DEVIAS: Learning Disentangled Video Representations of Action and Scene

Kyungho Bae[1*], Geo Ahn[1*], Youngrae Kim[2*], and Jinwoo Choi[1†]

[1]Kyung Hee University [2]KAIST

{kyungho.bae, ahngeo11, jinwoochoi}@khu.ac.kr, youngrae.kim@kaist.ac.kr

Abstract. Video recognition models often learn scene-biased action representation due to the spurious correlation between actions and scenes in the training data. Such models show poor performance when the test data consists of videos with unseen action-scene combinations. Although Scene-debiased action recognition models might address the issue, they often overlook valuable scene information in the data. To address this challenge, we propose to learn DisEntangled VIdeo representations of Action and Scene (DEVIAS), for more holistic video understanding. We propose an encoder-decoder architecture to learn disentangled action and scene representations with a single model. The architecture consists of a disentangling encoder (DE), an action mask decoder (AMD), and a prediction head. The key to achieving the disentanglement is employing both DE and AMD during training time. The DE uses the slot attention mechanism to learn disentangled action and scene representations. For further disentanglement, an AMD learns to predict action masks, given an action slot. With the resulting disentangled representations, we can achieve robust performance across diverse scenarios, including both seen and unseen action-scene combinations. We rigorously validate the proposed method on the UCF-101, Kinetics-400, and HVU datasets for the seen, and the SCUBA, HAT, and HVU datasets for unseen action-scene combination scenarios. Furthermore, DEVIAS provides flexibility to adjust the emphasis on action or scene information depending on dataset characteristics for downstream tasks. DEVIAS shows favorable performance in various downstream tasks: Diving48, Something-Something-V2, UCF-101, and ActivityNet. The code is available at https://github.com/KHU-VLL/DEVIAS.

Keywords: Action recognition · Video recognition · Scene recognition · Video representation learning · Disentangled representation learning

1 Introduction

Humans can naturally understand the content of a video by extracting human actions from the surrounding scene context. Even when encountering a previously unseen action-scene combination, humans easily recognize both the action and the scene: *e.g.* in Figure 1 (b), the people are *dancing* in a *football field*.

* Equally contributed first authors. † Corresponding author.

(a) Seen combination: *dancing* on a *stage*. (b) Unseen combination: *dancing* in a *football field*.

Fig. 1: Why do we need disentangled video representations? Video recognition models often learn scene-biased representations due to the spurious correlation [8, 39] between action and scene in the dataset. Although such models might work well when video contains an action-scene combination seen during training *e.g.* (a), they would fail when video contains an unseen combination *e.g.* (b). In contrast, scene debiased video models [14, 58] could accurately recognize the action regardless of combinations. However, they are not capable of predicting scenes. In this work, we propose to learn *disentangled* action and scene representations. The disentangled model can understand both action and scene well, including both seen and unseen action-scene combinations, *e.g.* it correctly predicts *dancing* on a *stage* (a) and *dancing* in a *football field* (b).

Unlike humans, most video action recognition methods struggle to decompose actions and scene context from an input video. Instead, video action recognition methods tend to learn scene-biased action representation due to the spurious correlation between actions and scenes in the video datasets. The existing video datasets [18, 32, 51] often consist of limited combinations of action-scene pairs for each action class, *e.g.* if the scene of a video is a football field, the action is always *playing football* and vice versa. However, in reality, diverse actions such as dancing or cheerleading can also take place on a football field [8]. The reason for the limited action-scene combinations in the dataset stems from the high cost of constructing a video dataset with diverse combinations, rather than a lack of such scenarios in the real world. Therefore, a desired model should have robust performance across diverse action-scene combination scenarios.

A scene-biased action recognition model is likely to predict actions based on the scene context rather than the action itself, leading to errors when encountering unseen action-scene combinations [8, 10, 39]. For example, as shown in Figure 1 (b), scene-biased action recognition models are likely to misclassify the action as *playing football* instead of *dancing*. Scene-debiased action recognition models [3, 8, 14, 38, 58] might be a solution to the problem. The scene-debiased action recognition models show significant improvement for unseen action-scene combinations [38] and for scenarios where actions and scenes are barely correlated [39, 62]. However, as illustrated in Figure 1, scene-debiased action recognition models often overlook the scene context as they are trained to disregard scene information, omitting potentially valuable context.

In this work, we move beyond the limitations of prior works that disregard scene context. We tackle an interesting yet relatively under-explored problem: learning DisEntangled VIdeo representations of Action and Scene (DEVIAS) for holistic video understanding. Having *both action and scene* representations provides richer information than scene-debiased representations. With *disentangled* action and scene representations, a video model can understand video regardless of seen or unseen action-scene combinations. As illustrated in the third row of Figure 1, with disentangled representations, a model could accurately recognize that the people are *dancing* whether on a *stage* (a) or in a *football field* (b).

Disentangled action-scene representations allow for tailored applications; one can adjust the emphasis on action or scene to suit specific tasks. For instance, we can leverage scene context in some datasets where it is beneficial [18, 32, 51] to boost action performance. On the other hand, in some tasks where scene information is non-beneficial, we could encourage a model to focus on the action rather than the scene context: *e.g.* barely correlated actions and scenes [24,39,62], and action-scene combinations vary between training and test phases [10, 38, 45].

In this paper, we propose DEVIAS, a novel encoder-decoder architecture to learn disentangled representations of action and scene with a single model. DEVIAS consists of three parts: a disentangling encoder (DE), an action mask decoder (AMD), and an action/scene classification head. The key to achieving the disentanglement is employing *both DE and AMD during training time.* In the DE, we employ the slot attention [42] to learn disentangled action and scene representations. In the slot attention, multiple learnable slots compete with each other as we normalize the attention coefficients over the slot axis. After a few iterations of slot attention, the learnable slots progressively learn distinct information, *e.g.* action, and scene. On top of DE, AMD further disentangles action and scene representations. AMD is a lightweight decoder that learns to predict action masks given an action slot as an input. Thanks to the complementary nature of the slot attention, DEVIAS encourages the DE to learn not only good action representation but also good scene representation. As a result, the DE effectively learns disentangled action and scene representations.

To validate the effectiveness of DEVIAS, we carefully design a set of controlled experiments. Through the experiments, we verify the effectiveness of each representation in seen action-scene combination scenarios: UCF-101 [51], Kinetics-400 [32], and HVU [13] and in unseen action-scene combination scenarios: SCUBA [38], HAT [10], and HVU [13]. DEVIAS shows favorable performance over the baselines in both seen and unseen combination scenarios. DEVIAS also show favorable performance on various downstream tasks: Diving48 [39], Something-Something-V2 [24], UCF-101 [51], and ActivityNet [18].

