PromptonomyViT: Multi-Task Prompt Learning Improves Video Transformers using Synthetic Scene Data

Anonymous ICCV submission

Paper ID 6823

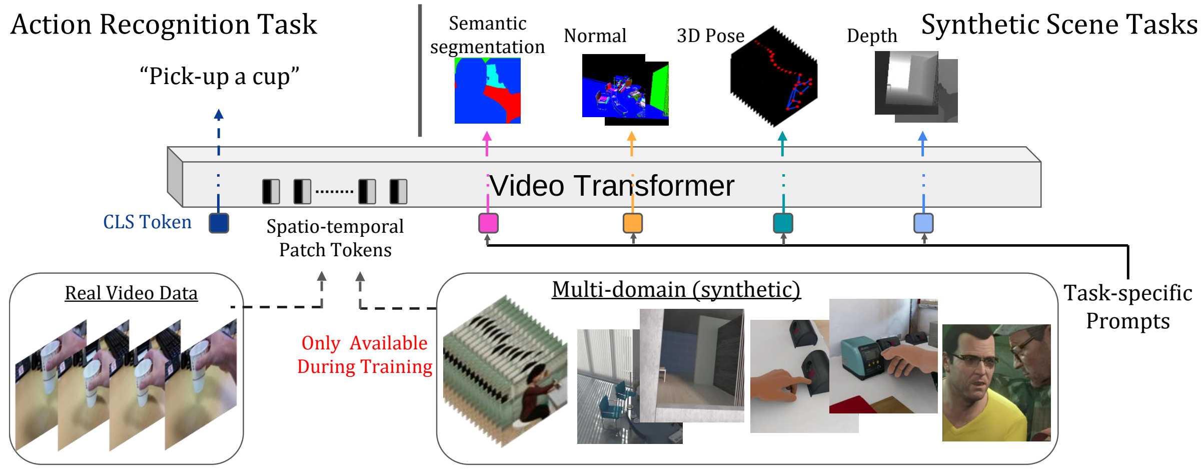


Figure 1: Our PromptonomyViT (PViT) adds a set of multiple prompts to a video transformer to capture inter-task structure and solve a downstream task. We consider the setting where automatically generated synthetic scene data for scene-level tasks (e.g., depth, semantic segmentation) is used for improving an action recognition model on real data. Our PViT model utilizes a multi-task prompt learning approach for video transformers, where a shared transformer backbone is enhanced with task-specific prompts (colored squares). The task prompts predict the synthetic labels for each task, and a CLS token (blue square) is used to predict the action recognition label. The use of task-specific prompts allows the model to benefit from task-related information.

Abstract

Action recognition models have achieved impressive results by incorporating scene-level annotations, such as objects, their relations, 3D structure, and more. However, obtaining annotations of scene structure for videos requires a significant amount of effort to gather and annotate, making these methods expensive to train. In contrast, synthetic datasets generated by graphics engines provide powerful alternatives for generating scene-level annotations across multiple tasks. In this work, we propose an approach to leverage synthetic scene data for improving video understanding. We present a multi-task prompt learning approach for video transformers, where a shared video transformer backbone is enhanced by a small set of specialized parameters for each task. Specifically, we add a set of "task prompts", each corresponding to a different task, and let each prompt predict task-related annotations. This design allows the model to capture information shared among synthetic scene tasks as well as information shared between synthetic scene tasks and a real video downstream task throughout the entire network. We refer to this approach as "Promptonomy", since the prompts model task-related structure. We propose the PromptonomyViT model (PViT), a video transformer that incorporates various types of scene-level information from synthetic data using the "Promptonomy" approach. PViT shows strong performance improvements on multiple video understanding tasks and datasets.

1. Introduction

Video understanding is a key challenge for machine vision and artificial intelligence. It is intuitively clear that video models should benefit from incorporating spatio-temporal scene-level information including objects, their relations, sizes of instances, 3D structure of a scene, its layout, depth and more. Indeed, several recent studies have explored the use of scene-level information for a variety of video tasks, such as action recognition [21, 26, 73, 40], action detection [52, 122], 3D understanding [84, 18, 1], and structured representations for videos [3, 33, 34, 40, 41, 48, 106, 111]. However, collecting and annotating real large-scale video datasets [36, 55] requires an extensive amount of effort and a large budget. This is especially true for complex labels such as 3D structure and segmentation maps.

In the absence of real-world data, synthetic datasets generated by graphics engines [92, 29] provide a powerful alternative for automatically generating scene-level annotations. Graphics engines can be used to generate a large amount of various types of labeled examples of scene-level information. However, learning from synthetic data requires models that can capture those aspects of the synthetic data that are relevant for downstream tasks, and overcome domain gap issues. An additional challenge is how to benefit from multiple types of scene labels (e.g., depth, normal, segmentation maps, 3D joints positions, and more). In this work, we propose a novel approach that can utilize synthetic data of various sources with multiple types of scene annotations to enhance video understanding models.

Our approach employs Vision Transformers (ViT) [25], which have recently emerged as the leading model for many vision applications [2, 27, 13], including for video understanding [5, 40, 66, 112]. Our key insight is that ViT can be naturally extended to multiple synthetic sources through the use of prompt learning. The key idea of prompt learning methods is to augment the transformer input with a set of additional learnable parameters. The notion of prompt learning has been used successfully in NLP [61], and more recently in machine vision [124, 123]. Inspired by this, we present a prompt learning approach for video transformers, where a shared backbone is enhanced by a small set of specialized parameters for each task. More specifically, we add a set of "task prompts", each dedicated to a unique task. With this design, it is possible to capture information shared among synthetic tasks as well as information shared between synthetic tasks and a real video downstream task, even without applying any domain gap techniques.[1]

The "task prompts" construction can be viewed as implementing "streams of information", each stream representing a task. This facilitates incorporating information from other tasks into the downstream task, starting from early layers and propagating into the spatio-temporal representations throughout the network. We refer to our prompt-per-task approach as "Promptonomy" since the prompts are intended to manage multiple tasks and capture inter-task structure, and name our model PromptonomyViT (PViT).[2] See Figure 1 for an overview.

Recently, the general idea of prompt tuning has been adapted to vision models by VPT [50], suggesting better efficiency of large vision models. Our model differs from recent "prompt tuning" approaches in that we refine a full transformer model rather than optimize a limited set of prompt tokens. As a result, information is propagated from the "task tokens" to all other tokens, enabling interaction across the entire network between the synthetic tasks and the real video downstream task. Furthermore, our multi-task prompts are supervised by auxiliary tasks, and not the primary action recognition task.

To summarize, our main contributions are as follows: (i) we propose a new method for exploiting synthetically generated labels for several tasks to improve video understanding models; (ii) we propose the concept of special "multi-task prompts" to capture task-related information through task supervision, while also interacting with prompts of other tasks and the downstream video task; (iii) we demonstrate improved performance on five tasks and five datasets on video understanding benchmarks: compositional and few-shot action recognition on SomethingElse, spatio-temporal action detection on AVA, standard action recognition on Something-Something V2, Diving48, and PNR Temporal Localization task on Ego4D, highlighting the effectiveness of the proposed approach.

2. Related Work

Prompt Tuning. Natural language prompting is a method of reformatting NLP tasks as natural language responses to natural language input. Recently, the concept of prompt tuning for efficient fine-tuning of language models was introduced by [61]. Several recent works [4, 90, 104], have explored prompt tuning in the context of multi-task learning in natural language processing. ATTEMPT [4] suggested a soft prompt tuning approach for parameter efficient multi-task knowledge sharing, UNIFIED PROMPT [90] suggested to use multi-task text prompting for zero-shot tasks, and the authors in [104] suggested the soft prompt tuning method for efficient fine-tuning. Additional recent works [50, 109, 108] suggested exploring the usage of prompt tuning in vision transformers. Specifically, VPT [50] uses prompt tuning to efficiently fine tune vision transformers, while others [109, 108] use prompts for continual learning. As opposed to these works, our focus is on the addition of multiple prompts that incorporate various types of scene-level information learned from synthetic data, which will lead to better video understanding. Last, we note that, since our focus is not on efficiency, the entire model is fine-tuned without freezing any parameters.

[1] Such techniques may improve performance further, but are orthogonal to our approach.

[2] The name also refers to the classic work on Taskonomy [119], which studied the structure and management of multiple tasks in images.

Learning from Synthetic Data. In the field of computer vision, synthetic data has been widely used as an alternative to real-world training data to solve various problems [29, 78, 83, 89, 24, 92, 102]. Many works attempted to generate synthetic data that mimics real data for image classification [29, 78], semantic segmentation [89, 107], action recognition [24, 102], object detection [82, 83], representation learning [110, 79], and more [92, 113]. Instead, our approach focuses on learning multiple tasks simultaneously from several synthetic domains and then transferring knowledge into the real world task by developing a multi-task prompting model and training scheme.

Multi-task Learning from Synthetic Data. The multi-task setting refers to the ability to learn multiple tasks simultaneously in which all model parameters are subject to a shared influence [15, 12, 22, 56, 32, 31, 96, 95, 71, 72, 117, 125]. Many recent works employ multi-task learning in CNNs [74, 101] and Transformers [116, 11, 44] to exploit the potential advantages of fast training, stronger results, and fewer parameters. MTFormer [116] is a transformer-based architecture, where multiple tasks share the same transformer encoder and decoder but has multiple modules layered on top of that for each task. MuIT [11] is a transformer-based encoder-decoder model with shared attention to learn task inter-dependencies, and UniT [44] jointly learns multiple tasks across different domains, from object detection to vision-and-language reasoning and natural language understanding. In contrast to these works, our work is a form of prompt-driven auxiliary task learning which uses synthetic scene-level annotations to train video transformers for improving action recognition on real video data.

Scene Understanding Models. Recently, scene understanding models that use scene-level annotations have been successfully applied to a wide range of computer vision applications: panoptic segmentation[85, 20], video relation understanding [67, 91, 97], vision and language [19, 63, 64, 98], relational reasoning [8, 9, 42, 59, 47, 86, 115, 118], human-object interactions [30, 54, 114], action recognition [3, 33, 34, 40, 41, 48, 80, 94, 106, 111, 121], and even image & video generation [7, 39, 51]. In our work, we demonstrate how video transformers can utilize shared representations from a variety of multiple different synthetic tasks to perform video downstream tasks.

Video Transformers. Vision Transformers [25, 100] recently proposed a new approach to image recognition by discarding the convolutional inductive bias entirely and instead employing self-attention operations. With the advent of ViT, and the fact that attention-based architectures are a natural choice for modeling long-range contextual relationships in video, a number of video transformer models, including TimeSformer [10], ViViT [2], Mformer (MF) [81], ORViT [40], MViT [27], MViTv2 [66] and Video Swin [75], form the latest era in action recognition. We choose to work with MViTv2, although our method can be used on top of any of these. Our work exploits the seamless ability of the transformer architecture to process multiple domains and to integrate the underlying structure among tasks for several downstream video-related tasks.

3. The PViT Model

Our PViT approach utilizes synthetic data of various domains with multiple types of scene annotations to enhance video understanding models. We consider the setting in which the main goal is to learn downstream video-understanding tasks, such as action recognition or action detection, while leveraging multiple synthetic scene-annotated datasets. The key idea of our work is that multi-task prompt learning can be used to incorporate synthetic scene tasks into the video model. This is achieved by adding a set of *task prompts*, each corresponding to a different task, and letting each prompt predict task-related annotations. Importantly, all prompts are part of the computation for any video, regardless of the underlying task, and thus enables sharing information among auxiliary tasks.

We begin by describing the video transformer architecture and the training setup (Section 3.1). We then introduce our Multi-task Prompts (Section 3.2) and the Training losses (Section 3.3). Our method is illustrated in Figure 2.

