Hierarchically Decoupled Spatial-Temporal Contrast for Self-supervised Video Representation Learning

Zehua Zhang David Crandall
Indiana University Bloomington
{zehzhang, djcran}@indiana.edu

Abstract

We present a novel technique for self-supervised video representation learning by: (a) decoupling the learning objective into two contrastive subtasks respectively emphasizing spatial and temporal features, and (b) performing it hierarchically to encourage multi-scale understanding. Motivated by their effectiveness in supervised learning, we first introduce spatial-temporal feature learning decoupling and hierarchical learning to the context of unsupervised video learning. We show by experiments that augmentations can be manipulated as regularization to guide the network to learn desired semantics in contrastive learning, and we propose a way for the model to separately capture spatial and temporal features at multiple scales. We also introduce an approach to overcome the problem of divergent levels of instance invariance at different hierarchies by modeling the invariance as loss weights for objective re-weighting. Experiments on downstream action recognition benchmarks on UCF101 and HMDB51 show that our proposed Hierarchically Decoupled Spatial-Temporal Contrast (HDC) makes substantial improvements over directly learning spatial-temporal features as a whole and achieves competitive performance when compared with other state-of-the-art unsupervised methods. Code will be made available.

1. Introduction

As a solution to the growing need for large-scale labeled data for training complex neural network models [6, 20, 35, 56, 58], unsupervised representation learning aims to learn good feature embeddings from data without annotations. Using the learned representations as initialization, downstream tasks only need to be fine-tuned on a relatively small labeled dataset in order to yield reasonable performance. Much recent progress in unsupervised image representation learning [3, 19, 21, 25, 44, 46, 57, 73, 78] is driven by contrastive learning. While they solve the same pretext task of instance-level variant matching, these methods differ in how they obtain variant embeddings of the same instance, e.g., using augmentations [3, 25, 73, 78], future representations [46], or momentum features [19]. By optimizing a contrastive loss [14], they are essentially maximizing intra-instance embedding similarity and minimizing inter-instance embedding similarity, which leads to a spread-out

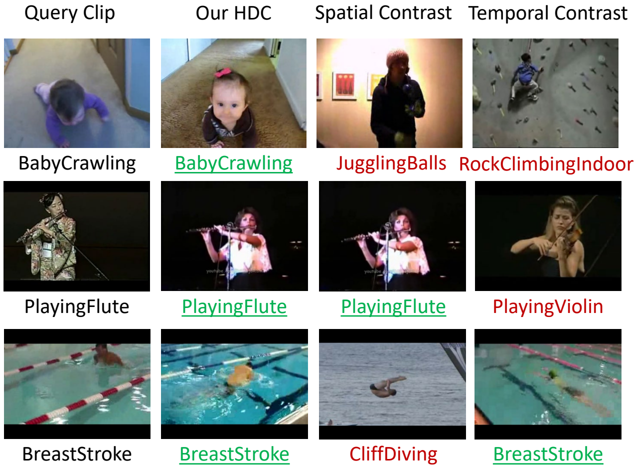

Figure 1: Sample results of nearest neighbor retrieval on UCF101 split 1 demonstrate the effectiveness of our approach for decoupling spatial-temporal feature learning. We compare the top-1 retrieved clip from our full model (*HDC*) with the clip retrieved by models performing only one subtask of HDC (Spatial Contrast (*SC*) or Temporal Contrast (*TC*)). Below each clip is its corresponding action category, with green underlined text indicating a correct retrieval and red text indicating an incorrect retrieval. During retrieval, SC tends to focus more on spatial similarity with the query clip, while TC focuses more on temporal similarity. For example, in the first row, SC's retrieval is perhaps based on spatial similarity of people indoors, while TC's retrieval may be due to temporal similarity of crawling. Since our proposed HDC solves SC and TC hierarchically and simultaneously, it is able to capture both spatial and temporal semantics and made the correct retrieval.

feature space [80].