In this work, we make the following major contributions:

- We tackle an interesting and challenging yet relatively under-explored problem of learning *disentangled action and scene representations.* We aim to *shift the paradigm* from merely recognizing actions to recognizing both actions and scenes in the video recognition field.
- We introduce DEVIAS, a novel encoder-decoder architecture designed to learn disentangled action and scene representations. The key to achieving the disentanglement is employing both the disentangling encoder with slot attention and the action mask decoder during training time.

– We conduct extensive experiments to validate the effectiveness of DEVIAS. DEVIAS shows robust performance over the baselines, in both seen and unseen action-scene combination scenarios and on various downstream tasks.

2 Related Work

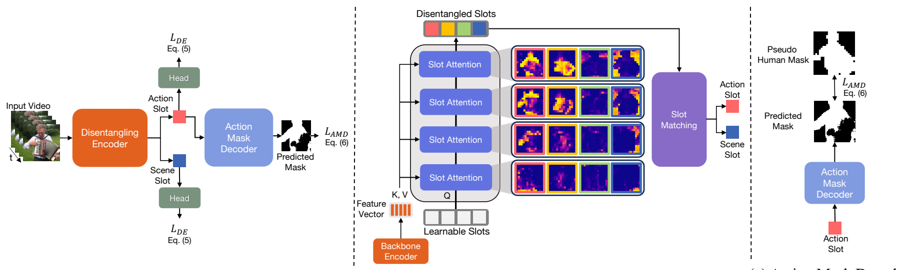
Video action recognition. The last decade has seen remarkable progress in video action recognition. Key approaches to action recognition include 2D CNNs [16,31,40,46,50,71], two-stream CNNs [21,50], 3D CNNs [6,20,21,30,53, 54,59], 2D and 1D separable CNNs [54,66] or more recent transformer architectures [1,4,19,22,27,43,47,63,68]. Despite the great progress, a common limitation among the previous works is scene-biased action representation. The bias stems from the spurious correlation between action and scene in the training data, often leading to poor performance on the test data with different distributions. In contrast to the prior works, we aim to reduce the influence of the spurious correlation by learning disentangled representations of action and scene.

Mitigating scene bias in action recognition. The community has identified scene bias [8,39] as the devil of action recognition as biased models do not generalize well to new tasks/domains. Scene debiasing is beneficial, enhancing performance in downstream tasks [3, 8, 14, 15, 58], data-efficiency [23, 73], and domain adaptation [9,49]. Yet, an often-overlooked aspect is that action representation, when debiased from the scene context, may lose valuable contextual information. In contrast to scene-debiasing, DEVIAS learns disentangled action and scene representations. The disentangled representations enable tailored emphasis on either action or scene depending on the specific downstream tasks and datasets.

Disentangled representation learning. Generative modeling works have extensively explored disentangled representation learning to manipulate each attribute for image and video generation [5, 7, 12, 28, 29, 36, 37, 44, 48, 57, 61, 67]. Disentangled representation learning for video understanding includes disentangling different attributes [55, 69], learning dynamic and static components of videos [56, 70], and disentangling action and scene [41, 60] to improve action recognition. We also focus on disentangled representations for video understanding. Unlike prior works that overlook the quality and utility of the scene representation [41,60], we aim to learn not only high-quality action but also high-quality scene representation. To the best of our knowledge, DEVIAS is a pioneering effort in achieving this balanced focus on both action and scene recognition.

3 DEVIAS

We introduce DEVIAS, an encoder-decoder architecture designed to learn disentangled action and scene representations within a unified model, as illustrated in Figure 2. DEVIAS consists of i) a disentangling encoder (DE), ii) an action mask decoder (AMD), and iii) an action/scene prediction head. To learn disentangled representations, it is crucial to employ *both DE and AMD* during training.



(a) Overview. (b) Disentangling Encoder. (c) Action Mask Decoder.

Fig. 2: Overview of DEVIAS. (a) DEVIAS consists of i) a disentangling encoder (DE), ii) an action mask decoder (AMD), and iii) a classification head. (b) Given an input video, the DE first extracts a feature vector using a backbone encoder. Then the DE learns multiple slots. Given input learnable slots as queries, the slot attention iteratively attends to encoded features as keys and values. As a result of the slot attention, the slots progressively learn distinct information, *i.e.* action, and scene. Then a matching function assigns each slot an action or a scene slot. We train action/scene slots with corresponding labels. (c) Given an action slot, the AMD predicts action masks to learn disentangled representations As slots are complementary in slot attention, the AMD encourages the DE to learn not only good action but also good scene representations.

Given an input video, DE extracts a feature vector $\mathbf{X} \in \mathbb{R}^{NT \times D}$ using a backbone transformer encoder, where N is the number of spatial patches, T is the number of frames, and D denotes the dimension of patch embeddings. DE captures distinct information from the feature vectors, *i.e.* action, and scene by slot attention. Given K learnable slots as queries, the slot attention iteratively attends to encoded features as keys and values. Then the DE outputs disentangled slots. With the slot attention, DEVIAS learns action and scene representations by competition among multiple slots. Then a matching function solves K to 2 matching problem to determine an action and a scene slot. We supervise action slot learning using action labels covering N_A action types and scene slot learning using scene labels covering N_S scene types. We use a shared head for the action and scene class prediction. For disentanglement, we employ a lightweight action mask decoder. Given an action slot, AMD learns to predict action masks. We provide detailed descriptions of the DE and the prediction head in Section 3.1, AMD in Section 3.2, and training & inference of DEVIAS in Section 3.3.

3.1 Disentangling Encoder

Slot attention for disentangled representation learning. To learn disentangled action and scene representations, DE employs the slot attention mechanism [33,42]. In each slot attention iteration, we project an input feature vector $\mathbf{X} \in \mathbb{R}^{NT \times D}$, and K learnable slots $\mathbf{S} \in \mathbb{R}^{K \times D}$ to a common space with the dimension D_h as follows: $\mathbf{Q} = \mathbf{S}\mathbf{W}_Q \in \mathbb{R}^{K \times D_h}$, $\mathbf{K} = \mathbf{X}\mathbf{W}_K \in \mathbb{R}^{NT \times D_h}$, and $\mathbf{V} = \mathbf{X}\mathbf{W}_V \in \mathbb{R}^{NT \times D_h}$, where $\mathbf{W}_Q$, $\mathbf{W}_K$, and $\mathbf{W}_V$ are the $D \times D_h$ query, key, and value projection matrices, respectively. Here, we omit the batch dimension B, layer normalization [2], and GELU [26] for brevity.