3.1. Preliminaries

Video Transformer Architecture. A typical Video Transformer model takes as input a video $X \in \mathbb{R}^{T\times 3\times H\times W}$, extracts N non-overlapping per-frame patches $x_i \in \mathbb{R}^{3\times h\times w}$ and projects them into a lower-dimension d (e.g., see [25]). Denote the transformer patches by Ex_i, which we refer to as "patch tokens". Then, spatio-temporal position embeddings $\text{PE} \in \mathbb{R}^{N\times d}$ are added for providing location and time location information, resulting in a new embedding: $z_i = Ex_i + \text{PE}_i$. This forms the sequence of input tokens to the video transformer:

$$z = [z_{CLS}, z_1, z_2, \cdots, z_N] \tag{1}$$

where z_{CLS} is a CLS token used for the downstream task. Next, a transformer is comprised of a stack the Multi-headed Attention (MHSA) blocks, which apply the self-attention operation over all patch tokens z (including the CLS token z_{CLS}) followed by a Feed-Forward Network (FFN), a layer normalization (LayerNorm [6]) step and a non-linear operation with residual connections [38].

Training Setup for Various Domains. In our approach, we aim to process batches of videos from various domains for n different tasks. A key desideratum in this context is to be able to input both videos of synthetic scene data across various domains for multiple tasks, as well as videos from

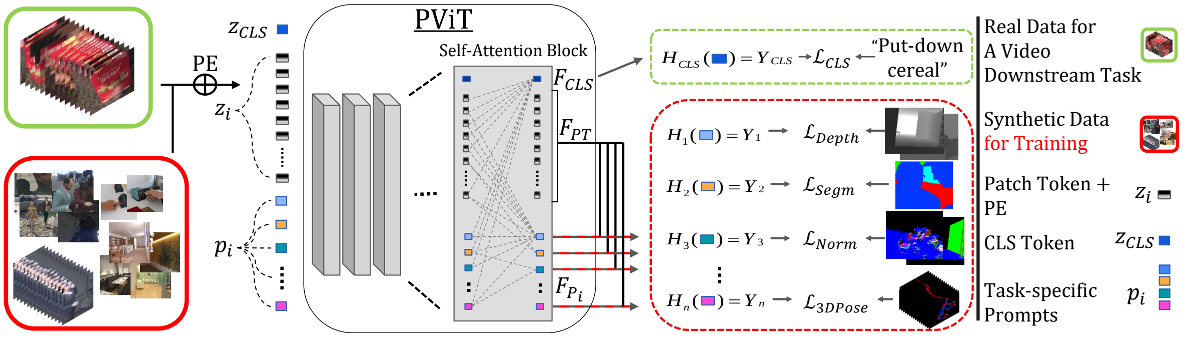

PE
z_{CLS}
z_i
p_i
PViT
Self-Attention Block
F_{CLS}
F_{PT}
F_{P_i}
H_{CLS} (■) = Y_{CLS} → $\mathcal{L}_{CLS}$ ← "Put-down cereal"
H_1(■) = Y_1 → $\mathcal{L}_{Depth}$
H_2(■) = Y_2 → $\mathcal{L}_{Segm}$
H_3(■) = Y_3 → $\mathcal{L}_{Norm}$
H_n(■) = Y_n → $\mathcal{L}_{3DPose}$
Real Data for A Video Downstream Task
Synthetic Data for Training
Patch Token + PE z_i
CLS Token z_{CLS}
Task-specific Prompts p_i

Figure 2: **PViT architecture**. We extend a transformer with a set of "task prompts", p_i, that are designed to capture information regarding each task, as well as capture the inter-task structure. The prompts are supervised by synthetic scene auxiliary tasks (depth, segmentation, normal, and 3D pose) available only during training, in order to enhance performance on a video task (predicting "put-down cereal"). Each task prompt in the attention block interacts with the patch tokens and CLS token, as well as other task prompts within the block.

the real domain into the same model. In contrast to standard training, where each sample contains a full set of annotations (e.g., depth, normal, etc.), in our case, only partial annotations are included. This is explained in greater detail in Section 3.3.

3.2. Multi-task Prompts

As mentioned earlier, our key observation is that multi-task prompt learning can be used to incorporate synthetic scene tasks into the video model. Towards this end, we add a set of "task prompts" designed to capture information regarding each task, as well as capture the inter-task structure. Specifically, we define a fixed number of n learned vectors $p_1, p_2, \cdots, p_n \in \mathbb{R}^{1\times d}$ for tasks $T_1, \cdots, T_n$. We refer to these vectors as the learned task prompts.

Let $P = \{p_1, p_2, \cdots, p_n\}$ be the set of task prompts. These prompts are concatenated to the patch tokens to obtain the following set of inputs to the transformer:

$$z = [z_{CLS}, z_1, z_2, ..., z_N, p_1, p_2, ..., p_n] \tag{2}$$

The transformer processes the input z, resulting in a new representation for each token z (i.e., the CLS token, the patch tokens, and the task-prompts). We denote $F_{CLS}(z)$ as the representation of the CLS token, and let $F_{P_i}(z)$ denote the representation of the i^{th} task-prompt. We also use $F_{PT}(z)$ as the final representation of all the patch tokens.

Next, these final output tokens are used for predicting labels. For the action recognition task, we simply predict using $F_{CLS}(z)$ and a prediction head $\hat{Y}_{CLS} = H_{CLS}(F_{CLS}(z))$. For the synthetic tasks, the task i has a prediction head $\hat{Y}_i = H_i(F_{P_i}(z), F_{PT}(z))$ that is used for predicting labels corresponding to this task. It uses the patch tokens only for cases where a dense prediction is required (e.g., segmentation maps, normal and depth estimation). The task heads H_i for localization tasks (e.g., boxes and 3D poses), are a simple FC layer, while for dense prediction tasks, we upsample patch token outputs from several layers and concatenate them with the corresponding task token to predict the task output map. Figure 3 also visualizes the "task prompts" learned by our model. For more info about the prediction heads see Section C in Supplementary.

3.3. Training and Inference

Our training data consists of labeled examples from n synthetic tasks, as well as the downstream task of action recognition. As mentioned above, we have $n+1$ predictions heads corresponding to those. During training, for each training video we add a loss corresponding to the labels provided for that video. For example, if the synthetic video X contains labels for task 2 (e.g., depth) and task 5 (e.g., normal), we take the output of prediction heads F_2 and F_5 and compare them to the ground-truth labels for these two tasks. We formally describe the task-specific losses below. We use $\hat{Y}$ to refer to predicted labels, and Y for ground-truth labels. **Losses**. For Depth Estimation, we first downsample the ground-truth depth map Y_{Depth} to a fixed scale of $\tilde{h} \times \tilde{w}$ map. Next, we predict a fixed scale map $\hat{Y}_{depth}$, and clip large values to focus on relatively closer objects. Finally, we use the MSE loss for computing the per-pixel depth error:

$$\mathcal{L}_{Depth} = \frac{1}{\tilde{h} \times \tilde{w}} \cdot \text{MSE}\left(\hat{Y}_{Depth}, Y_{Depth}\right) \tag{3}$$

For Normal Estimation, we predict the normal map $\hat{Y}_{Normal} \in \mathbb{R}^{h\times w\times 3}$ for every axis in world coordinates. We again down-sample the ground truth map Y_{Normal} and compute the MSE loss:

$$\mathcal{L}_{Normal} = \frac{1}{\tilde{h} \times \tilde{w}} \cdot \text{MSE}\left(\hat{Y}_{Normal}, Y_{Normal}\right) \tag{4}$$

For Semantic Segmentation, we use per-pixel multi-label classification to compute a map for different semantic instances in the scene. We downsample the ground-truth map Y_{Segm} and compute pixel-level cross-entropy loss followed by a Softmax function:

$$\mathcal{L}_{Segm} = \frac{1}{\tilde{h} \times \tilde{w}} \cdot \text{CE}\left(\hat{Y}_{Segm}, Y_{Segm}\right) \tag{5}$$

For 3D Pose estimation, we predict a tensor $\hat{Y}_{Pose3d} \in \mathbb{R}^{1\times 75}$ corresponding to a 25×3 of 3D joins in KinectV2 format [16]. Each training sample consists of a single individual. We define the loss for 3D Pose Estimation to be:

$$\mathcal{L}_{3DPose} = \frac{1}{75} \cdot \text{MSE}\left(\hat{Y}_{3DPose}, Y_{3DPose}\right) \tag{6}$$

For Bounding Box Prediction, we set a fixed number of O objects per training sample and use the $L1$ loss function to compute boxes predictions $\hat{Y}_{Boxes} \in \mathbb{R}^{O\times 4}$ and the corresponding ground-truth coordinates Y_{Boxes}:

$$\mathcal{L}_{Boxes} = \text{L}_1\left(\hat{Y}_{Boxes}, Y_{Boxes}\right) + \text{GIoU}\left(\hat{Y}_{Boxes}, Y_{Boxes}\right) \tag{7}$$

where the GIoU is used as in [87].

Last, for the video downstream task (denoted as DT), on which we evaluate our model, we consider the standard cross-entropy loss between the predicted logits $\hat{Y}_{CLS}$ and the true video labels Y_{CLS} as follows:

$$\mathcal{L}_{DT} = \text{CE}\left(\hat{Y}_{CLS}, Y_{CLS}\right) \tag{8}$$

The total loss is the sum of all of the losses described above. We note that only losses for which the samples have ground truth are added since the ground truth changes across instances, as our training does not use explicit correspondences between different input modalities. Each of the task terms in the loss is multiplied by a hyper-parameter (λ), and these were chosen such that all loss components have the same scale (see Supplementary). The total loss is the weighted combination of all terms:

$$\begin{aligned}\mathcal{L}_{Total} = \lambda_{DT}\mathcal{L}_{DT} + \lambda_{Depth}\mathcal{L}_{Depth} + \lambda_{Normal}\mathcal{L}_{Normal}\\ + \lambda_{Segm}\mathcal{L}_{Segm} + \lambda_{3DPose}\mathcal{L}_{Pose3d} + \lambda_{Boxes}\mathcal{L}_{Boxes}\end{aligned} \tag{9}$$

For simplicity, we omit the temporal dimension when predicting the losses above per frame.

Inference. For inference, PViT receives input from the real videos without requiring any additional synthetic data.

Finally, our method can be applied on top of a variety of video transformers (MViT [27], TimeSformer [10], Mformer [81]). For our experiments, we use the MViTv2 [66] model because it performs well empirically.

4. Experiments and Results

We begin by describing the datasets (Section 4.1), implementation details (Section 4.2), and baselines (Section 4.3). Next, we evaluate our approach on several benchmarks and tasks. Specifically, we consider the following tasks: Compositional Action Recognition (Section 4.4), Object State Change Classification & Localization (Section 4.5), Action Recognition (Section 4.6), and Spatio-Temporal Action Detection (Section 4.7).

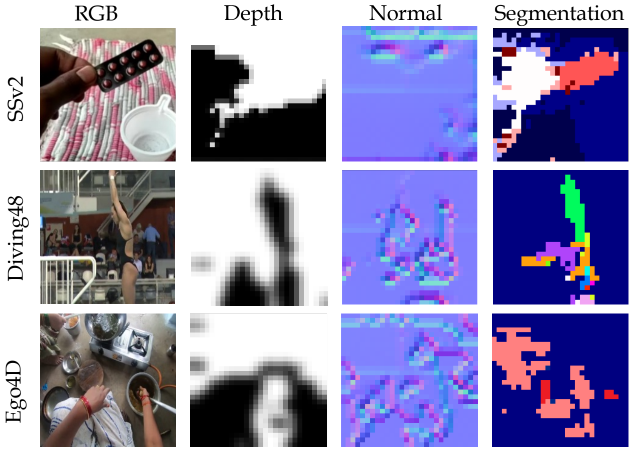


Figure 3: **"Task Prompts" Visualization**. Visualization of the output of the "task prompts" prediction heads on frames from the SSv2, Diving48, and Ego4D datasets. The model was trained with Something-Else as the action recognition dataset. Shown are prediction head outputs (i.e., H_i) for depth, normal, and semantic segmentation. It can be seen that the task prompts produce meaningful maps, despite not receiving such labels for real videos.

