your eye on the ball: Trajectory attention in video transformers. NeurIPS (2021) 4, 5, 8, 12, 13, 14
50. Alayrac, J.B., Recasens, A., Schneider, R., Arandjelović, R., Ramapuram, J., De Fauw, J., Smaira, L., Dieleman, S., Zisserman, A.: Self-supervised multimodal versatile networks. NeurIPS (2020) 4
51. Nagrani, A., Albanie, S., Zisserman, A.: Learnable pins: Cross-modal embeddings for person identity. In: Proc. ECCV. (2018) 4
52. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: Proc. CVPR. (2009) 4
53. Everingham, M., Eslami, S., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes challenge: A retrospective. IJCV (2015) 4
54. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Proc. ECCV. (2014) 4
55. Kristan, M., Matas, J., Leonardis, A., Felsberg, M., Pflugfelder, R., Kämäräinen, J.K., Chang, H.J., Danelljan, M., Cehovin, L., Lukežič, A., et al.: The ninth visual object tracking vot2021 challenge results. In: Proc. ICCV. (2021) 4
56. Huang, L., Zhao, X., Huang, K.: Got-10k: A large high-diversity benchmark for generic object tracking in the wild. IEEE PAMI (2019) 4
57. Dave, A., Khurana, T., Tokmakov, P., Schmid, C., Ramanan, D.: Tao: A large-scale benchmark for tracking any object. In: Proc. ECCV. (2020) 4
58. Dendorfer, P., Osep, A., Milan, A., Schindler, K., Cremers, D., Reid, I., Roth, S., Leal-Taixé, L.: Motchallenge: A benchmark for single-camera multiple target tracking. IJCV (2021) 4
59. Perazzi, F., Pont-Tuset, J., McWilliams, B., Van Gool, L., Gross, M., Sorkine-Hornung, A.: A benchmark dataset and evaluation methodology for video object segmentation. In: Proc. CVPR. (2016) 4
60. Xu, N., Yang, L., Fan, Y., Yue, D., Liang, Y., Yang, J., Huang, T.: Youtube-vos: A large-scale video object segmentation benchmark. arXiv preprint arXiv:1809.03327 (2018) 4
61. Yang, L., Fan, Y., Xu, N.: Video instance segmentation. In: Proc. ICCV. (2019) 4
62. Wang, W., Feiszli, M., Wang, H., Tran, D.: Unidentified video objects: A benchmark for dense, open-world segmentation. In: Proc. ICCV. (2021) 4
63. Girdhar, R., Ramanan, D.: CATER: A diagnostic dataset for Compositional Actions and TEmporal Reasoning. In: ICLR. (2020) 4
64. Yi, K., Gan, C., Li, Y., Kohli, P., Wu, J., Torralba, A., Tenenbaum, J.B.: Clevrer: Collision events for video representation and reasoning. arXiv preprint arXiv:1910.01442 (2019) 4

65. Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.J., Shamma, D.A., et al.: Visual genome: Connecting language and vision using crowdsourced dense image annotations. IJCV (2017) 4
66. Krishna, R., Chami, I., Bernstein, M., Fei-Fei, L.: Referring relationships. In: Proc. CVPR. (2018) 4
67. Johnson, J., Karpathy, A., Fei-Fei, L.: Densecap: Fully convolutional localization networks for dense captioning. In: Proc. CVPR. (2016) 4
68. Saenko, K., Packer, B., Chen, C., Bandla, S., Lee, Y., Jia, Y., Niebles, J., Koller, D., Fei-Fei, L., Grauman, K., et al.: Mid-level features improve recognition of interactive activities. Technical report, Dept. of Elec. Engg. and Computer Sc., University of California, Berkeley (2012) 4
69. Xu, H., Yang, L., Sclaroff, S., Saenko, K., Darrell, T.: Spatio-temporal action detection with multi-object interaction. arXiv preprint arXiv:2004.00180 (2020) 4
70. Battaglia, P., Hamrick, J.B.C., Bapst, V., Sanchez, A., Zambaldi, V., Malinowski, M., Tacchetti, A., Raposo, D., Santoro, A., Faulkner, R., Gulcehre, C., Song, F., Ballard, A., Gilmer, J., Dahl, G.E., Vaswani, A., Allen, K., Nash, C., Langston, V.J., Dyer, C., Heess, N., Wierstra, D., Kohli, P., Botvinick, M., Vinyals, O., Li, Y., Pascanu, R.: Relational inductive biases, deep learning, and graph networks. arXiv preprint arXiv:1806.01261 (2018) 4
71. Ye, Y., Singh, M., Gupta, A., Tulsiani, S.: Compositional video prediction. In: Proc. ICCV. (2019) 4
72. Liang, J., Jiang, L., Niebles, J.C., Hauptmann, A.G., Fei-Fei, L.: Peeking into the future: Predicting future person activities and locations in videos. In: Proc. CVPR. (2019) 4