Given the query, key, and value, we define slot attention operation as follows:

$$\mathbf{M} = \mathbf{K}\mathbf{Q}^{\top}/\sqrt{D_h}. \tag{1}$$

We normalize the attention map $\mathbf{M} \in \mathbb{R}^{NT \times K}$ along the *slot*-axis fostering competitive learning of action and scene representations in slots:

$$\mathbf{A}(n,k) = \frac{\exp\left(\mathbf{M}(n,k)\right)}{\sum_{i=1}^{K} \exp\left(\mathbf{M}(n,i)\right)}. \tag{2}$$

Here, n is the key index and k is the slot index. Then we normalize $\mathbf{A}$ along the key-axis using the L^2 norm function, yielding $\hat{\mathbf{A}}$. Then we attend to input features with the attention map $\hat{\mathbf{A}}$ as follows:

$$\mathbf{Z} = \hat{\mathbf{A}}^{\top}\mathbf{V}. \tag{3}$$

For each iteration $m \in [1, M]$, we update the slots as follows:

$$\mathbf{S} = \mathtt{MLP}(\mathbf{S} + \mathbf{Z}) + \mathbf{S} + \mathbf{Z}. \tag{4}$$

The slots progressively encode disentangled action and scene information with iterative updates as shown in Figure 2 (b). To the best of our knowledge, this is the first work that successfully utilizes slot attention to obtain disentangled representations in video recognition.

Slot Matching. Among K learnable slots, DE selects an action and a scene slot by solving a K to 2 matching problem. We use the cross-entropy between a true label and a prediction as the cost function for the matching. We use a classification head ψ, shared across the action and scene tasks, for the prediction. After computing a $K \times 2$ cost matrix, we solve the matching problem using the Hungarian algorithm [35]. Please see the supplementary material for details.

Disentangling encoder loss. For action and scene slot learning, we define the disentangling encoder loss with a unified head for predicting a $N_A + N_S$ dimensional vector as follows:

$$L_{DE} = -\sum_{c=1}^{N_A+N_S} [y_c^a \log(\psi(\mathbf{S}_{action})) + y_c^s \log \psi(\mathbf{S}_{scene})]. \tag{5}$$

Here, y^a denotes the ground-truth action label and y^s represents the ground-truth scene label. $\mathbf{S}_{action}$ is the action slot, and $\mathbf{S}_{scene}$ is the scene slot. In cases where the dataset does not provide ground truth scene labels, we obtain y^s by running a frozen off-the-shelf scene recognition model. For more details, please see the supplementary material.

Limitation. Due to the spurious correlation between actions and scenes in training data, *naively* using the slot attention does not fully disentangle action and scene. As shown in Figure 2 (b), different slots represent distinct regions of a video *e.g.* action (human), object, and scene regions. Since the training data has a spurious correlation, the disentangling loss (5) could be decreased even if an action slot and scene slot are assigned as opposite slots. The resulting model would learn entangled representations, leading to poor performance on test data with action-scene combinations different from the training data. To address the issue, we design AMD as described in the next subsection.

3.2 Action Mask Decoder

We introduce the Action Mask Decoder (AMD) as shown in Figure 2 (c). Given an action slot, AMD learns to predict an action mask. For the AMD to predict a high-quality mask, the input slot should contain action information not scene or object information. Therefore, employing the AMD prevents the DE from learning entangled action and scene representations. Since slots are complementary to each other in slot attention, learning good action representation by AMD encourages the DE to learn good scene representation as well.

Inspired by reconstruction-based representation learning methods [25,52], we employ a lightweight decoder ϕ. The decoder takes the action slots $\mathbf{S}_{action}$, and the pseudo-human mask $\hat{\mathbf{H}}$ as input and reconstructs an action mask. To obtain pseudo-human masks, we can employ any off-the-shelf method [11,14,64,65]. We choose a simple mask extraction approach [14] without learning. We define the action mask decoding loss as follows:

$$L_{AMD} = -\tilde{\mathbf{H}} \log(\phi(\mathbf{S}_{action})) - (1 - \tilde{\mathbf{H}}) \log(1 - \phi(\mathbf{S}_{action})). \tag{6}$$

$\tilde{\mathbf{H}} \in \mathbb{R}^N$ is the temporally averaged version of pseudo-human masks $\hat{\mathbf{H}} \in \mathbb{R}^{NT}$.

3.3 Training and Inference

Training. For model training, we define a total loss function as follows:

$$L = L_{DE} + \alpha L_{AMD} + \beta L_{AG} + \gamma L_{cos}. \tag{7}$$

Here, L_{AG} is the attention guidance loss to further guide the action slot learning. We define L_{AG} as a L^2 loss between the attention map and the pseudo-human masks: $L_{AG} = ||\mathbf{A}(:, k_{action}) - \hat{\mathbf{H}}||_2^2$, where k_{action} is the action slot index. We also incorporate the cosine similarity loss between every pair of slots to diversify slots: $L_{cos} = \frac{1}{K} \sum_{i=1}^{K} \sum_{j \neq i} [1 - \cos(\mathbf{S}_i, \mathbf{S}_j)]$. α, β, and γ are hyperparameters.

Inference. During the inference, we use the disentangling encoder only. We feed a video into the model to extract K slots. Among the K slots, we assign action and scene slots based on the highest output probability. The linear classifier ψ takes the action and scene slots to predict the action and scene labels.

4 Experimental Results

In this section, we carefully design and conduct extensive experiments to answer the following research questions: (1) Are the learned representations disentangled? (Section 4.5) (2) Are the disentangled representations beneficial for achieving a balanced action and scene recognition performance in seen and unseen action-scene combination scenarios? (Section 4.5) (3) Are disentangled representations beneficial for transfer learning? (Section 4.6) (4) How can we disentangle representations? (Section 4.7) To this end, we first provide details about the implementations in Section 4.1, the datasets in Section 4.2, the evaluation metrics in Section 4.3, and the baselines in Section 4.4.