4.1. Datasets

We first describe the datasets used for the downstream video tasks, followed by the datasets used as auxiliary synthetic datasets including their annotations. We use the following video datasets: **(1) Something-Something v2 (SSv2)** [35] is a dataset containing 174 action categories of common human-object interactions. **(2) SomethingElse [77]** which exploits the compositional structure of SSv2, where a combination of a verb and a noun defines an action. We follow the official compositional split from [77], which assumes the set of noun-verb pairs available for training is disjoint from the set given at test time. **(3) Ego4D** [36] is a new large-scale dataset of more than 3,670 hours of video data, capturing the daily-life scenarios of more than 900 unique individuals from nine different countries around the world. **(4) Diving48** [65] contains 48 fine-grained categories of diving activities. **(5) Atomic Visual Actions (AVA)** [37] is a benchmark for human action detection, we report Mean Average Precision (mAP) on AVA-V2.2. For "auxiliary" synthetic datasets, we use **(1) SURREACT** [103], a novel synthetic data generation method based on real human motion from real datasets. The method renders 3D SMPL [76] sequences with randomized cloth textures, lighting, and body shapes from 3D skeleton joints extracted by Kinect V2 [55] from the two following datasets: **(i) NTU RGB+D** [93] is a large-scale multi-view video dataset of RGB-D human actions with 56,880 samples collected from 40 subjects, including depth maps and 3D skeleton joints. **(ii) UESTC RGB-D** [49] is also a multi-view action dataset that with 40 categories of aero-

Model	Compositional		Base		Few-Shot	
	Top-1	Top-5	Top-1	Top-5	5-Shot	10-Shot
I3D [14]	42.8	71.3	73.6	92.2	21.8	26.7
SlowFast [28]	45.2	73.4	76.1	93.4	22.4	29.2
TimeSformer [10]	44.2	76.8	79.5	95.6	24.6	33.8
Mformer [81]	60.2	85.8	82.8	96.2	28.9	33.8
MViTv2 [66]	63.3	87.5	83.7	96.8	32.7	40.2
MViTv2 MT	63.0	87.6	79.8	95.8	32.7	40.6
MViTv2 VPT	53.0	81.8	76.8	94.8	31.8	39.0
PViT (Ours)	**65.5** (+2.2)	**89.0** (+2.5)	**85.0** (+1.3)	**97.4** (+0.6)	**34.3** (+1.6)	**41.3** (+1.1)

Table 1: **Compositional and Few-Shot Action Recognition** on the SomethingElse dataset.

Model	Temporal Localization Error	PNR Classification Top-1
Bi-LSTM	0.790	65.3
BMN [70]	0.780	-
I3D ResNet-50 [14]	0.739	68.7
MViTv2 [66]	0.702	71.6
MViTv2 MT	0.640	73.6
MViTv2 VPT	0.791	64.2
PViT (Ours)	**0.637** (-0.065)	**74.8** (+3.2)

Table 2: Results for the **PNR Temporal Localization** task on the Ego4D dataset.

bic exercise along with depth maps and 3D skeleton joints. **(2) HyperSim** [88] is a photorealistic synthetic dataset for holistic indoor scene understanding. This dataset contains 77,400 HD images of 461 indoor scenes as well as ground truth depth and normal values for each pixel. **(3) Procedural Human Action Videos (PHAV)** [23] is a human action video dataset which relies on procedural generation and other computer graphics techniques of modern game engines. There are 39,982 actions in 35 categories, as well as ground truths for optical flow, semantic segmentation, and depth maps. **(4) KIST SynADL** [45] generated by the ElderSim engine, is a large-scale synthetic dataset of elders' activities. There are 462K RGB videos representing 55 action classes, along with 2D, 3D skeleton joints positions used as ground truth. **(5) Egocentric Human-Object Interactions (EHOI)** [60] consists of 20K synthetic image dataset of first-person view, annotated with segmentation masks, and hand-object interaction boxes of 19 categories.

4.2. Implementation Details

PViT is implemented in PyTorch, and the code will be released upon acceptance and is included in the supplementary. Our training recipes and code are based on the MViTv2-S, 16×4 model, and were taken from https://github.com/facebookresearch/mvit. We pretrain the PromptonomyViT model on the K400 [55] video dataset. Then, we finetune on the downstream video task (detailed in Section 4.1) with the synthetic datasets and the PromptonomyViT loss. In the training batch, there are 64 videos with the number of synthetic videos being at most $\times 3$ the number of real videos. For more implementation details, see Section C in Supplementary.

4.3. Baselines

In our experiments, we compare PViT to several models reported in previous work for the corresponding datasets. These include the following methods: BMN [70], *I3D* [14], *SlowFast* [28], as well as the state-of-the-art transformers – *SViT* [5], *TimeSformer* [10], *ViViT* [2], and *MViTv2* [66].

Additionally, we explore two alternative ViT-based baselines. First, we consider a model we call *MViTv2 multi-task (MViTv2 MT)*, and is perhaps the simplest application of ViT to our task. It augments the MViTv2 model with multiple prediction heads (one per synthetic task) operating on the CLS token, but *does not use additional task prompts*. The prediction heads have the same architecture as H_i used in PViT. We also consider a model we refer to as *MViTv2 VPT*, which is an implementation of the VPT [50] approach for action recognition. This is a simple prompt-based approach that utilizes the additional task prompts included in PViT but does not use additional synthetic data and *keeps the backbone frozen*. The advantage of MViTv2 VPT is training efficiency, as fewer parameters are used in training.

4.4. Compositional & Few-Shot Action Recognition

In several video datasets, an action is defined as the combination of a verb and a noun. Hence, one of the challenges is to identify combinations of words that were not seen during training. This "compositional" setting was explored in the "SomethingElse" dataset [77], where verb-noun combinations in the test data do not occur in the training data. We also evaluate the few-shot compositional action recognition task in [77] (See Section C.2 in supplementary).

Table 1 reports the results for these two tasks. PViT outperforms MViTv2 baseline for both the *Compositional* and *Few-shot* tasks by 2.2% for the compositional task, and by 1.6%, 1.1% for the 5-shot, and 10-shot tasks, respectively.

Furthermore, PViT outperforms MViTv2 MT, suggesting that the design of our task prompts approach is beneficial for learning from synthetic data. Last, it can be seen that MViT VPT performance is adversely affected, resulting in 53%. We note that this might be expected as the number of training parameters is only 0.3% of the MViTv2 baseline. Based on this result (common to all tables 1-3), it can be concluded that the VPT alternative may not the optimal approach for improving video transformers.

4.5. Object State Change Tasks

Human activity relies heavily on hands and objects. Two tasks studying hand-object interaction have recently been introduced to the Ego4D [36] dataset. The first is temporal localization, which involves finding key frames that indicate a change in object state within a video clip. The second is the classification of object state changes, which indicates whether an object state has changed or not.

(a) **Something–Something V2**

Model	Pretrain	Top-1	Top-5
SlowFast [28], R101	K400	63.1	87.6
ViViT-L [2]	IN+K400	65.4	89.8
MViTv1 [27]	K400	64.7	89.2
MViTv2 [66]	K400	68.2	91.4
MViTv2 MT	K400	68.4	91.4
MViTv2 VPT	K400	61.5	87.5
PViT (Ours)	K400	**69.4** (+1.2)	**91.6** (+0.2)

(b) **Diving48**

Model	Pretrain	Frames	Top-1
SlowFast [28], R101	K400	16	77.6
TimeSformer [10]	IN	16	74.9
MViTv2 [66]	K400	16	73.1
SViT [5]	K400	16	79.8
MViTv2 MT	K400	16	82.2
MViTv2 VPT	K400	16	69.8
PViT (Ours)	K400	16	**85.8** (+6.0)

(c) **AVA-V2.2**

Model	Pretrain	mAP
SlowFast [28], R50	K400	22.7
SlowFast [28], R101	K400	23.8
MViTv1 [27]	K400	25.5
MViTv2 [66]	K400	26.8
MViTv2 MT	K400	26.3
MViTv2 VPT	K400	19.0
PViT (Ours)	K400	**28.4** (+1.6)

Table 3: **Results on SSv2, Diving48, and AVA-V2.2 datasets.** We report (a) Top-1 and top-5 accuracy on SSv2. (b) Top-1 on Diving48. (c) mAP metric on AVA. IN refers to ImageNet-21K. For additional comparisons, see Section A.1 in supplementary.

Table 2 reports results on the above two tasks in Ego4D. We observe that PViT performs better than MViTv2 by 3.2%/-0.065 on the classification/localization tasks. As in Section 4.4, it can be seen that PViT consistently outperforms MViTv2 MT and MViTv2 VPT baselines. Overall, these results indicate that PViT successfully leverages scene data, even for another downstream video task.

4.6. Action Recognition

Tables 3a and 3b report results for the standard action recognition task on the SSv2 and Diving48 datasets. It can be seen that in Diving48, our method improves over the MViTv2 baseline by 6.0%, outperforming the other methods. We hypothesize that this relatively high gain is due to (i) the large availability of synthetic pose annotations (which is likely to help in human actions in the Diving dataset; See Figure 4d). (ii) Since Diving is a small dataset, the introduction of additional synthetic supervision results in a larger effect. Finally, PViT achieves a 1.2%, improvement in SSv2, indicating that PViT can improve on large datasets (180K videos). Last, PViT consistently outperforms MViTv2 MT and MViTv2 VPT baselines, as above.

4.7. Spatio-temporal Action Detection

Gu et al. [37] describes the action detection task on AVA as a two-stage prediction procedure. As a first step, boxes are detected using an off-the-shelf person detector, followed by a prediction of the action of each detected box. For fair comparisons, the person boxes are kept identical across approaches, and the final result is measured by the Mean Average Precision (MAP) metric.

Table 3c reports results for spatio-temporal action detection on the AVA dataset. We observe that PViT improves the MViTv2 baseline by 1.6%, thereby demonstrating the ability to leverage "task prompts" to detect and localize human actions. In addition, PViT consistently outperforms MViTv2 MT and MViTv2 VPT baselines, as above.

4.8. Ablations

We perform a comprehensive ablation study on the "SomethingElse" [77] dataset to measure the contribution of the different PViT components (See Table 4). For more ablations, see Section A in supplementary.

The Role of Prompts and Tuning. PViT contains two main concepts: (i) the addition of multiple task-specific prompts dedicated to unique tasks. (ii) training these prompt representations to predict task-related labels from synthetic data. We present results for different combinations of these two factors in Table 4a. First, to demonstrate the importance of having multiple prompts, one per task, we suggest the *MViTv2 one-prompt (OP)* variant. This variant is similar to PViT but uses *a single prompt* instead of n prompts for n auxiliary tasks. Since the number of prompts decreases, we compensate by increasing the dimension size. As shown in Table 4a, PViT outperforms the OP variant, suggesting that multiple prompts are important for integrating information across tasks.

Next, we consider the MViTv2 *neutral-prompts (NP)* variant, which is simply MViTv2 with additional prompts but without additional synthetic supervision (similar to the MViTv2 VPT, but with an unfrozen backbone). The purpose of this variant is to examine whether the model performance is due to the increased model capacity. This result (63.4) is similar to the baseline without synthetic data (MViTv2, 63.3), suggesting that the gain of PViT is due to the use of synthetic data. Last, the *PViT VPT* variant is a simply PViT with a frozen backbone. MViTv2 VPT differs from this variant since here, synthetic data is used for training. The result (53.9) emphasizes the importance of fine-tuning the backbone even when using synthetic data.

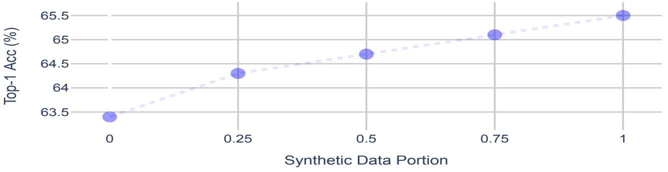
Model Capacity and Efficiency Analysis. To determine whether the performance improvement is a result of increasing parameter size, Table 4a compares the number of parameters, FLOPS, and inference runtime between the methods. The main difference between the models is due to the additional task prompts and the task heads since the latter contains the most overhead (only during training). In our setting, task prompts only add 20K parameters, while the task heads add 6.8M parameters. However, during test time, the heads are not used, and thus the parameter sizes are almost equal to the baseline (i.e., 38.2M), resulting in similar inference runtime and FLOPS as the baseline.