Compared with the success of contrastive learning for images, most state-of-the-art unsupervised video representation learning work relies on context-based proxy tasks such as time arrow classification [72] or clip order prediction [74]. While the results of some recent methods [15,16,33,55] suggest the potential of contrastive learning for videos, they work in a "one for all" manner by solving a single contrastive learning task with the activation maps from the final layer, expecting the model to capture all the features through the learning procedure. As a result, the learned model may lack a general understanding of spatial and temporal semantics, and instead just memorize spatial-temporal combinations.

In this paper, we argue that good video representations should be able to capture spatial and temporal features in a more general form at multiple scales, and thus it is helpful to decompose the overall goal of learning spatial-temporal features into hierarchical subtasks, respectively emphasizing spatial and temporal features. To this end, we present Hierarchically Decoupled Spatial-Temporal Contrast (HDC), in which we decouple the learning objective into separate subtasks of Spatial Contrast and Temporal Contrast and perform the learning hierarchically.

Neural networks are notorious for learning shortcuts to "cheat" [11, 72]: if there is an easy way to solve the problem, the network will hardly try to find a more complex solution. We make use of this property for the decoupling and direct the network to learn different features by manipulating augmentations. In particular, in the Spatial Contrast subtask, we deliberately provide a shortcut by creating augmented variants with only spatial augmentations (random spatial cropping, color jittering, etc.). As the timestamps of the query clip and its augmented copy are the same, it is possible to solve the matching task based merely on consistency of spatial semantics, and thus the network will try to "cheat" by focusing more on spatial features. In Temporal Contrast learning, we randomly select a new clip from the video of the query clip (random temporal cropping) before applying spatial augmentations in order to obtain a variant whose spatial semantics are as different from the query clip as possible. Since spatial context may vary dramatically after applying the temporal and spatial transformations, the model is prevented from "cheating" through spatial similarity and encouraged to rely more on similarity of temporal semantics to solve the pretext task. Fig. 1 shows nearest neighbor retrieval results demonstrating the effectiveness of our approach for the decoupling, and suggests the potential of directing the network to learn desired features through manipulating augmentations as regularization.

In order to capture multi-scale features, we further perform Spatial Contrast and Temporal Contrast learning hierarchically by optimizing towards a compound loss. During hierarchical learning, we model the significance of instance-wise consistency in a given layer with different weights, because features from different layers do not share the same level of invariance against augmentations [79].

In summary, our contributions are as follows:

- We demonstrate the effectiveness of spatial-temporal feature learning decoupling and hierarchical learning in the context of unsupervised learning for the first time.
- We show that the network can be guided to learn desired features in contrastive learning by manipulating augmentations as regularization, and introduce a new way for the network to separately capture spatial and temporal semantics in self-supervised video representation learning.
- We propose an approach to improve hierarchical contrastive learning by modeling the divergent levels of invariance in different layers as different loss weights.
- By optimizing a novel compound loss, our Hierarchically Decoupled Spatial-Temporal Contrast (HDC) outperforms other unsupervised methods and sets a new state-of-the-art on downstream tasks of action recognition on UCF101 and HMDB51.

2. Related Work

Unsupervised Video Representation Learning was originally based on input reconstruction [23, 24, 37, 38, 50, 62], while more recent methods derive implicit pseudo-labels from the unlabeled data to use as self-supervision signals for the corresponding pretext task [49, 70]. For example, several models use chronological order of video frames to define proxy tasks such as frame order prediction [39] or verification [13, 45], clip order prediction [74], and time arrow classification [72]. Other pretext tasks, such as spatial-temporal jigsaw [32], future prediction [15, 40, 42, 54, 63, 64], temporal correspondence estimation [28, 29, 70, 71], audio-video clustering [2], video colorization [65], motion and appearance statistics prediction [67], and loss distillation across multiple tasks [47] have also been explored.