4.1 Implementation Details

In this section, we briefly explain the implementation details. For the comprehensive details, please refer to the supplementary material.
Training. We densely sample 16 frames from each video to construct an input clip. We apply random cropping and resizing to every frame to get 224×224 pixels for each frame. We employ the ViT [17] pre-trained with self-supervised VideoMAE [52] on the target dataset, *e.g.* UCF-101 in Table 2, and Kinetics-400 in Table 3, as our backbone encoder. We employ a 3-layer MLP as our action mask decoder. We employ an off-the-shelf and non-learning-based pseudo-human mask extractor [14] which computes foreground probability using pixel value statistics. In cases where the ground-truth scene labels are not provided by the dataset, *e.g.* UCF-101 and Kinetics-400, we employ the ViT pre-trained on the Places365 [72] dataset as the frozen scene model to obtain pseudo scene labels.
Inference. During inference, we average predictions over multiple temporal views and spatial crops, resulting in 2×3 views in all experiments.

4.2 Datasets

In this section, we briefly describe the datasets used. For the complete details, please refer to the supplementary material. We evaluate DEVIAS across both *seen* and *unseen* action-scene combination scenarios, employing the standard training splits of either UCF-101 [51] or Kinetics-400 [32] for model training.
Seen combination. We use the original validation split of either UCF-101 [51] or Kinetics-400 [32] for testing.
Unseen combination. We use the two synthetic datasets, SCUBA and HAT, to test the models in diverse unseen action-scene combination scenarios. SCUBA [38] and HAT [10] contain diverse combinations in several dataset versions, denoted as SCUBA VQGAN-CLIP/Sinusoidal and HAT Random/Far.
Realistic dataset. To evaluate performance using more realistic data, we rearrange the holistic video understanding (HVU) dataset [13] which provides both action and scene labels. We train models on the train split of the HVU. We test the models on the seen and unseen combination splits we rearranged. Specifically, from the HVU validation set, we select videos with the same action-scene

(a) (b) (c) (d) (e)

Fig. 3: Example frames of the datasets. (a) *walking with dog* on the *grass* from HVU Seen [13], (b) *walking with dog* in *snowfield* from HVU Unseen [13], (c) *dancing* on a *golf course* from HAT Far [10], (d) *snowfield* from HAT Scene-Only [10], and (e) *feeding goats* from SCUBA VQGAN-CLIP [38].

combinations as the training set to construct a seen combination split. We select videos with the different action-scene combinations from the training set to construct an unseen split. We show example frames of the datasets in the Figure 3.

Downstream datasets. We pre-train models on the Kinetics-400. For fine-tuning, we use both temporal-biased datasets, the Something-Something V2 [24], and Diving48 [39], and scene-biased datasets, UCF-101 [51], and ActivityNet [18].

4.3 Evaluation Metric

We report action and scene recognition performance across both *seen* and *unseen* action-scene combination scenarios. We use top-1 action recognition accuracy for all the datasets and top-1 scene recognition for the HVU dataset. Since UCF-101 and Kinetics-400 datasets do not have scene labels, we resort to using pseudo-labels generated by a Places365 [72] pre-trained scene model. Given the fine-grained nature of the Places365 categories, we report top-5 accuracy for UCF-101 and Kinetics-400 scene recognition. To gauge a model's balanced performance in both seen and unseen combinations scenarios, we report the *harmonic mean* (H.M.) across four performance metrics as our main metric: i) seen, and ii) unseen combinations action, iii) seen, and iv) unseen combinations scene.

4.4 Baselines

Multi-task supervision. For a *fair* comparison, we mainly compare DEVIAS to the baselines with both action and scene supervision *exactly same* as ours. The Two-Token baseline involves a feature encoder and two distinct learnable tokens: one for action and another for scene, appended to the input patches. We supervise the action and scene tokens using their respective ground-truth labels. Additionally, we explore two variations of this baseline, each incorporating either the BE [58] or FAME [14] debiasing methods. Unlike the Two-Token approach, the One-Token baseline has a single token and employs a multi-task loss combining both action and scene losses to learn the token. We provide detailed descriptions and accompanying figures of the baselines in the supplementary materials.

Table 1: Sanity check of disentanglement using k-NN. We compare models trained on the UCF-101 [51] using k-NN classification accuracy (%) on the UCF-101 and HMDB-51 [34]. For each column, we show {`train`}-{`test`} features of k-NN. 'A' and 'S' represent action and scene, respectively. The **best** performance is highlighted.

Method	k-NN Normal Features (↑)				k-NN Reverse Features (↓)			
	UCF-101		HMDB-51		UCF-101		HMDB-51	
	A-A	S-S	A-A	S-S	A-S	S-A	A-S	S-A
Random	1.0	0.3	1.0	0.3	1.0	0.3	1.0	0.3
One-Token	87.7	43.5	**43.4**	**27.5**	87.7	43.5	43.4	27.5
Two-Token	84.2	**43.6**	37.4	25.7	62.8	37.1	27.3	20.7
DEVIAS	**89.7**	41.8	38.8	26.3	**4.5**	**0.3**	**1.9**	**0.9**
Upperbound	92.0	48.8	60.5	38.1	-	-	-	-

Single-task supervision. We compare DEVIAS with the baselines with single-task supervision for reference. There are two naive ViT baselines trained using only action or scene labels and two scene-debiasing baselines: BE [58] and FAME [14] trained only with action labels.

4.5 Quantitative Analysis

Sanity-check: k-NN experiments. Before the main experiments, we check the sanity of DEVIAS. For representation learning, we train DEVIAS and the baselines on the UCF-101 [51] train split. Then we store the action and scene feature vectors from all the training videos in the target datasets, UCF-101 and HMDB-51 [34]. For k-NN testing, we evaluate performance in two scenarios: using the same feature types as in training (k-NN Normal Features), and using the alternate feature types (k-NN Reverse Features), as shown in Table 1.

e.g. 'A-S' indicates training the k-NN classifier with action features and testing with scene features. We anticipate that if a model has disentangled action and scene representations, it would perform well in the k-NN Normal Features scenario, and show near-random performance in the k-NN Reverse Features scenario. In Table 1, we compare our method with the One-Token and Two-Token baselines and the random performance.

For the experiments, we employ a 10-NN setting and report top-1 accuracy. We train both the baseline and DEVIAS on UCF-101. The upper bound represents the performance achieved with supervised training on each dataset. In the k-NN Normal Features scenario, all the methods compared show reasonable action and scene recognition performances. However, in the k-NN Reverse scenario, only DEVIAS shows near-random performance. The results verify the disentangled action and scene representations of DEVIAS.