Effect of Synthetic Data Size. Here, we examine the impact of the synthetic data portion on performance. In Fig-

(a) **The Role of Prompts and Tuning**

Model	Top-1	Top-5	Synthetic Data	Train/Test Params ($\times 10^6$)	FLOPS ($\times 10^6$)	Runtime (ms)
MViTv2 [66]	63.3	87.5	✗	38.2/38.2	70.6	132.2
MViTv2 MT	62.7	87.6	✓	45.0/38.2	89.3	131.4
MViTv2 OP	63.5	88.0	✓	45.0/38.2	89.5	137.7
MViTv2 NP	63.4	87.8	✗	38.2/38.2	79.9	154.5
MViTv2 VPT	53.0	81.8	✗	0.13/38.2	82.3	154.5
PViT VPT	53.9	82.4	✓	7.2/38.2	93.9	143.7
PViT	**65.5**	**89.0**	✓	45.0/38.2	93.9	142.8

(b) **Effect of Synthetic Data Size**



(c) **Auxiliary Tasks Contribution**

Datasets	Depth	Segm.	Normal	3D Poses	2D Boxes	Top-1	Top-5
-	✗	✗	✗	✗	✗	63.3	87.5
PHAV+HS+SURR	✓	✗	✗	✗	✗	64.8	88.7
SUR+EHOI	✗	✓	✗	✗	✗	65.0	88.7
HS	✗	✗	✓	✗	✗	63.9	88.2
SUR+ES	✗	✗	✗	✓	✗	64.1	88.4
EHOI	✗	✗	✗	✗	✓	64.7	88.6
best combination	✓	✓	✗	✗	✓	**65.5**	**89.0**
All	✓	✓	✓	✓	✓	65.1	88.8

(d) **Dataset-Task Agreement**

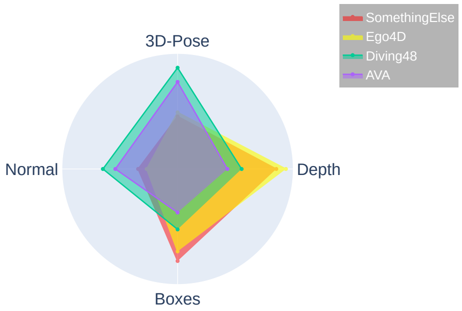


Table 4: **Ablations.** We show (a) The Role of Prompts and Tuning. (b) Effect of Synthetic Data Size. (c) Contribution of Auxiliary Tasks. (d) Dataset-Task Agreement. A polygon represents a real video dataset, and the closer a vertex is to the circle border, the greater the gain from applying that synthetic task. The gains are scaled for comparison

ure 4b, we plot the performance of PViT as a function of the synthetic data portion when the largest value is obtained using all synthetic data. The positive slope suggests that adding synthetic data consistently improves results, which is an advantage since synthetic data is abundant. We note that the synthetic data we used is the size of the real data.

Contribution from Auxiliary Tasks. To investigate the impact of each auxiliary task on performance, we examined in Table 4c how the auxiliary tasks contribute to performance individually, as well as the most effective combinations of auxiliary tasks. As can be seen, we find that performing PViT on auxiliary tasks individually does improve performance (see also *Dataset Task Agreement* below). However, using all tasks (last line) improves more than any individual task, and is also close to the optimal combination. This reinforces our strategy of simply training on all tasks.

Dataset-Task Agreement. We next aim to explore how different synthetic tasks help real datasets. Figure 4d illustrates the gain for real datasets when trained on individual auxiliary tasks[3]. It can be seen that the datasets are roughly clustered into two sets: (i) SomthingElse and Ego4D, which benefit more from Depth and Boxes. These datasets indeed contain hands interacting with *objects* within close range of the camera and therefore having clearly expressed *depth*. (ii) AVA and Diving48, which benefit more from Normals and Poses. These datasets generally consist of zoomed-out frames with mostly *full human bodies* in scenes containing *solid surfaces* (for example, pools, walls, etc.). For more details, see Section A.2 in the supplementary.

Domain Gap Between Synthetic and Real Data. In this work, we show that training PViT on synthetic data leads to improved performance on real data. However, as synthetic and real data come from different domains, it is not apriori clear why the former should aid the latter. We hypothesize that our synthetic tasks are mostly low-level (e.g., depth/normal maps, segm. masks), and for these, there may be a smaller gap between synthetic and real domains (See [17, 105]). To illustrate this, we use our learned task heads to predict labels on real data. Recall that these heads are learned only on synthetic data. Figure 3 shows results for this prediction, and it can be seen that the synthetic prompts predict well also on real data. This demonstrates that the synthetic tasks learned are also usable on real data.

5. Discussion and Limitations

Semantic understanding of videos is a key element of human visual perception, but its modeling is still challenging for machine vision. In this work, we propose a new method for exploiting various types of scene-level data to improve the performance of video understanding tasks. We present a multi-task prompt learning approach for video transformers, where a shared transformer backbone is enhanced with task-specific prompts. The use of task-specific prompts allows the model to benefit from task-related information, among different domains. We demonstrate improved performance on several video understanding benchmarks, highlighting the effectiveness of the proposed approach. However, the multi-task prompt learning method is not necessarily limited to synthetic scene data, and thus we leave to future research the challenge of extending the work to train the method on real data as well as improving other downstream tasks in addition to video understanding.

[3] The plot excludes segm. since it contributes equally to all datasets.

Supplementary Material

In this supplementary file, we provide additional information about our experimental results, qualitative examples, implementation details and datasets. Specifically, Section A provides more experiment results, Section B provides qualitative visualizations to illustrate our approach, Section C provides additional implementation details, and Section D provides additional datasets details.

A. Additional Experiment Results

We begin by presenting additional baseline results for all datasets and tasks in (Section A.1). Next, we present additional ablations (Section A.2) we performed in order to test the contribution of the different PViT components.

A.1. Baselines Comparison

Here, we evaluate several alternative ViT approaches (*MViTv2 MT* and *MViTv2 VPT*) to our task of using synthetic data towards improving action recognition models. Additionally, we report additional baselines that are comparable in compute and size to further compare to other approaches in (see Table 5), such as ORViT Mformer [40], UniFormer-S [62], SViT [5], VideoMAE [99], Video SWIN Transformer [75], STIN [77], and SAFCAR [57]. We can observe that our PViT approach improves upon MViTv2 and is competitive with other strong models. We note that even compared to VideoMAE, a recent self-supervised learning method, our results are similar in AVA (+1.3) and SSv2 (-0.3), although VideoMAE utilizes a larger backbone and more computing for training. Finally, PViT can be applied to any pretrained backbone, which gives it an advantage over other methods.

A.2. Additional Ablations

Next, we provide additional ablations that further illustrates the benefits of our PViT.

The importance of synthetic scene data. To examine how important the information provided by the synthetic scene data is, we test the PViT model, but provide it with “useless” synthetic label information. Specifically, we run an experiment in which the synthetic scene annotations are shuffled. As a result, the ground truth of the instance-level is shuffled for each synthetic scene task (e.g., for dense prediction tasks, the GT maps are shuffled). This ablation obtained 63.4%, similar to the baseline (63.3%). This is expected since wrong scene annotations are not likely to provide additional benefit beyond the baseline. Moreover, the model is capable of ignoring prompts if they are not required, so they should not have a negative impact beyond the baseline.

Prompts for real-world data. Even though real-world datasets are less reach in annotations compared to synthetic, PViT can still use them if available. To examine this, we added 2D hand-object boxes from SomethingElse as an additional auxiliary task along with its own prompt. This improved results by +1.2, suggesting that real data, if available, is beneficial. Clearly, the combination of synthetic and real data offers many promising and interesting directions, and we leave those to future work.

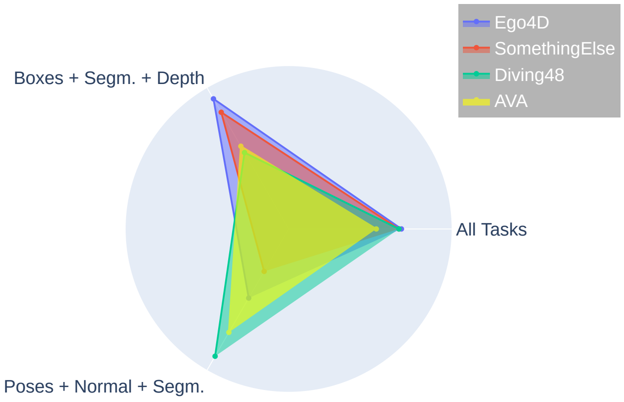


Figure 4: Dataset-Task Agreement. A polygon represents a real video dataset, and the closer a vertex is to the circle border, the greater the gain from using that synthetic task. The gains are scaled for comparison.

Comparison to a pretraining approach. Another approach for using synthetic datasets is first to pretrain on the synthetic data, and then finetune on the video-related task. Here, we demonstrate the effectiveness of our PViT approach as compared to this standard pretraining approach. To implement pretraining, we add prediction heads on top of MViTv2, and train them only on the synthetic datasets. Next, we remove these prediction heads and finetune the model by predicting using the CLS token. This approach achieved 61.9% compared to 63.3% for MViTv2 baseline and 65.5% for our PViT approach. This indicates that our PViT approach utilizes task information more effectively than a standard pretraining approach.

Number of task prompts. This ablation tests whether adding more prompts per task will improve the results compared to PViT, which uses one prompt per task. We add a total of 20 prompts to each task, which results in 65.4%, demonstrating that the addition of more prompts does not necessarily improve its performance. Clearly, there are many possible design choices, such as selecting a number of prompts per task, their dimension, integrating into different depths, etc., and we leave those to future work.

Dataset-Task Agreement. In Figure 4, we aim to explore how a different synthetic task combination helps real datasets. Since there are multiple possible subsets, we simplify and focus on only two subsets: S_1 = {Boxes, Segmentation, Depth} and S_2 = {Poses, Normal, Segmentation}.

(a) **Something–Something V2**

Model	Pretrain	Top-1	Top-5
SlowFast [28], R101	K400	63.1	87.6
MViTv1 [27]	K400	64.7	89.2
ViViT-L [2]	IN+K400	65.4	89.8
UniFormer-S [62]	IN+K600	67.9	92.1
ORViT Mformer [40]	K400	67.9	90.5
VideoMAE (ViT-S)	K400	66.8	90.3
MViTv2 [66]	K400	68.2	91.4
MViTv2 MT	K400	68.4	91.3
MViTv2 VPT	K400	61.5	87.5
PViT (Ours)	K400	**69.6** (+1.2)	**91.6** (+0.2)

(b) **Diving48**

Model	Pretrain	Frames	Top-1
SlowFast [28], R101	K400	16	77.6
TimeSformer [10]	IN	16	74.9
TimeSformer-L [10]	IN	96	81.0
SViT [5]	K400	16	79.8
MViTv2 [66]	K400	16	73.1
MViTv2 MT	K400	16	75.6
MViTv2 VPT	K400	16	69.8
PViT (Ours)	K400	16	**85.8** (+6.0)

(c) **AVA-V2.2**

Model	Pretrain	mAP
SlowFast [28], R50	K400	22.7
SlowFast [28], R101	K400	23.8
ORViT MViT-B [40]	K400	26.6
VideoMAE (ViT-S) [99]	K400	22.5
VideoMAE (ViT-B) [99]	K400	26.7
MViTv1 [27]	K400	25.5
MViTv2 [66]	K400	26.8
MViTv2 MT	K400	27.2
MViTv2 VPT	K400	19.0
PViT (Ours)	K400	**28.4** (+1.6)

(d) **SomethingElse**

Model	Compositional		Base		Few-Shot	
	Top-1	Top-5	Top-1	Top-5	5-Shot	10-Shot
I3D [14]	42.8	71.3	73.6	92.2	21.8	26.7
SlowFast [28]	45.2	73.4	76.1	93.4	22.4	29.2
TimeSformer [10]	44.2	76.8	79.5	95.6	24.6	33.8
STIN [77]	48.2	72.6	-	-	-	-
TSM [68]	52.3	78.0	-	-	-	-
Mformer [81]	60.2	85.8	82.8	96.2	28.9	33.8
SAFCAR [57]	60.7	84.2	-	-	-	-
MViTv2 [66]	63.3	87.5	83.7	96.8	32.7	40.2
MViTv2 MT	62.7	87.6	81.4	96.2	34.0	40.9
MViTv2 VPT	53.0	81.8	76.8	94.8	31.8	39.0
PViT (Ours)	**65.5** (+2.2)	**89.0** (+2.5)	**85.0** (+1.3)	**97.4** (+0.6)	**34.3** (+1.6)	**41.3** (+1.1)

(e) **Ego4D**

Model	Temporal Localization Error	PNR Classification Top-1
Bi-LSTM	0.790	65.3
BMN [70]	0.780	-
I3D ResNet-50 [14]	0.739	68.7
EgoVLP (TimeSformer) [69]	0.666	73.9
Video Swin Transformer [75]	0.660	69.5
MViTv2 [66]	0.702	71.6
MViTv2 MT	0.640	73.6
MViTv2 OP	0.652	73.7
PViT (Ours)	0.637 (-0.065)	**74.8** (+3.2)

Table 5: **Results on SSv2, Diving48, AVA-V2.2, SomethingElse, and Ego4D datasets.** We report top-1 and top-5 accuracy on SSv2 and SomethingElse. On AVA, we report the mAP metric. On Diving48, we report top-1. On Ego4D we report classification error. IN refers to ImageNet-21K.