Contrastive Learning [14] is very effective for unsupervised representation learning for images [3, 19, 21, 25, 44, 46, 57, 73, 78]. These methods try to learn a feature space in which variants of the same sample are close together while variants from different samples are far apart, and mainly differ in how they create the variants. For example, Oord *et al.* [46] predict the future in the latent space as a variant of the real embeddings of the future. In the video domain, Han *et al.* [15] predict dense feature maps of future clips and match them with corresponding real embeddings from other distractions. This idea is further extended to a memory-augmented version [16] for improvement. Sun *et al.* [55] proposed to use bidirectional transformers for multi-modal contrastive learning from text and videos. A cyclecontrastive loss inspired by CycleGAN [82] is presented by Kong *et al.* [33] to use the relationship between videos and frames. Recasens *et al.* [51] and Wang *et al.* [66] both explore a new pretext task of matching short-term view to long-term view of the same video. Tschannen *et al.* [60] also use video-induced invariance to formulate a pretext task based on contrastive learning. Despite its motivation, interestingly, it aims to learn image representations instead of video embeddings. There is more discussion in Sec. 3.4.

3. Our Method

To explore the potential of self-supervised video representation learning merely from RGB clips, we formulate a novel pretext task of Hierarchically Decoupled Spatial-Temporal Contrast (HDC) in which we maximize intra-instance representation similarity and minimize inter-instance representation similarity spatially (Fig. 2), temporally (Fig. 2), and hierarchically (Fig. 3).

3.1. Decoupled Contrast

Motivated by the observation in supervised learning that factoring 3D filters into separate spatial and temporal components yields significant gains [59], we propose to decouple the overall objective of unsupervised spatial-temporal feature learning into separate subtasks and provide regularizations to guide them to emphasize spatial and temporal features, respectively (Fig. 2).

Spatial Contrast is designed to focus on learning spatial representations. Neural networks are notorious for learning shortcuts to "cheat" [11, 72]. While previous work tries to avoid all cheating [11, 32, 39, 72], we intentionally make use of this property. In Spatial Contrast, the network is provided with a shortcut by augmenting clips with only spatial transformations. This allows the model to "cheat" by using only spatial features to capture intra-instance similarity and inter-instance difference. In other words, the network will not bother with temporal features because capturing spatial similarity is already enough for solving the matching task.

In particular, as shown in Fig. 2, given a batch of clips $X = [x_1, x_2, ..., x_B]$, each from a different video, we augment each clip i with augmentations $\varphi_{o,i}$ and $\varphi_{s,i}$ sampled from families of candidate spatial transformations Φ_o and Φ_s to obtain corresponding variants $u_{o,i} = \varphi_{o,i}(x_i)$, $\varphi_{o,i}(\cdot) \sim \Phi_o$ and $u_{s,i} = \varphi_{s,i}(x_i)$, $\varphi_{s,i}(\cdot) \sim \Phi_s$, which are only spatially-augmented.

Our primary goal is to train an encoder so that the similarity between feature embeddings of $u_{o,i}$ and $u_{s,j}$ is maximized when $i = j$, and minimized otherwise. Let $f(\cdot)$ denote the encoder. The embeddings of $u_{o,i}$ and $u_{s,j}$ are $v_{o,i} = f(u_{o,i})$ and $v_{s,j} = f(u_{s,j})$, where $v_{o,i}$ and $v_{s,j}$ are both vector embeddings, e.g., obtained through global pooling. By measuring similarity with cosine distance $sim(v_{o,i}, v_{s,j}) = (v_{o,i} \cdot v_{s,j})/(|v_{o,i}| \cdot |v_{s,j}|)$, our goal can be achieved by optimizing a contrastive loss called InfoNCE [46],

$$L_s = -\sum_{i=1}^{B} \log \frac{\exp(sim(v_{o,i}, v_{s,i})/\tau)}{\sum_{j=1}^{B} \exp(sim(v_{o,i}, v_{s,j})/\tau)}, \quad (1)$$

where τ is a temperature controlling the concentration of the feature embedding distribution [22] (usually around 0.1). Eq. 1 can be viewed as a cross entropy loss between a pseudo prediction $S \in R^{B\times B}$ and a pseudo label $\bar{S} \in R^{B\times B}$. $\bar{S}$ is an identity matrix and S is a similarity matrix where S_{ij} measures the similarity of one clip variant $u_{o,i}$ to another variant $u_{s,j}$ in the feature space,

$$S_{ij} = \frac{\exp(sim(v_{o,i}, v_{s,j})/\tau)}{\sum_{m=1}^{B} \exp(sim(v_{o,i}, v_{s,m})/\tau)}. \quad (2)$$

In this spirit, the Spatial Contrast subtask is a self-supervised multi-way classification problem in which we want to match one clip variant to another when they come from the same video and distinguish it from variants coming from different videos.