UCF-101 results. In Table 2, we evaluate the methods on the UCF-101 dataset. Notably, the action performance of the Two-Token baseline decreases compared to the Naive Action ViT baseline in unseen combination scenarios: 16.7% *vs.* 15.5%. The results indicate that the Two-Token baseline still learns scene-biased representations. When we apply the scene debiasing techniques to the Two-

Table 2: Action and scene recognition performance on the UCF-101 dataset. We report the Top-1 action recognition and the Top-5 scene recognition accuracies (%). We evaluate both seen and unseen action-scene combination scenarios. We also report the harmonic mean (H.M.) of the action recognition and scene recognition. V.C./Sin. denotes the SCUBA [38] VQGAN-CLIP/Sinusoidal; S.O./Rand. denotes the HAT [10] Scene-Only/Random. The **best** and the <u>second-best</u> H.M. numbers are highlighted.

Training Strategy	Supervision		Method	Action(↑)				Scene(↑)					H.M.
				Seen	Unseen			Seen	Unseen				
	Action	Scene			V.C.	Sin.	Mean		S.O.	Rand.	Far	Mean	
Single-Task	✓	×	Naive Action ViT	92.9	12.4	21.0	16.7	-	-	-	-	-	-
	✓	×	BE [58]	92.3	12.1	38.7	25.4	-	-	-	-	-	-
	✓	×	FAME [14]	91.6	24.8	15.6	20.2	-	-	-	-	-	-
	×	✓	Naive Scene ViT	-	-	-	-	72.0	61.7	62.8	69.6	64.7	-
Multi-Task	✓	✓	One-Token	91.9	5.0	21.8	13.4	74.0	60.5	58.0	66.5	61.7	34.7
	✓	✓	Two-Token	86.0	11.1	19.9	15.5	72.3	59.6	59.2	67.1	62.0	37.7
	✓	✓	Two-Token w/ BE [58]	89.9	13.0	20.5	16.8	74.2	62.3	59.3	69.5	63.7	40.1
	✓	✓	Two-Token w/ FAME [14]	89.5	25.3	15.3	20.3	73.2	61.4	62.8	70.3	64.8	<u>44.7</u>
Disentangle	✓	✓	DEVIAS	90.1	40.1	38.6	39.4	74.0	61.0	62.4	70.2	64.5	**61.1**

Token baseline, we observe action performance improvement of 1.3 $\sim$ 4.8 points in unseen combination scenarios, and 2.4 $\sim$ 7.0 points improvement in H.M. compared to the Two-Token without debiasing. DEVIAS stands out by showing a more balanced performance of action and scene, achieving a significant 16.4 points boost over the second-best method in the H.M.. Remarkably, as a single model, DEVIAS surpasses the oracle performance of individual Action ViT with debiasing (BE) and Naive Scene ViT models: 50.3% *vs*. 61.1%.

Kinetics-400 results. In Table 3, we present the experimental results on the Kinetics-400 dataset, where DEVIAS shows a 1.6 points improvement in H.M. over the second-best method. The overall trend among methods remains similar to the trend we observe in the UCF-101 dataset.

HVU results. To further validate DEVIAS using *more realistic data*, we show the results on the HVU dataset [13] in Table 4. DEVIAS achieves the best performance in both action and scene recognition performance compared to baselines, showing a 2.4 points improvement in H.M. over the second-best method. The results demonstrate the effectiveness of DEVIAS when using more realistic data.

4.6 Downstream Task

We investigate whether the disentangled action and scene representation is beneficial or not in various downstream tasks. We conduct a set of transfer learning experiments: using the model weights trained on the source dataset as the initialization, we fine-tune the models on the target datasets. For the target dataset fine-tuning, we use only the cross-entropy loss with action ground-truth labels.

We compare DEVIAS with naive baselines (Action ViT, Scene ViT), scene debiasing methods (BE [58], FAME [14]), and the multi-task baselines (Two-

Table 3: Action and scene recognition performance on the Kinetics-400. We report the Top-1 action recognition and the Top-5 scene recognition accuracies (%). We evaluate both seen and unseen action-scene combination scenarios. We also report the harmonic mean (H.M.) of the action recognition and scene recognition. V.C./Sin. denotes the SCUBA [38] VQGAN-CLIP/Sinusoidal; S.O./Rand. denotes the HAT [10] Scene-Only/Random. The **best** and the <u>second-best</u> H.M. numbers are highlighted.

Training Strategy	Supervision		Method	Action(↑)				Scene(↑)					H.M.
				Seen	Unseen			Seen	Unseen				
	Action	Scene			V.C.	Sin.	Mean		S.O.	Rand.	Far	Mean	
Single-Task	✓	×	Naive Action ViT	76.8	41.6	49.6	45.6	-	-	-	-	-	-
	✓	×	BE [58]	77.6	43.2	52.2	47.7	-	-	-	-	-	-
	✓	×	FAME [14]	77.8	49.7	56.8	53.3	-	-	-	-	-	-
	×	✓	Naive Scene ViT	-	-	-	-	86.5	82.6	79.9	81.2	81.2	-
Multi-Task	✓	✓	One-Token	74.2	35.2	45.6	40.4	87.9	83.8	80.8	81.5	82.0	64.7
	✓	✓	Two-Token	75.0	34.9	46.6	40.8	86.4	75.8	78.3	80.3	78.1	64.3
	✓	✓	Two-Token w/ BE [58]	75.1	35.8	48.0	41.9	87.7	82.9	80.0	81.5	81.5	65.7
	✓	✓	Two-Token w/ FAME [14]	75.0	45.5	51.5	48.5	87.3	77.4	81.1	82.6	80.4	<u>69.2</u>
Disentangle	✓	✓	DEVIAS	77.3	50.3	58.8	54.6	82.0	76.5	75.7	77.1	76.4	**70.8**

Table 4: Action and scene recognition performance on the HVU dataset. We report the Top-1 accuracy (%) in both seen and unseen scenarios. We also report the harmonic mean (H.M.) of the action and scene performance. The **best** and the <u>second-best</u> H.M. numbers are highlighted.