The former relates to hand-object interaction (HOI), and the latter to human action (HA). The figure shows the accuracy for real datasets when trained on either S_1, S_2, or all five tasks. This confirms our original hypothesis from the main paper that the datasets are roughly clustered into two categories: (i) SomthingElse and Ego4D benefit more from the HOI set. These datasets indeed usually contain hands interacting with objects, often in first person and with a low field of view. (ii) AVA and Diving48 belong benefit more from the HA group. These datasets generally consist of zoomed-out frames with mostly full human bodies.

Contribution from Datasets and Tasks. In order to quantify the impact of each dataset and task, we conducted a comprehensive analysis in Table 6. At the top of the table we display the contribution of each synthetic dataset to the downstream task, and at the bottom we display the contribution of each synthetic task (namely, we use all existing annotations from across all of our synthetic datasets). We observe that EHOI achieves the highest gains. This is similar to our observation in the main paper that hand-object interaction videos (HOI) benefit more from bounding box supervision. For more details, see the Dataset-Task Agreement ablation (Figure 4d) in the main paper.

In the bottom portion of the table, we examined in the auxiliary tasks contribute to performance individually, as well as the most effective combinations of auxiliary tasks. As can be seen, we find that performing PViT on auxiliary tasks individually does improve performance (see also *Dataset Task Agreement* below). However, using all tasks (last line) improves more than any individual task, and is also close to the optimal combination. This reinforces our strategy of simply training on all tasks. For a visualization of the datasets, see Section B in supplementary.

B. Qualitative Visualizations

Figure 5 and Figure 3 in the main paper show visualizations of "task prompts" predictions on examples of real videos from SSv2, Diving48, Ego4D, and AVA. It can be seen that predictions are reasonable, despite the model not being trained on these labels for the real videos. For better illustration, we show in Figure 6 the different auxiliary

Dataset	Depth	Segm.	Normal	3D Poses	2D Boxes	Top-1	Top-5
-	✗	✗	✗	✗	✗	63.3	87.5
PHAV	✓	✓	✗	✗	✗	64.2	87.6
SUR	✓	✗	✗	✓	✗	63.9	88.4
ES	✗	✗	✗	✓	✗	63.9	88.1
HS	✓	✗	✓	✗	✗	64.1	87.4
EHOI	✗	✓	✗	✗	✓	65.0	88.5
PHAV+HS+SURR	✓	✗	✗	✗	✗	64.8	88.7
SUR+EHOI	✗	✓	✗	✗	✗	65.0	88.7
HS	✗	✗	✓	✗	✗	63.9	88.2
SUR+ES	✗	✗	✗	✓	✗	64.1	88.4
EHOI	✗	✗	✗	✗	✓	64.7	88.6
best combination	✓	✓	✗	✗	✓	**65.5**	**89.0**
All	✓	✓	✓	✓	✓	65.1	88.8

Table 6: **Compositional Action Recognition task on the SomethingElse dataset.** The contribution of every synthetic auxiliary dataset (top) and task (bottom).

synthetic datasets we used in the main paper, as described in Section 4.1 and further elaborated upon in Section D.1.

C. Additional Implementation Details

Our PViT model can be used on top of the most common video transformers (MViT [27], TimeSformer [10], Mformer [81], Video Swin [75]). For our experiments, we choose the MViTv2 [66] model because it performs well empirically. These are all implemented based on the MViTv2 [66] library (available at https://github.com/facebookresearch/mvit), and we implement PViT based on this repository. Furthermore, we set the λ parameters (see Equation 9) for the $\mathcal{L}_{Depth}$, $\mathcal{L}_{Normal}$, $\mathcal{L}_{Segm}$, $\mathcal{L}_{3DPose}$, $\mathcal{L}_{Boxes}$, and $\mathcal{L}_{DT}$ losses, to 0.5, 0.5, 0.1, 3.0, 0.1 and 1 respectively (across all datasets). We choose these lambda components such that all loss components have the same scale. We elaborate next on the additional implementation details for each dataset, including information about optimization, and training and inference.

Dense Prediction Heads. In order to preserve the spatio-temporal information in dense prediction tasks, we use patch tokens in addition to task the tokens. First, we upsample patch tokens from layers 2, 12, 15 (out of 16) using a 3D convolution layer, followed by Dropout and concatenation. We then concatenate them with relevant task tokens and forward them to an MLP for a final prediction.

C.1. Diving48

Dataset. Diving48 [65] contains 16K training and 3K testing videos spanning 48 fine-grained diving categories of diving activities. For all of these datasets, we use standard classification accuracy as our main performance metric.

Optimization details. We train using 16 frames with sample rate 4 and batch-size 128 (comprising 64 videos and 64 auxiliary synthetic datasets) on 8 RTX 3090 GPUs. We train our network for 10 epochs with Adam optimizer [58] with a momentum of $9e-1$ and Gamma $1e-1$. Following [66], we use $lr = 1.5e-4$ with half-period cosine decay.

Training details. We use crops of size 224 for the standard model and jitter scales between $256-320$. together with RandomFlip augmentation. Finally, we sample T frames from the start and end annotation times, following [120].

Inference details. We take 3 spatial crops per single clip to form predictions over a single video in testing, as in [10].

C.2. SomethingElse

Dataset. The SomethingElse dataset [77] contains 174 action categories with 54,919 training and 57,876 validation samples. The compositional [77] split in this dataset provides disjoint combinations of a verb (action) and noun (object) in the training and testing set, defining two disjoint groups of nouns $\{\mathcal{A}, \mathcal{B}\}$ and verbs $\{1, 2\}$. Given the splits of groups, they combine the training set as $1\mathcal{A}+2\mathcal{B}$, while the validation set is constructed by flipping the combination into $1\mathcal{B}+2\mathcal{A}$. In this way, different combinations of verbs and nouns are divided into training or testing splits.

Few Shot Compositional Action Recognition. As mentioned in Section 4.4, we also evaluate on the few-shot compositional action recognition task in [77]. For this setting, we use 88 *base* action categories and 86 *novel* action categories. We train on the base categories (113K/12K for training/validation) and fine-tune on few-shot samples from the novel categories (for 5-shot, 430/50K for training/validation; for 10-shot, 860/44K for training/validation). We use the same training recapie as in C.2.

Optimization details. We train using 16 frames with sample rate 4 and batch-size 128 (comprising 64 videos and 64 auxiliary synthetic datasets) on 8 RTX 3090 GPUs. We train our network for 100 epochs with Adam optimizer [58] with a momentum of $9e-1$ and Gamma $1e-1$. Following [66], we use $lr = 7e-5$ with half-period cosine decay.

Regularization details. We use weight decay of $1e-4$, and a dropout [43] of $5e-1$ before the final perdition.

Training details. We use standard crop size of 224, and we jitter scales from 256 to 320.

Inference details. We take 3 spatial crops per single clip to form predictions over a single video in testing.

C.3. Something-Something v2

Dataset. The SSv2 [77] is a ∼160K-video dataset contains 174 action categories of common human-object interactions. We follow the official splits from [35].

Optimization details. For the standard SSv2 [77] dataset, we train using 16 frames with sample rate 4 and batch-size 128 (comprising 64 videos and 64 auxiliary synthetic datasets) on 8 RTX 3090 GPUs. We train our network for

100 epochs with Adam optimizer [58] with a momentum of $9e-1$ and Gamma $1e-1$. Following [66], we use $lr = 7e-5$ with half-period cosine decay.
Regularization details. We use weight decay of $1e-4$, and a dropout [43] of $5e-1$ before the final classification.
Training details. We use a standard crop size of 224, and we jitter the scales from 256 to 320 along with RandomFlip.
Inference details. We take 3 spatial crops per single clip to form predictions over a single video in testing as in [66].

C.4. Ego4D

Dataset. Ego4D [36] is a new large-scale dataset of more than 3,670 hours of video data, capturing the daily-life scenarios of more than 900 unique individuals from nine different countries around the world. The videos contain audio, 3D meshes of the environment, eye gaze, stereo and/or synchronized videos from multiple egocentric cameras.
Metrics. In the Object State Change Temporal Localization task, the absolute error (in seconds) is used for evaluation. In the Object State Change Classification task, the top-1 accuracy is used for evaluation, following [36] protocol.
Optimization details. We train using 16 frames with sample rate 4 and batch-size 128 (comprising 64 videos and 64 auxiliary synthetic datasets) on 8 RTX 3090 GPUs. We train our network for 10 epochs with Adam optimizer [58] with a momentum of $9e-1$ and Gamma $1e-1$. Following [66], we use $lr = 1.5e-5$ with half-period cosine decay. Additionally, we used Automatic Mixed Precision, which is implemented by PyTorch.
Training details. We use a standard crop size of 224, and we jitter the scales from 256 to 320.
Inference details. We follow the official evaluation, both for the state change temporal localization and the state change classification tasks, available at https://github.com/EGO4D/hands-and-objects.

C.5. AVA-2.2

Dataset. AVA-2.2 (Atomic Visual Action) dataset [37] contains bounding box annotations for spatio-temporal localization of human actions. There are 211K training videos and 57K validation videos in the dataset. We report mean Average Precision (mAP) on 60 classes [37] on AVA v2.2 according to the standard evaluation protocol.
Architecture. SlowFast [28] and MViTv2 [66] use a detection architecture with a RoI Align head on top of the spatio-temporal features. We follow their implementation to allow a direct comparison, elaborating on the RoI Align head proposed in SlowFast [28]. First, we extract the feature maps from our PViT model by using the RoIAlign layer. Next, we take the 2D proposal at a frame into a 3D RoI by replicating it along the temporal axis, followed by a temporal global average pooling. Then, we max-pooled the RoI features and fed them to an MLP classifier for prediction.

Dataset	Available Annots.	#Training Samples ($\times 10^3$)	Real/Synt.
PHAV	D+S	39.9	Synt.
SURREACT	D+S+P3D	108.3	Synt.
ElderSim	P3D	48.8	Synt.
HyperSim	N+D	31.1	Synt.
EHOI	B+S	20.0	Synt.
SomethingElse	-	54.91	Real
SSv2	-	157.4	Real
AVA-2.2	-	193.3	Real
Ego4D	-	41.1	Real
Diving48	-	15.0	Real

Table 7: **Real and synthetic dataset details**. We show (a) Top: the auxiliary synthetic datasets, and (b) Bottom: downstream real datasets. The available annotations are depth maps (D), segmentation (S), 3D poses (P3D), normal maps (N) and boxes (B).

Optimization details. To allow a direct comparison, we used the same configuration as in MViTv2 [66]. We trained 16 frames with sample rate 4, depth of 16 layers and batch-size 32 (comprising 16 videos and 16 auxiliary synthetic datasets) on 8 RTX 3090 GPUs. We train our network for 30 epochs with an SGD optimizer. We use $lr = 0.03$ with a weight decay of $1e-8$ together with early-stopping and a half-period cosine schedule of learning rate decaying.
Training details. We use a standard crop size of 224 and we jitter the scales from 256 to 320. We use the same ground-truth boxes and proposals that overlap with ground-truth boxes by $IoU > 0.9$ as in [28].
Inference details. We perform inference on a single clip with 16 frames. For each sample, the evaluation frame is centered in frame 8. We take 1 spatial crop of 224 with 10 different randomly sampled clips to aggregate predictions over a single video in testing.