Temporal Contrast emphasizes learning temporal representations. To do this, we need to prevent the network from cheating through spatial feature similarity: we want variants whose spatial context varies dramatically from the original clip, but whose temporal context remains nearly the same and thus is essential for capturing instance-level invariance. To do this, we add temporal augmentations of random temporal cropping before applying spatial transformations to produce another temporally-augmented variant. Note that spatial augmentations are crucial here to further alter spatial context in order to prevent cheating, and we show its importance in Section 4.1.

Specifically, to produce a temporally-augmented variant $u_{t,j}$ for each clip x_j, random temporal cropping $\Gamma(\cdot)$ is first applied, followed by spatial augmentations $\varphi_{t,j}$ sampled from another family of candidate spatial transformations Φ_t, after which we have $u_{t,j} = \varphi_{t,j}(\Gamma(x_j)),\ \varphi_{t,j}(\cdot) \sim \Phi_t$. Temporal Contrast is modeled between $u_{o,i}$ and $u_{t,j}$ using a similar technique as in Spatial Contrast. Then we minimize,

$$L_t = -\sum_{i=1}^{B} \log \frac{\exp(sim(v_{o,i}, v_{t,i})/\tau)}{\sum_{j=1}^{B} \exp(sim(v_{o,i}, v_{t,j})/\tau)}. \quad (3)$$

As with Spatial Contrast learning, the Temporal Contrast subtask is also a self-supervised multi-way classification problem of matching clip variants of the same video.

3.2. Hierarchical Contrast

Inspired by the effectiveness of multi-scale features in supervised learning [26, 81], we introduce hierarchical learning to unsupervised learning by conducting Decoupled Contrast learning hierarchically. As illustrated in Fig. 3,