Training Strategy	Supervision		Method	Action(↑)		Scene(↑)		H.M.
	Action	Scene		Seen	Unseen	Seen	Unseen	
Single-Task	✓	×	Naive Action ViT	82.5	34.9	-	-	-
	✓	×	FAME [14]	81.0	35.4	-	-	-
	×	✓	Naive Scene ViT	-	-	97.0	45.4	-
Multi-Task	✓	✓	Two-Token	80.5	34.9	98.5	45.9	<u>54.8</u>
	✓	✓	Two-Token w/ FAME [14]	81.5	34.1	98.5	47.2	<u>54.8</u>
Disentangle	✓	✓	DEVIAS	83.5	36.2	99.0	49.3	**57.2**

Token, Two-Token w/ FAME). For fine-tuning DEVIAS, we concatenate the action and scene tokens and feed the feature vector into the classification head. In Table 5, DEVIAS shows favorable performance on the downstream tasks across the temporal-biased and scene-biased tasks compared to the baselines.

4.7 Ablation Study

We conduct extensive ablation studies to validate the efficacy of each proposed module and design choices. We train all models on the UCF-101 [51] train split. For evaluating action performance, we report top-1 accuracy on the validation split of UCF-101 (seen), and SCUBA-VQGAN-CLIP (unseen). [38]. For scene performance, we report top-5 accuracy on the validation split of UCF-101 (seen) and UCF-101-Scene-only [10] (unseen). We report the harmonic mean of the four

Table 5: Downstream task performance. We report Top-1 accuracy (%). All models are pre-trained on the Kinetics-400 and then fine-tuned on the downstream datasets. SSV2 denotes the Something-Something-V2 dataset. The **best** and the second-best H.M. numbers are highlighted.

Pretraining Strategy	Method	Temporal-biased		Scene-biased		H.M.
		Diving48	SSV2	UCF-101	ActivityNet	
Single-Task	Naive Action ViT	81.5	74.2	98.5	84.4	83.8
	BE [58]	81.9	74.5	98.3	84.6	84.0
	FAME [14]	80.6	74.2	98.3	83.8	83.4
	Naive Scene ViT	73.1	71.8	92.0	73.1	76.7
Multi-Task	Two-Token	80.1	73.7	98.2	83.7	83.0
	Two-Token w/ FAME [14]	78.7	73.5	98.1	81.5	82.0
Disentangle	DEVIAS	84.4	75.2	98.4	84.5	**84.8**

performances to assess the balanced performance of action and scene recognition. Please refer to the supplementary material for more ablation experiments.

Effect of the disentangling encoder. We investigate the effect of the DE. As shown in Table 6 (a), incorporating the DE results in 13.4 points enhancement in the H.M. compared to the baseline without the DE (with the AMD).

Effect of the action mask decoder. In Table 6 (b), we investigate the effect of the AMD. Compared to the baseline without the AMD (with the DE), employing AMD shows a significant improvement of 5.9 points in the H.M.. The results in Table 6 (a) and (b), underscore the importance of employing *both DE and AMD* for effective disentanglement of action and scene representations.

Ablation study on the decoder design choices. In Table 6 (c), we investigate various decoder design choices. In the first column, 'Pixel' and 'Mask' denote an AMD reconstructing RGB pixel values and action masks, respectively. Compared to the action pixel reconstruction, the action mask reconstruction shows 2.6 points improvement. Decoding the action slot only shows the best performance (60.7%) compared to decoding the scene slot only (53.1%) and decoding both the action and scene slots (52.8%).

Effect of hyperparameters. Increasing the number of slot attention iterations improves the H.M. by 4.9 points as shown in the first and second rows in Table 6 (d). Using shared parameters for slot attention shows 5.6 points improvement compared to using separate parameters for each slot attention layer. We see a decrease in performance when using more slots, as shown in the fourth row.

Effect of slot assignment method. We examine the Hungarian matching when assigning disentangled slots as action or scene slots in Table 6 (g). We observe a superior performance of 60.7 points H.M. when training with Hungarian matching, compared to both hard assignment and greedy matching of each slot to action and scene roles.

Effect of softmax normalization axis. In Table 6 (h), we analyze the effect of the softmax normalization axis. Applying softmax normalization along the slot-axis, as opposed to the conventional key-axis normalization, results in a gain of

Table 6: Ablation study. To validate the effect of each component, we show the results on the UCF-101 dataset. In every experiment, we use a ViT backbone pre-trained on the UCF-101. We report the Top-1 accuracy (%) for the action and the Top-5 accuracy (%) for the scene recognition, along with the harmonic mean (H.M.) of the two accuracies. The **best** numbers are highlighted.

(a) Effect of disentangling encoder.

Method	Action(↑)		Scene(↑)		H.M.
	Seen	Unseen	Seen	Unseen	
w/o disentangling encoder	89.1	23.7	71.5	58.7	47.3
w/ disentangling encoder	90.1	40.1	74.0	61.0	**60.7**

(b) Effect of action mask decoder.

Method	Action(↑)		Scene(↑)		H.M.
	Seen	Unseen	Seen	Unseen	
w/o action mask decoder	90.0	31.6	73.7	59.8	54.8
w/ action mask decoder	90.1	40.1	74.0	61.0	**60.7**

(c) Ablations on decoder design choices.

Target	Action	Scene	Action(↑)		Scene(↑)		H.M.
			Seen	Unseen	Seen	Unseen	
Pixel	✓	×	89.7	36.4	72.5	61.0	58.1
Mask	✓	×	90.1	40.1	74.0	61.0	**60.7**
Mask	×	✓	89.4	28.9	74.7	61.9	53.1
Mask	✓	✓	89.3	28.6	74.6	61.6	52.8

(d) Effect of hyperparameters.

Hyperparameters			Action(↑)		Scene(↑)		H.M.
No. Slots	No. Iter.	Shared?	Seen	Unseen	Seen	Unseen	
2	2	✓	90.5	33.1	73.5	59.4	55.8
2	4	✓	90.1	40.1	74.0	61.0	**60.7**
2	4	×	89.1	33.4	70.7	57.7	55.1
4	4	✓	89.1	39.5	71.3	58.6	59.1

(g) Effect of slot assignment.

Method	Action(↑)		Scene(↑)		H.M.
	Seen	Unseen	Seen	Unseen	
Hard assign	89.4	35.1	71.1	58.3	56.4
Greedy	87.2	37.1	69.7	56.7	56.8
Hungarian	90.1	40.1	74.0	61.0	**60.7**

(h) Effect of softmax axis.

Axis	Action(↑)		Scene(↑)		H.M.
	Seen	Unseen	Seen	Unseen	
Keys	89.6	33.0	73.9	61.6	56.2
Slots	90.1	40.1	74.0	61.0	**60.7**

(i) Effect of mask extraction.