D. Additional Synthetic Datasets Details

Here we provide additional information about the "auxiliary synthetic datasets" (Section D.1), as well as the licenses and privacy policies for these datasets (Section D.2). Figure 6 shows examples of the synthetic videos we used to train on, while Table 7 presents the size of training samples across all synthetic and real datasets.

D.1. Auxiliary Synthetic Datasets

Synthetic datasets. There has been recent interest in learning video understanding from synthetic data, including several popular synthetic datasets that have been proposed to improve video understanding. More specifically, a novel approach to data generation has been proposed by SURREACT [103] and UESTC [49] for synthesizing humans for actions. KIST SynADL [45] is a large-scale synthetic dataset of elders' activities generated by the ElderSim engine [46]. The PHAV [23] dataset is a human action dataset that relies on the procedural generation of modern game engines. NTU RGB+D [93] and UESTC RGB-D [49] are large-scale synthetic datasets that was proposed in order to allow the training of large video models for video understanding. HyperSim [88] is a photo-realistic synthetic dataset for holistic indoor scene understanding. Egocentric Human-Object Interactions (EHOI) [60] explores hand-object interaction in an industrial environment involving different objects, e.g. power supply, electrical panels, sockets, and more. In spite of the fact that these datasets contain different dataset styles, our approach is able to enhance video understanding models by utilizing synthetic data from various sources with multiple types of scene annotations. Next, we provide more details for each dataset separately.

SURREACT [103]. The SURREACT dataset, which stands for Synthetic hUmans foR REal ACTions, renders video sequences from 3D skeleton joints by using a Skinned Multi-Person Linear Model (SMPL). The ground truth joints are extracted either by Kinect v2, or HMML [53]. SURREACT consists of **(1) NTU RGB+D**, which is a large-scale dataset for RGB-D human action recognition. It consists of 56,880 samples of 60 action classes collected from 40 subjects. The actions are generally categorized into three categories: 40 daily actions (e.g., drinking, eating, reading), nine health-related actions (e.g., sneezing, staggering, falling down), and 11 mutual actions (e.g., punching, kicking, hugging). These actions take place under 17 different scene conditions corresponding to 17 video sequences (i.e., S001–S017). The actions were captured using three cameras with different horizontal imaging viewpoints, namely, $-45°$,$0°$, and $+45°$ degrees. Last, multi-modality information is provided for action characterization, including depth maps, 3D skeleton joint position, RGB frames, and infrared sequences; and **(2) UESTC RGB-D**, which contains 40 categories of aerobic exercise. The authors utilized two KinectV2 cameras in 8 fixed directions and 1 round direction to capture these actions with the data modalities of RGB video, 3D skeleton and depth map sequences.

HyperSim [88]. The HyperSim dataset is a high-resolution dataset consisting of 77,400 images from 461 indoor scenes with detailed per-pixel labels and corresponding ground truth geometry. It contains material and lighting information for every scene as well as dense per-pixel semantic instance segmentation, as well as complete camera information for every image. HyperSim was originally designed to handle the challenging per-pixel annotation of real data.

KIST SynADL [45]. KIST SynADL is a synthetic dataset that focuses on elders' daily activities, which differ from other natural actions due to their high degree of variety. The activities of elders, such as *sitting down* or *washing face*, are more consistent psychically, shorter, and often rely on body position. Last, the dataset is generated using ElderSim [46], a synthetic action simulation platform aimed at generating synthetic data on elders' daily activities. Throughout the paper, we refer to ElderSim as KIST SynADL.

Procedural Human Action Videos (PHAV). The PHAV dataset is a diverse, realistic, and physically plausible dataset of human action videos. It contains a total of 39,982 videos, with more than 1,000 examples of each action in 35 categories across 7 different environments and 4 types of weather. The data is generated based on the existing motion-based real database CMU MOCAP database, for basic human animations. One of its key components is the use of Ragdoll physics to animate a human model while respecting basic physics properties such as connected joint limits, angular limits, weight, and strength. The videos are generated at 30fps and a resolution of 340x256.

Egocentric Human-Object Interactions (EHOI) [60]. EHOI is a synthetic image dataset that explores hand-object interaction in an industrial environment involving different objects, such as a power supply, electrical panels, sockets, etc. To create 3D models, several 3D scanners are applied, then using Blender, the authors generate the following: (1) photo-realistic RGB images; (2) depth maps; (3) semantic segmentation masks, objects, and hand-bounding boxes with contact states; and (4) distance between hands and objects in 3D space. The generated synthetic dataset contains a total of 20,000 images, 29,034 hands (of which 14,589 are involved in an interaction), 123,827 object instances (14,589 of which are active objects), and 19 object categories including portable industrial tools (e.g., screwdrivers, electrical boards) and instruments.

D.2. Licenses and Privacy

The license, PII, and consent details of each dataset are in the respective papers. In addition, we wish to emphasize that the datasets we use do not contain any harmful or offensive content, as many other papers in the field also use them. Thus, we do not anticipate a specific negative impact, but, as with any Machine Learning method, we recommend to exercise caution.

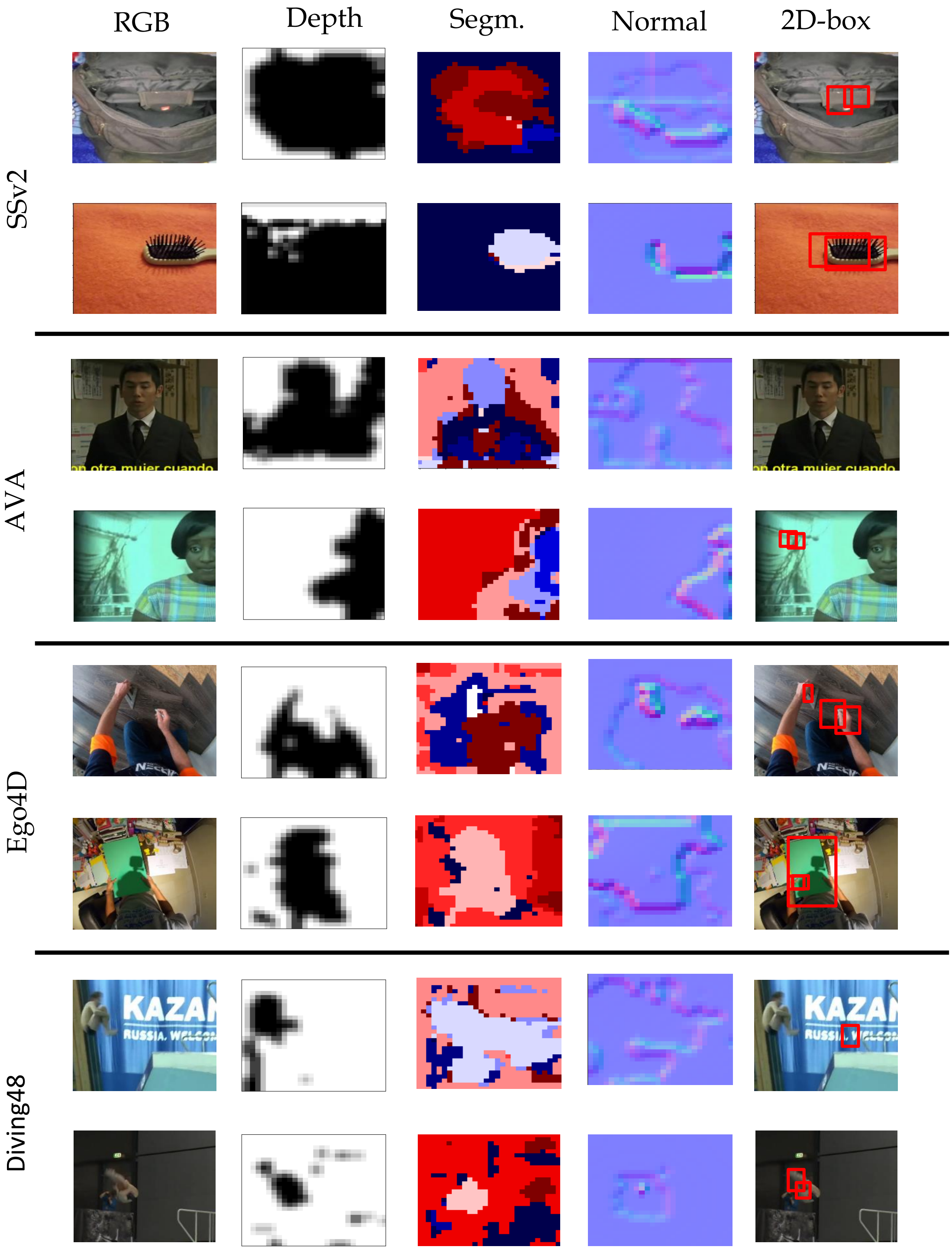


Figure 5: **Qualitative visualization of the "Task Prompts"**. Visualization of the output of the "task prompts" prediction heads on frames from the SSv2, Diving48, Ego4D, and AVA datasets. The model was trained on the SomethingElse dataset for action recognition. The predictions are the head outputs, H_i, for depth, normal, part-semantic segmentation and hand-object 2D boxes. It can be observed that the task prompts produce meaningful maps, despite not receiving labels for the real videos.

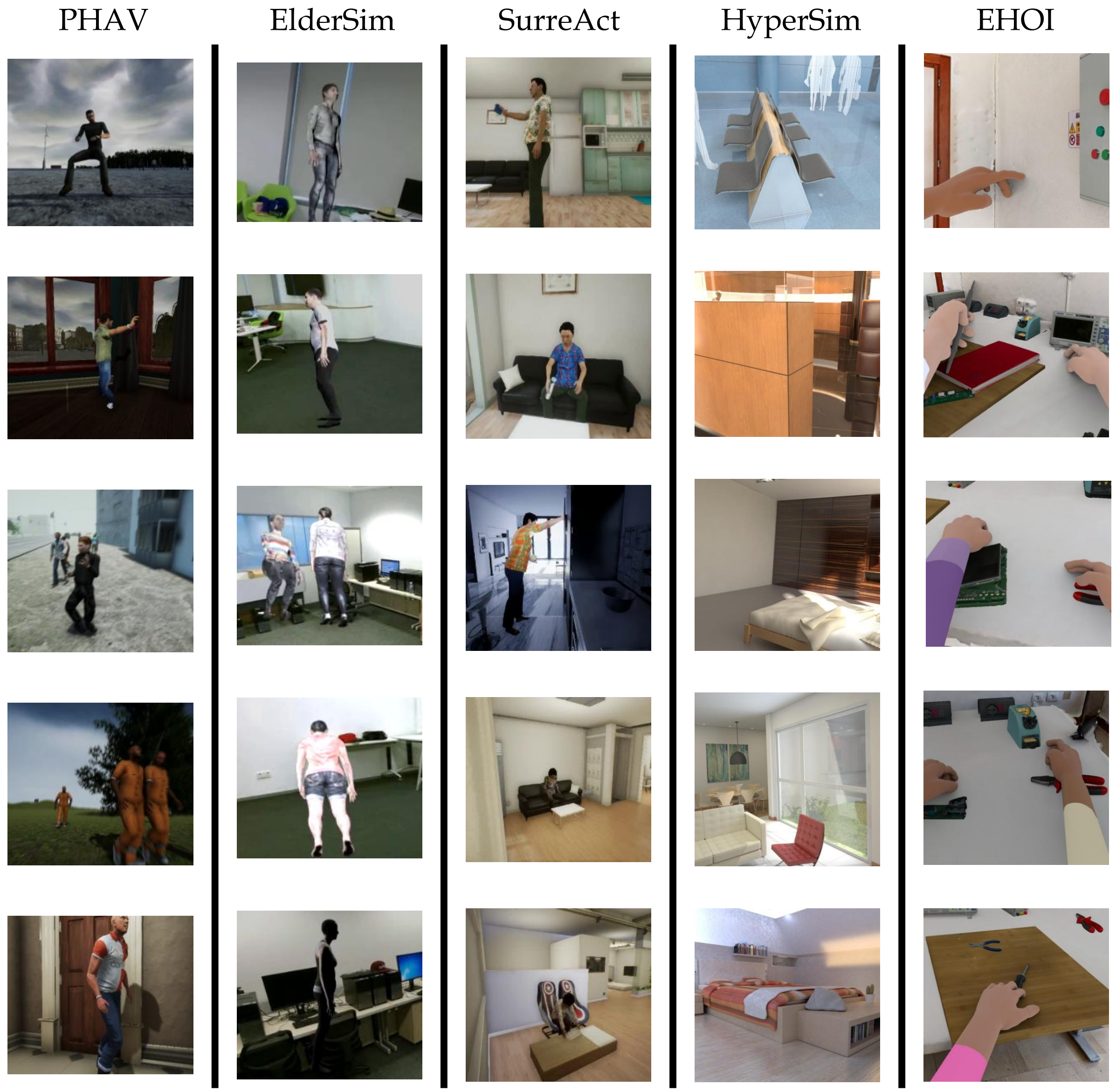


Figure 6: **Synthetic Datasets Visualization.** Our training datasets for PViT consist of several synthetic datasets that each emphasize different topics, including multi-views, static objects, hand-object interaction, and human motion activities.