73. Wu, Y., Gao, R., Park, J., Chen, Q.: Future video synthesis with object motion prediction. In: Proc. CVPR. (2020) 4
74. Greff, K., Kaufman, R.L., Kabra, R., Watters, N., Burgess, C., Zoran, D., Matthey, L., Botvinick, M., Lerchner, A.: Multi-object representation learning with iterative variational inference. In: Proc. ICML. (2019) 4
75. Henderson, P., Lampert, C.H.: Unsupervised object-centric video generation and decomposition in 3d. NeurIPS (2020) 4
76. Yang, C., Lamdouar, H., Lu, E., Zisserman, A., Xie, W.: Self-supervised video object segmentation by motion grouping. In: Proc. ICCV. (2021) 4
77. Burgess, C.P., Matthey, L., Watters, N., Kabra, R., Higgins, I., Botvinick, M., Lerchner, A.: Monet: Unsupervised scene decomposition and representation. arXiv preprint arXiv:1901.11390 (2019) 4
78. Engelcke, M., Kosiorek, A.R., Jones, O.P., Posner, I.: Genesis: Generative scene inference and sampling with object-centric latent representations. arXiv preprint arXiv:1907.13052 (2019) 5
79. Johnson, J., Gupta, A., Fei-Fei, L.: Image generation from scene graphs. In: Proc. CVPR. (2018) 5
80. Herzig, R., Raboh, M., Chechik, G., Berant, J., Globerson, A.: Mapping images to scene graphs with permutation-invariant structured prediction. NeurIPS (2018) 5
81. Park, T., Liu, M.Y., Wang, T.C., Zhu, J.Y.: Gaugan: semantic image synthesis with spatially adaptive normalization. In: ACM SIGGRAPH 2019 Real-Time Live. (2019) 5
82. Singh, K.K., Ojha, U., Lee, Y.J.: Finegan: Unsupervised hierarchical disentanglement for fine-grained object generation and discovery. In: Proc. CVPR. (2019) 5

83. Yang, B., Zhang, Y., Xu, Y., Li, Y., Zhou, H., Bao, H., Zhang, G., Cui, Z.: Learning object-compositional neural radiance field for editable scene rendering. In: Proc. ICCV. (2021) 5
84. Herzig, R., Bar, A., Xu, H., Chechik, G., Darrell, T., Globerson, A.: Learning canonical representations for scene graph to image generation. In: Proc. ECCV. (2020) 5
85. Ye, Y., Gandhi, D., Gupta, A., Tulsiani, S.: Object-centric forward modeling for model predictive control. In: Proc. CoRL. (2020) 5
86. Devin, C., Abbeel, P., Darrell, T., Levine, S.: Deep object-centric representations for generalizable robot learning. In: Proc. Intl. Conf. on Robotics and Automation. (2018) 5
87. Bapst, V., Sanchez-Gonzalez, A., Doersch, C., Stachenfeld, K., Kohli, P., Battaglia, P., Hamrick, J.: Structured agents for physical construction. In: Proc. ICML. (2019) 5
88. Anand, A., Racah, E., Ozair, S., Bengio, Y., Côté, M.A., Hjelm, R.D.: Unsupervised state representation learning in atari. NeurIPS (2019) 5
89. Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018) 7
90. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: Proc. ICLR. (2019) 8
91. Goyal, R., Ebrahimi Kahou, S., Michalski, V., Materzynska, J., Westphal, S., Kim, H., Haenel, V., Fruend, I., Yianilos, P., Mueller-Freitag, M., et al.: The" something something" video database for learning and evaluating visual common sense. In: Proc. ICCV. (2017) 9
92. Sigurdsson, G.A., Varol, G., Wang, X., Farhadi, A., Laptev, I., Gupta, A.: Hollywood in homes: Crowdsourcing data collection for activity understanding. In: Proc. ECCV. (2016) 9
93. Shan, D., Geng, J., Shu, M., Fouhey, D.F.: Understanding human hands in contact at internet scale. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. (2020) 9869–9878 9, 12
94. Zhang, Y., Sun, P., Jiang, Y., Yu, D., Yuan, Z., Luo, P., Liu, W., Wang, X.: Bytetrack: Multi-object tracking by associating every detection box. arXiv preprint arXiv:2110.06864 (2021) 9
95. Arnab, A., Dehghani, M., Heigold, G., Sun, C., Lučić, M., Schmid, C.: Vivit: A video vision transformer. In: Proc. ICCV. (2021) 13
96. Wu, C.Y., Feichtenhofer, C., Fan, H., He, K., Krahenbuhl, P., Girshick, R.: Long-term feature banks for detailed video understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. (2019) 284–293 14