Method	Action(↑)		Scene(↑)		H.M.
	Seen	Unseen	Seen	Unseen	
FAME [14]	90.1	40.1	74.0	61.0	60.7
Segformer [65]	90.0	46.6	73.8	59.9	**63.7**

4.5 points in the H.M. The results indicate that slot attention, by competitively isolating features, significantly contributes to disentanglement.

Effect of mask extraction method. Throughout our experiments, we utilize a simple mask extraction approach without learning [14] by default. However, when using a learned segmentation method *e.g.* SegFormer [65], we observe a further improvement: 3.0 points increase in H.M. as shown in Table 6 (i).

5 Conclusions

In this paper, we tackle the under-explored problem of disentangled action and scene representation learning for holistic video understanding. We propose DEVIAS, a novel method that employs a disentangling encoder with the slot attention mechanism and action mask decoder to effectively learn disentangled action and scene representations. Through rigorous experiments, we assess both the action and scene recognition performance of DEVIAS in seen and unseen action-scene combination scenarios. DEVIAS shows favorable performance compared to the baselines. Furthermore, we demonstrate that disentangled action and scene representations are beneficial for various downstream tasks. The results showcase the effectiveness of the DEVIAS in learning disentangled action and scene representations as a single model. We believe our work provides interesting insights into the video understanding community and will inspire future advancements.

Acknowledgment. This work was supported in part by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea Government (MSIT) under grant RS-2024-00353131, RS-2021-II212068 (Artificial Intelligence Innovation Hub), and RS-2022-00155911 (Artificial Intelligence Convergence Innovation Human Resources Development (Kyung Hee University)). Additionally, it was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea Government (MSIT) (No. 2022R1F1A1070997).

References

1. Arnab, A., Dehghani, M., Heigold, G., Sun, C., Lučić, M., Schmid, C.: Vivit: A video vision transformer. In: ICCV (2021)
2. Ba, J.L., Kiros, J.R., Hinton, G.E.: Layer normalization (2016)
3. Bahng, H., Chun, S., Yun, S., Choo, J., Oh, S.J.: Learning de-biased representations with biased representations. In: ICML (2020)
4. Bertasius, G., Wang, H., Torresani, L.: Is space-time attention all you need for video understanding? In: ICML (2021)
5. Bhagat, S., Uppal, S., Yin, Z., Lim, N.: Disentangling multiple features in video sequences using gaussian processes in variational autoencoders. In: ECCV (2020)
6. Carreira, J., Zisserman, A.: Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In: CVPR (2018)
7. Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., Abbeel, P.: Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In: NeurIPS (2016)
8. Choi, J., Gao, C., Messou, J.C., Huang, J.B.: Why can't i dance in the mall? learning to mitigate scene bias in action recognition. In: NeurIPS (2019)
9. Choi, J., Sharma, G., Schulter, S., Huang, J.B.: Shuffle and attend: Video domain adaptation. In: ECCV (2020)
10. Chung, J., Wu, Y., Russakovsky, O.: Enabling detailed action recognition evaluation through video dataset augmentation. In: NeurIPS (2022)
11. Croitoru, I., Bogolin, S.V., Leordeanu, M.: Unsupervised learning from video to detect foreground objects in single images. In: CVPR (2017)
12. Denton, E.L., et al.: Unsupervised learning of disentangled representations from video. In: NeurIPS (2017)
13. Diba, A., Fayyaz, M., Sharma, V., Paluri, M., Gall, J., Stiefelhagen, R., Van Gool, L.: Large scale holistic video understanding. In: ECCV (2020)
14. Ding, S., Li, M., Yang, T., Qian, R., Xu, H., Chen, Q., Wang, J., Xiong, H.: Motion-aware contrastive video representation learning via foreground-background merging. In: CVPR (2022)
15. Ding, S., Qian, R., Xiong, H.: Dual contrastive learning for spatio-temporal representation. In: Proceedings of the 30th ACM international conference on multimedia. pp. 5649–5658 (2022)
16. Donahue, J., Hendricks, L.A., Rohrbach, M., Venugopalan, S., Guadarrama, S., Saenko, K., Darrell, T.: Long-term Recurrent Convolutional Networks for Visual Recognition and Description. In: CVPR (2015)
17. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR (2021)

18. Fabian Caba Heilbron, Victor Escorcia, B.G., Niebles, J.C.: Activitynet: A large-scale video benchmark for human activity understanding. In: CVPR (2015)
19. Fan, H., Xiong, B., Mangalam, K., Li, Y., Yan, Z., Malik, J., Feichtenhofer, C.: Multiscale vision transformers. In: ICCV (2021)
20. Feichtenhofer, C.: X3d: Expanding architectures for efficient video recognition. In: CVPR (2020)
21. Feichtenhofer, C., Fan, H., Malik, J., He, K.: Slowfast networks for video recognition. In: ICCV (2019)
22. Girdhar, R., Singh, M., Ravi, N., van der Maaten, L., Joulin, A., Misra, I.: Omnivore: A Single Model for Many Visual Modalities. In: CVPR (2022)
23. Gowda, S.N., Rohrbach, M., Keller, F., Sevilla-Lara, L.: Learn2augment: learning to composite videos for data augmentation in action recognition. In: ECCV (2022)
24. Goyal, R., Ebrahimi Kahou, S., Michalski, V., Materzynska, J., Westphal, S., Kim, H., Haenel, V., Fruend, I., Yianilos, P., Mueller-Freitag, M., et al.: The" something something" video database for learning and evaluating visual common sense. In: ICCV (2017)
25. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. In: CVPR (2022)
26. Hendrycks, D., Gimpel, K.: Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415 (2016)
27. Herzig, R., Ben-Avraham, E., Mangalam, K., Bar, A., Chechik, G., Rohrbach, A., Darrell, T., Globerson, A.: Object-region video transformers. In: CVPR (2022)
28. Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., Lerchner, A.: beta-vae: Learning basic visual concepts with a constrained variational framework. In: ICLR (2016)
29. Hsieh, J.T., Liu, B., Huang, D.A., Fei-Fei, L.F., Niebles, J.C.: Learning to decompose and disentangle representations for video prediction. In: NeurIPS (2018)
30. Ji, S., Xu, W., Yang, M., Yu, K.: 3D convolutional neural networks for human action recognition. TPAMI (2013)
31. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: CVPR (2014)
32. Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., et al.: The kinetics human action video dataset. arXiv preprint arXiv:1705.06950 (2017)
33. Kim, D., Kim, N., Kwak, S.: Improving cross-modal retrieval with set of diverse embeddings. In: CVPR (2023)
34. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: Hmdb: a large video database for human motion recognition. In: ICCV (2011)
35. Kuhn, H.W.: The hungarian method for the assignment problem. Naval research logistics quarterly (1955)
36. Kulkarni, T.D., Whitney, W.F., Kohli, P., Tenenbaum, J.: Deep convolutional inverse graphics network. In: NeurIPS (2015)
37. Lai, Z., Liu, S., Efros, A.A., Wang, X.: Video autoencoder: self-supervised disentanglement of static 3d structure and motion. In: CVPR (2021)
38. Li, H., Liu, Y., Zhang, H., Li, B.: Mitigating and evaluating static bias of action representations in the background and the foreground. In: ICCV (2023)
39. Li, Y., Li, Y., Vasconcelos, N.: Resound: Towards action recognition without representation bias. In: ECCV (2018)
40. Lin, J., Gan, C., Han, S.: TSM: Temporal Shift Module for Efficient Video Understanding. In: ICCV (2019)