References

[1] Iro Armeni, Sasha Sax, Amir Roshan Zamir, and Silvio Savarese. Joint 2d-3d-semantic data for indoor scene understanding. *ArXiv*, abs/1702.01105, 2017. 2

[2] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer, 2021. 2, 3, 6, 7

[3] Anurag Arnab, Chen Sun, and Cordelia Schmid. Unified graph structured models for video understanding. In *ICCV*, 2021. 2, 3

[4] Akari Asai, Mohammadreza Salehi, Matthew E. Peters, and Hannaneh Hajishirzi. Attentional mixtures of soft prompt tuning for parameter-efficient multi-task knowledge sharing. *ArXiv*, abs/2205.11961, 2022. 2

[5] Elad Ben Avraham, Roei Herzig, Karttikeya Mangalam, Amir Bar, Anna Rohrbach, Leonid Karlinsky, Trevor Darrell, and Amir Globerson. Bringing image scene structure to video via frame-clip consistency of object tokens. In *Thirty-Sixth Conference on Neural Information Processing Systems*, 2022. 2, 6, 7, 1

[6] Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. *ArXiv*, abs/1607.06450, 2016. 3

[7] Amir Bar, Roei Herzig, Xiaolong Wang, Anna Rohrbach, Gal Chechik, Trevor Darrell, and A. Globerson. Compositional video synthesis with action graphs. In *ICML*, 2021. 3

[8] Fabien Baradel, Natalia Neverova, Christian Wolf, Julien Mille, and Greg Mori. Object level visual reasoning in videos. In *ECCV*, pages 105–121, 2018. 3

[9] Peter W Battaglia, Jessica B Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, et al. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018. 3

[10] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *Proceedings of the International Conference on Machine Learning (ICML)*, July 2021. 3, 5, 6, 7, 2

[11] Deblina Bhattacharjee, Tong Zhang, Sabine Süsstrunk, and Mathieu Salzmann. Muit: An end-to-end multitask learning transformer. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12021–12031, 2022. 3

[12] David Brüggemann, Menelaos Kanakis, Stamatios Georgoulis, and Luc Van Gool. Automated search for resource-efficient branched multi-task networks. *ArXiv*, abs/2008.10292, 2020. 3

[13] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 213–229, 2020. 2

[14] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 6, 2

[15] Rich Caruana. Multitask learning. In *Encyclopedia of Machine Learning and Data Mining*, 1998. 3

[16] L. Caruso, R. Russo, and S. Savino. Microsoft kinect v2 vision system in a manufacturing application. *Robotics and Computer-Integrated Manufacturing*, 48:174–181, 2017. 5

[17] Yuhua Chen, Wen Li, Xiaoran Chen, and Luc Van Gool. Learning semantic segmentation from synthetic data: A geometrically guided input-output adaptation approach. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1841–1850, 2019. 8

[18] Yujin Chen, Matthias Nießner, and Angela Dai. 4dcontrast: Contrastive learning with dynamic correspondences for 3d scene understanding. *ArXiv*, abs/2112.02990, 2021. 2

[19] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *ECCV*, 2020. 3

[20] Bowen Cheng, Maxwell D. Collins, Yukun Zhu, Ting Liu, Thomas S. Huang, Hartwig Adam, and Liang-Chieh Chen. Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12472–12482, 2020. 3

[21] Vasileios Choutas, Philippe Weinzaepfel, Jérôme Revaud, and Cordelia Schmid. Potion: Pose motion representation for action recognition. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7024–7033, 2018. 2

[22] Michael Crawshaw. Multi-task learning with deep neural networks: A survey. *ArXiv*, abs/2009.09796, 2020. 3

[23] CR De Souza, A Gaidon, Y Cabon, and AM Lopez Pena. Procedural generation of videos to train deep action recognition networks. In *CVPR*, 2017. 6, 5

[24] César Roberto de Souza, Adrien Gaidon, Yohann Cabon, and Antonio Manuel López Peña. Procedural generation of videos to train deep action recognition networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2594–2604, 2017. 3

[25] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021. 2, 3

[26] Yong Du, Wei Wang, and Liang Wang. Hierarchical recurrent neural network for skeleton based action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1110–1118, 2015. 2

[27] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. In *ICCV*, 2021. 2, 3, 5, 7

[28] C. Feichtenhofer, H. Fan, J. Malik, and K. He. Slowfast networks for video recognition. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6201–6210, 2019. 6, 7, 2, 4

[29] Chuang Gan, Jeremy Schwartz, Seth Alter, Martin Schrimpf, James Traer, Julian De Freitas, Jonas Kubilius, Abhishek Bhandwaldar, Nick Haber, Megumi Sano, Kuno Kim, Elias Wang, Damian Mrowca, Michael Lingelbach, Aidan Curtis, Kevin T. Feigelis, Daniel Bear, Dan Gutfreund, David Cox, James J. DiCarlo, Josh H. McDermott, Joshua B. Tenenbaum, and Daniel L. K. Yamins. Threedworld: A platform for interactive multi-modal physical simulation. *ArXiv*, abs/2007.04954, 2021. 2, 3

[30] Chen Gao, Jiarui Xu, Yuliang Zou, and Jia-Bin Huang. Drg: Dual relation graph for human-object interaction detection. *ArXiv*, abs/2008.11714, 2020. 3

[31] Yuan Gao, Haoping Bai, Zequn Jie, Jiayi Ma, Kui Jia, and Wei Liu. Mtl-nas: Task-agnostic neural architecture search towards general-purpose multi-task learning. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11540–11549, 2020. 3

[32] Yuan Gao, Qi She, Jiayi Ma, Mingbo Zhao, W. Liu, and Alan Loddon Yuille. Nddr-cnn: Layerwise feature fusing in multi-task cnns by neural discriminative dimensionality reduction. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3200–3209, 2019. 3

[33] Rohit Girdhar, Joao Carreira, Carl Doersch, and Andrew Zisserman. Video action transformer network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 244–253, 2019. 2, 3

[34] Rohit Girdhar, Deva Ramanan, Abhinav Kumar Gupta, Josef Sivic, and Bryan C. Russell. Actionvlad: Learning spatio-temporal aggregation for action classification. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3165–3174, 2017. 2, 3

[35] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense. In *ICCV*, page 5, 2017. 5, 3

[36] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, Miguel Martin, Tushar Nagarajan, Ilija Radosavovic, Santhosh Kumar Ramakrishnan, Fiona Ryan, Jayant Sharma, Michael Wray, Mengmeng Xu, Eric Zhongcong 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 Gonzalez, James Hillis, Xuhua Huang, Yifei Huang, Wenqi Jia, Weslie Khoo, Jachym Kolar, 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 Somasundaram, Audrey Southerland, Yusuke Sugano, Ruijie Tao, Minh Vo, Yuchen Wang, Xindi Wu, Takuma Yagi, Yunyi Zhu, Pablo Arbelaez, David Crandall, Dima Damen, Giovanni Maria Farinella, Bernard Ghanem, Vamsi Krishna Ithapu, C. V. Jawahar, Hanbyul Joo, Kris Kitani, Haizhou Li, Richard 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. *CoRR*, abs/2110.07058, 2021. 2, 5, 6, 4

[37] Chunhui Gu, Chen Sun, David A. Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, Cordelia Schmid, and Jitendra Malik. AVA: A video dataset of spatio-temporally localized atomic visual actions. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 6047–6056. IEEE Computer Society, 2018. 5, 7, 4

[38] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 3

[39] Roei Herzig, Amir Bar, Huijuan Xu, Gal Chechik, Trevor Darrell, and Amir Globerson. Learning canonical representations for scene graph to image generation. In *European Conference on Computer Vision*, 2020. 3

[40] Roei Herzig, Elad Ben-Avraham, Karttikeya Mangalam, Amir Bar, Gal Chechik, Anna Rohrbach, Trevor Darrell, and Amir Globerson. Object-region video transformers. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2, 3, 1

[41] Roei Herzig, Elad Levi, Huijuan Xu, Hang Gao, Eli Brosh, Xiaolong Wang, Amir Globerson, and Trevor Darrell. Spatio-temporal action graph networks. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019. 2, 3

[42] Roei Herzig, Moshiko Raboh, Gal Chechik, Jonathan Berant, and Amir Globerson. Mapping images to scene graphs with permutation-invariant structured prediction. In *Advances in Neural Information Processing Systems (NIPS)*, 2018. 3

[43] Geoffrey E. Hinton, Nitish Srivastava, A. Krizhevsky, Ilya Sutskever, and R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *ArXiv*, abs/1207.0580, 2012. 3, 4

[44] Ronghang Hu and Amanpreet Singh. Unit: Multimodal multitask learning with a unified transformer. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1419–1429, 2021. 3

[45] Hochul Hwang, Cheongjae Jang, Geonwoo Park, Junghyun Cho, and Ig-Jae Kim. Eldersim: A synthetic data generation platform for human action recognition in eldercare applications, 2020. 6, 5

[46] Hochul Hwang, Cheongjae Jang, Geonwoo Park, Junghyun Cho, and Ig-Jae Kim. Eldersim: A synthetic data generation platform for human action recognition in eldercare applications. *ArXiv*, abs/2010.14742, 2020. 5

[47] Achiya Jerbi, Roei Herzig, Jonathan Berant, Gal Chechik, and Amir Globerson. Learning object detection from captions via textual scene attributes. *ArXiv*, abs/2009.14558, 2020. 3

[48] Jingwei Ji, Ranjay Krishna, Li Fei-Fei, and Juan Carlos Niebles. Action genome: Actions as composition of spatio-temporal scene graphs. *arXiv preprint arXiv:1912.06992*, 2019. 2, 3

[49] Yanli Ji, Feixiang Xu, Yang Yang, Fumin Shen, Heng Tao Shen, and Wei-Shi Zheng. A large-scale varying-view rgb-d action dataset for arbitrary-view human action recognition, 2019. 5, 4

[50] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European Conference on Computer Vision (ECCV)*, 2022. 2, 6

[51] Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image generation from scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1219–1228, 2018. 3

[52] Vicky S. Kalogeiton, Philippe Weinzaepfel, Vittorio Ferrari, and Cordelia Schmid. Action tubelet detector for spatio-temporal action localization. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 4415–4423, 2017. 2

[53] Angjoo Kanazawa, Jason Y. Zhang, Panna Felsen, and Jitendra Malik. Learning 3d human dynamics from video, 2018. 5

[54] Keizo Kato, Yin Li, and Abhinav Gupta. Compositional learning for human object interaction. In *ECCV*, 2018. 3

[55] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 2, 5, 6

[56] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7482–7491, 2018. 3

[57] Tae Soo Kim and Gregory D. Hager. Safcar: Structured attention fusion for compositional action recognition, 2020. 1, 2

[58] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 3, 4

[59] Ranjay Krishna, Ines Chami, Michael S. Bernstein, and Li Fei-Fei. Referring relationships. *ECCV*, 2018. 3

[60] Rosario Leonardi, Francesco Ragusa, Antonino Furnari, and Giovanni Maria Farinella. Egocentric human-object interaction detection exploiting synthetic data, 2022. 6, 5

[61] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, Online and Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics. 2

[62] Kunchang Li, Yali Wang, Peng Gao, Guanglu Song, Yu Liu, Hongsheng Li, and Yu Qiao. Uniformer: Unified transformer for efficient spatiotemporal representation learning, 2022. 1, 2

[63] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *ArXiv*, abs/1908.03557, 2019. 3

[64] Xiujun Li, Xi Yin, Chunyuan Li, Xiaowei Hu, Pengchuan Zhang, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, Yejin Choi, and Jianfeng Gao. Oscar: Object-semantics aligned pre-training for vision-language tasks. *ECCV 2020*, 2020. 3

[65] Yingwei Li, Yi Li, and Nuno Vasconcelos. Resound: Towards action recognition without representation bias. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 5, 3

[66] Yanghao Li, Chao-Yuan Wu, Haoqi Fan, Karttikeya Mangalam, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. Mvitv2: Improved multiscale vision transformers for classification and detection. In *CVPR*, 2022. 2, 3, 5, 6, 7, 8, 4

[67] Junwei Liang, Lu Jiang, Juan Carlos Niebles, Alexander G. Hauptmann, and Li Fei-Fei. Peeking into the future: Predicting future person activities and locations in videos. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5718–5727, 2019. 3

[68] Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding, 2018. 2

[69] Kevin Qinghong Lin, Alex Jinpeng Wang, Mattia Soldan, Michael Wray, Rui Yan, Eric Zhongcong Xu, Difei Gao, Rongcheng Tu, Wenzhe Zhao, Weijie Kong, et al. Egocentric video-language pretraining. *arXiv preprint arXiv:2206.01670*, 2022. 2

[70] Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen. Bmn: Boundary-matching network for temporal action proposal generation. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3888–3897, 2019. 6, 2

[71] Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. In *NeurIPS*, 2021. 3

[72] Liyang Liu, Yi Li, Zhanghui Kuang, Jing-Hao Xue, Yimin Chen, Wenming Yang, Qingmin Liao, and Wayne Zhang. Towards impartial multi-task learning. In *ICLR*, 2021. 3

[73] Mengyuan Liu, Hong Liu, and Chen Chen. Enhanced skeleton visualization for view invariant human action recognition. *Pattern Recognition*, 68:346–362, 2017. 2

[74] Shikun Liu, Edward Johns, and Andrew J. Davison. End-to-end multi-task learning with attention. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1871–1880, 2019. 3

[75] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. *arXiv preprint arXiv:2106.13230*, 2021. 3, 1, 2

[76] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. Smpl: a skinned multi-person linear model. *ACM Trans. Graph.*, 34:248:1–248:16, 2015. 5

[77] Joanna Materzynska, Tete Xiao, Roei Herzig, Huijuan Xu, Xiaolong Wang, and Trevor Darrell. Something-else: Com-

positional action recognition with spatial-temporal interaction networks. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 5, 6, 7, 1, 2, 3

[78] Hiroaki Mikami, Kenji Fukumizu, Shogo Murai, Shuji Suzuki, Yuta Kikuchi, Taiji Suzuki, Shin ichi Maeda, and Kohei Hayashi. A scaling law for synthetic-to-real transfer: How much is your pre-training effective?, 2021. 3

[79] Samarth Mishra, Rameswar Panda, Cheng Perng Phoo, Chun-Fu Chen, Leonid Karlinsky, Kate Saenko, Venkatesh Saligrama, and Rogério Schmidt Feris. Task2sim: Towards effective pre-training and transfer from synthetic data. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9184–9194, 2022. 3

[80] 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. 3

[81] Mandela Patrick, Dylan Campbell, Yuki M. Asano, Ishan Misra Florian Metze, Christoph Feichtenhofer, Andrea Vedaldi, and Joao F. Henriques. Keeping your eye on the ball: Trajectory attention in video transformers, 2021. 3, 5, 6, 2

[82] Xingchao Peng, Baochen Sun, Karim Ali, and Kate Saenko. Learning deep object detectors from 3d models. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1278–1286, 2015. 3

[83] Aayush Prakash, Shaad Boochoon, Mark Brophy, David Acuna, Eric Cameracci, Gavriel State, Omer Shapira, and Stan Birchfield. Structured domain randomization: Bridging the reality gap by context-aware synthetic data. *2019 International Conference on Robotics and Automation (ICRA)*, pages 7249–7255, 2019. 3

[84] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10313–10322, 2021. 2

[85] Siyuan Qiao, Yukun Zhu, Hartwig Adam, Alan Loddon Yuille, and Liang-Chieh Chen. Vip-deeplab: Learning visual perception with depth-aware video panoptic segmentation. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3996–4007, 2021. 3

[86] Moshiko Raboh, Roei Herzig, Gal Chechik, Jonathan Berant, and Amir Globerson. Differentiable scene graphs. In *WACV*, 2020. 3

[87] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 5

[88] Mike Roberts, Jason Ramapuram, Anurag Ranjan, Atulit Kumar, Miguel Angel Bautista, Nathan Paczan, Russ Webb, and Joshua M. Susskind. Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In *International Conference on Computer Vision (ICCV) 2021*, 2021. 6, 5

[89] Germán Ros, Laura Sellart, Joanna Materzynska, David Vázquez, and Antonio M. López. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3234–3243, 2016. 3

[90] Victor Sanh, Albert Webson, Colin Raffel, Stephen Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Fevry, Jason Alan Fries, Ryan Teehan, Teven Le Scao, Stella Biderman, Leo Gao, Thomas Wolf, and Alexander M Rush. Multitask prompted training enables zero-shot task generalization. In *International Conference on Learning Representations*, 2022. 2

[91] Adam Santoro, David Raposo, David G Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap. A simple neural network module for relational reasoning. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. 3

[92] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A platform for embodied ai research. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9338–9346, 2019. 2, 3

[93] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. Ntu rgb+d: A large scale dataset for 3d human activity analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1010–1019, 2016. 5

[94] Chen Sun, Abhinav Shrivastava, Carl Vondrick, Kevin Murphy, Rahul Sukthankar, and Cordelia Schmid. Actor-centric relation network. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 318–334, 2018. 3

[95] Guolei Sun, Thomas Probst, Danda Pani Paudel, Nikola Popovic, Menelaos Kanakis, Jagruti R. Patel, Dengxin Dai, and Luc Van Gool. Task switching network for multitask learning. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8271–8280, 2021. 3

[96] Ximeng Sun, Rameswar Panda, and Rogério Schmidt Feris. Adashare: Learning what to share for efficient deep multitask learning. *ArXiv*, abs/1911.12423, 2020. 3

[97] Xu Sun, Tongwei Ren, Yuan Zi, and Gangshan Wu. Video visual relation detection via multi-modal feature fusion. *Proceedings of the 27th ACM International Conference on Multimedia*, 2019. 3

[98] Hao Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. In *EMNLP*, 2019. 3

[99] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training, 2022. 1, 2

[100] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pages 10347–10357. PMLR, 2021. 3

[101] Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. Mti-net: Multi-scale task interaction networks for multi-task learning. In *ECCV*, 2020. 3

[102] Gül Varol, Ivan Laptev, Cordelia Schmid, and Andrew Zisserman. Synthetic humans for action recognition from unseen viewpoints. *ArXiv*, abs/1912.04070, 2021. 3

[103] Gül Varol, Ivan Laptev, Cordelia Schmid, and Andrew Zisserman. Synthetic humans for action recognition from unseen viewpoints. In *IJCV*, 2021. 5, 4

[104] Tu Vu, Brian Lester, Noah Constant, Rami Al-Rfou', and Daniel Cer. SPoT: Better frozen model adaptation through soft prompt transfer. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5039–5059, Dublin, Ireland, May 2022. Association for Computational Linguistics. 2

[105] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Mathieu Cord, and Patrick Pérez. Dada: Depth-aware domain adaptation in semantic segmentation. In *ICCV*, 2019. 8

[106] Xiaolong Wang and Abhinav Gupta. Videos as space-time region graphs. In *ECCV*, 2018. 2, 3

[107] Zhonghao Wang, Mo Yu, Yunchao Wei, Rogério Schmidt Feris, Jinjun Xiong, Wen mei W. Hwu, Thomas S. Huang, and Humphrey Shi. Differential treatment for stuff and things: A simple unsupervised domain adaptation method for semantic segmentation. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12632–12641, 2020. 3

[108] Zifeng Wang, Zizhao Zhang, Sayna Ebrahimi, Ruoxi Sun, Han Zhang, Chen-Yu Lee, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, et al. Dualprompt: Complementary prompting for rehearsal-free continual learning. *European Conference on Computer Vision*, 2022. 2

[109] Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 139–149, 2022. 2

[110] Yo whan Kim, SouYoung Jin, Rameswar Panda, Hilde Kuehne, Leonid Karlinsky, Samarth Mishra, Venkatesh Saligrama, Kate Saenko, Aude Oliva, and Rogério Schmidt Feris. How transferable are video representations based on synthetic data?, 2022. 3

[111] Chao-Yuan Wu and Philipp Krähenbühl. Towards Long-Form Video Understanding. In *CVPR*, 2021. 2, 3

[112] Chao-Yuan Wu, Yanghao Li, Karttikeya Mangalam, Haoqi Fan, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. Memvit: Memory-augmented multiscale vision transformer for efficient long-term video recognition. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13577–13587, 2022. 2

[113] F. Xia, Amir Roshan Zamir, Zhi-Yang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. Gibson env: Real-world perception for embodied agents. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9068–9079, 2018. 3

[114] Bingjie Xu, Yongkang Wong, Junnan Li, Qi Zhao, and M. Kankanhalli. Learning to detect human-object interactions with knowledge. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2019–2028, 2019. 3

[115] Keyulu Xu, Jingling Li, Mozhi Zhang, Simon S. Du, Ken ichi Kawarabayashi, and Stefanie Jegelka. What can neural networks reason about? In *International Conference on Learning Representations*, 2020. 3

[116] Yangyang Xu, Xiangtai Li, Haobo Yuan, Yibo Yang, Jing Zhang, Yunhai Tong, Lefei Zhang, and Dacheng Tao. Multi-task learning with multi-query transformer for dense prediction. *ArXiv*, abs/2205.14354, 2022. 3

[117] Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *ArXiv*, abs/2001.06782, 2020. 3

[118] Vinicius Zambaldi, David Raposo, Adam Santoro, Victor Bapst, Yujia Li, Igor Babuschkin, Karl Tuyls, David Reichert, Timothy Lillicrap, Edward Lockhart, et al. Relational deep reinforcement learning. *arXiv preprint arXiv:1806.01830*, 2018. 3

[119] Amir Roshan Zamir, Alexander Sax, Bokui (William) Shen, Leonidas J. Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3712–3722, 2018. 2

[120] Chuhan Zhang, Ankush Gputa, and Andrew Zisserman. Temporal query networks for fine-grained video understanding. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 3

[121] Yubo Zhang, Pavel Tokmakov, Martial Hebert, and Cordelia Schmid. A structured model for action detection. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9967–9976, 2019. 3

[122] Jiaojiao Zhao, Yanyi Zhang, Xinyu Li, Hao Chen, Shuai Bing, Mingze Xu, Chunhui Liu, Kaustav Kundu, Yuanjun Xiong, Davide Modolo, Ivan Marsic, Cees G. M. Snoek, and Joseph Tighe. Tuber: Tubelet transformer for video action detection. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13588–13597, 2022. 2

[123] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16795–16804, 2022. 2

[124] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *Int. J. Comput. Vis.*, 130:2337–2348, 2022. 2

[125] Lingli Zhou, Zhen Cui, Chunyan Xu, Zhenyu Zhang, Chaoqun Wang, Tong Zhang, and Jian Yang. Pattern-structure diffusion for multi-task learning. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4513–4522, 2020. 3