41. Lin, X., Campos, V., Giro-i Nieto, X., Torres, J., Ferrer, C.C.: Disentangling motion, foreground and background features in videos. arXiv preprint arXiv:1707.04092 (2017)
42. Locatello, F., Weissenborn, D., Unterthiner, T., Mahendran, A., Heigold, G., Uszkoreit, J., Dosovitskiy, A., Kipf, T.: Object-centric learning with slot attention. In: NeurIPS (2020)
43. Long, F., Qiu, Z., Pan, Y., Yao, T., Luo, J., Mei, T.: Stand-alone inter-frame attention in video models. In: CVPR (2022)
44. Massagué, A.C., Zhang, C., Feric, Z., Camps, O.I., Yu, R.: Learning disentangled representations of videos with missing data. In: NeurIPS (2020)
45. Miech, A., Alayrac, J.B., Laptev, I., Sivic, J., Zisserman, A.: Rareact: A video dataset of unusual interactions. arxiv:2008.01018 (2020)
46. Ng, J.Y.H., Hausknecht, M., Vijayanarasimhan, S., Vinyals, O., Monga, R., Toderici, G.: Beyond Short Snippets: Deep Networks for Video Classification. In: CVPR (2015)
47. Patrick, M., Campbell, D., Asano, Y., Misra, I., Metze, F., Feichtenhofer, C., Vedaldi, A., Henriques, J.F.: Keeping your eye on the ball: Trajectory attention in video transformers. In: NeurIPS (2021)
48. Qian, R., Ding, S., Liu, X., Lin, D.: Static and dynamic concepts for self-supervised video representation learning. In: ECCV (2022)
49. Sahoo, A., Shah, R., Panda, R., Saenko, K., Das, A.: Contrast and mix: Temporal contrastive video domain adaptation with background mixing. In: NeurIPS (2021)
50. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: NeurIPS (2014)
51. Soomro, K., Zamir, A.R., Shah, M.: Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402 (2012)
52. Tong, Z., Song, Y., Wang, J., Wang, L.: Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In: NeurIPS (2022)
53. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3d convolutional networks. In: ICCV (2015)
54. Tran, D., Wang, H., Torresani, L., Ray, J., LeCun, Y., Paluri, M.: A closer look at spatiotemporal convolutions for action recognition. In: CVPR (2018)
55. Tran, L., Yin, X., Liu, X.: Disentangled representation learning gan for pose-invariant face recognition. In: CVPR (2017)
56. Villegas, R., Yang, J., Hong, S., Lin, X., Lee, H.: Decomposing motion and content for natural video sequence prediction. In: ICLR (2017)
57. Wang, J., Jiao, J., Bao, L., He, S., Liu, Y., Liu, W.: Self-supervised spatio-temporal representation learning for videos by predicting motion and appearance statistics. In: CVPR (2019)
58. Wang, J., Gao, Y., Li, K., Lin, Y., Ma, A.J., Cheng, H., Peng, P., Huang, F., Ji, R., Sun, X.: Removing the background by adding the background: Towards background robust self-supervised video representation learning. In: CVPR (2021)
59. Wang, X., Girshick, R., Gupta, A., He, K.: Non-local neural networks. In: CVPR (2018)
60. Wang, Y., Hoai, M.: Pulling actions out of context: Explicit separation for effective combination. In: CVPR (2018)
61. Wang, Y., Bilinski, P., Bremond, F., Dantcheva, A.: G3an: Disentangling appearance and motion for video generation. In: CVPR (2020)
62. Weinzaepfel, P., Rogez, G.: Mimetics: Towards understanding human actions out of context. IJCV (2021)

63. Wu, C.Y., Li, Y., Mangalam, K., Fan, H., Xiong, B., Malik, J., Feichtenhofer, C.: Memvit: Memory-augmented multiscale vision transformer for efficient long-term video recognition. In: CVPR (2022)
64. Xie, C., Xiang, Y., Harchaoui, Z., Fox, D.: Object discovery in videos as foreground motion clustering. In: CVPR (2019)
65. Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P.: Segformer: Simple and efficient design for semantic segmentation with transformers. In: NeurIPS (2021)
66. Xie, S., Sun, C., Huang, J., Tu, Z., Murphy, K.: Rethinking spatiotemporal feature learning for video understanding. In: ECCV (2018)
67. Xing, X., Gao, R., Han, T., Zhu, S.C., Wu, Y.N.: Deformable generator networks: unsupervised disentanglement of appearance and geometry. TPAMI (2020)
68. Yan, S., Xiong, X., Arnab, A., Lu, Z., Zhang, M., Sun, C., Schmid, C.: Multiview transformers for video recognition. In: CVPR (2022)
69. Zhang, Z., Tran, L., Yin, X., Atoum, Y., Liu, X., Wan, J., Wang, N.: Gait recognition via disentangled representation learning. In: CVPR (2019)
70. Zhao, Y., Xiong, Y., Lin, D.: Recognize actions by disentangling components of dynamics. In: CVPR (2018)
71. Zhou, B., Andonian, A., Oliva, A., Torralba, A.: Temporal Relational Reasoning in Videos. In: ECCV (2018)
72. Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., Torralba, A.: Places: A 10 million image database for scene recognition. TPAMI (2017)
73. Zou, Y., Choi, J., Wang, Q., Huang, J.B.: Learning representational invariances for data-efficient action recognition. Computer Vision and Image Understanding (2023)