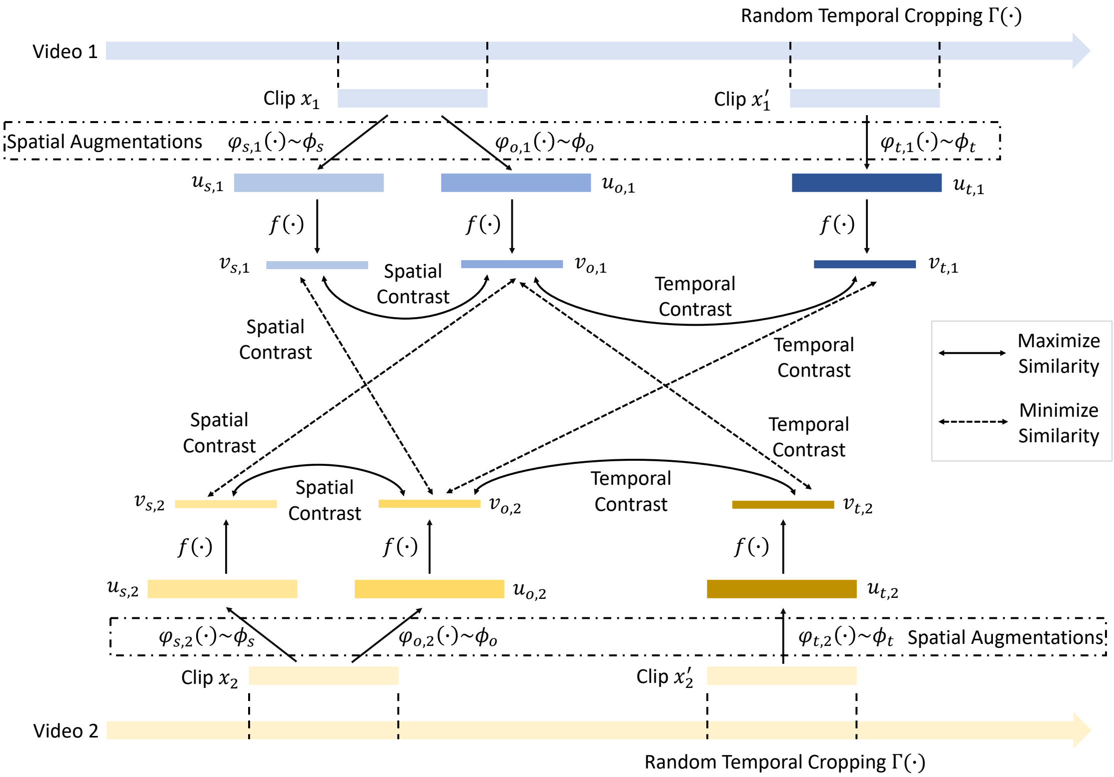

Figure 2: A simple example of how our Decoupled Contrast works between two sample videos. In Spatial Contrast (SC), two sets of spatial transformations, $\varphi_{o,i}$ and $\varphi_{s,i}$, are sampled from families of candidate spatial transformations Φ_o and Φ_s, and applied to the same clip x_i to obtain variants $u_{o,i}$ and $u_{s,i}$ for SC learning. In Temporal Contrast (TC), we first extract another clip x_i' from the same video as x_i by random temporal cropping $\Gamma(\cdot)$, and then apply $\varphi_{t,i}$ sampled from families of candidate spatial transformations Φ_t to obtain a variant $u_{t,i}$, which will be used with $u_{o,i}$ for TC learning.

feature maps from different layers or blocks of the encoder $f(\cdot)$ are collected and pooled to produce multi-scale vector embeddings. Two pooling strategies are applied to fully leverage instance-level consistency among different layers: (1) Temporal Contrast uses 3D global average pooling along both temporal and spatial dimensions, and (2) Spatial Contrast performs 2D global average pooling only along the spatial dimension. Therefore, for each scale, we obtain one vector representation for each clip variant in Hierarchical Temporal Contrast learning, but may get more than one vector for each variant in Hierarchical Spatial Contrast learning, depending on the length of input clip and the temporal downsample factor of the encoder for a certain layer. Our motivation is that because the timestamps in Spatial Contrast are the same for two clip variants from the same video, the spatial instance-level invariance should exist not only for the whole clip, but also for corresponding sub-clips.

As shown in Fig. 3, for Hierarchical Spatial Contrast learning at scale k, we apply 2D global average pooling and obtain vector embeddings of $v_{o,i}^{k,n}$, $n = 1, 2, ..., N$ for clip variant $u_{o,i}$ and vector embeddings of $v_{s,j}^{k,n}$, $n = 1, 2, ..., N$ for variant $u_{s,j}$, where N is the size of the temporal dimension of the feature maps. The loss at scale k becomes,

$$L_s^k = -\frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{B}\log\frac{\exp(sim(v_{o,i}^{k,n}, v_{s,i}^{k,n})/\tau)}{\sum_{j=1}^{B}\exp(sim(v_{o,i}^{k,n}, v_{s,j}^{k,n})/\tau)}, \tag{4}$$

while the loss for Hierarchical Temporal Contrast remains almost the same for scale k,

$$L_t^k = -\sum_{i=1}^{B}\log\frac{\exp(sim(v_{o,i}^k, v_{t,i}^k)/\tau)}{\sum_{j=1}^{B}\exp(sim(v_{o,i}^k, v_{t,j}^k)/\tau)}, \tag{5}$$

where $v_{o,i}^k$ are $v_{t,i}^k$ are obtained from 3D global average pooling.

3.3. Self-supervised Learning with HDC

We consider 3 different backbones, C3D [58], 3D-ResNet18 [18, 20] and R(2+1)D-10 [59], as our encoder $f(\cdot)$ for fair comparison with existing methods. All fully-connected layers are removed (last 3 layers of C3D, and last layer of 3D-ResNet18 and R(2+1)D-10). Batch normalization layers [27] in 3D-ResNet18 and R(2+1)D-10 are replaced with instance normalization layers [61] to prevent

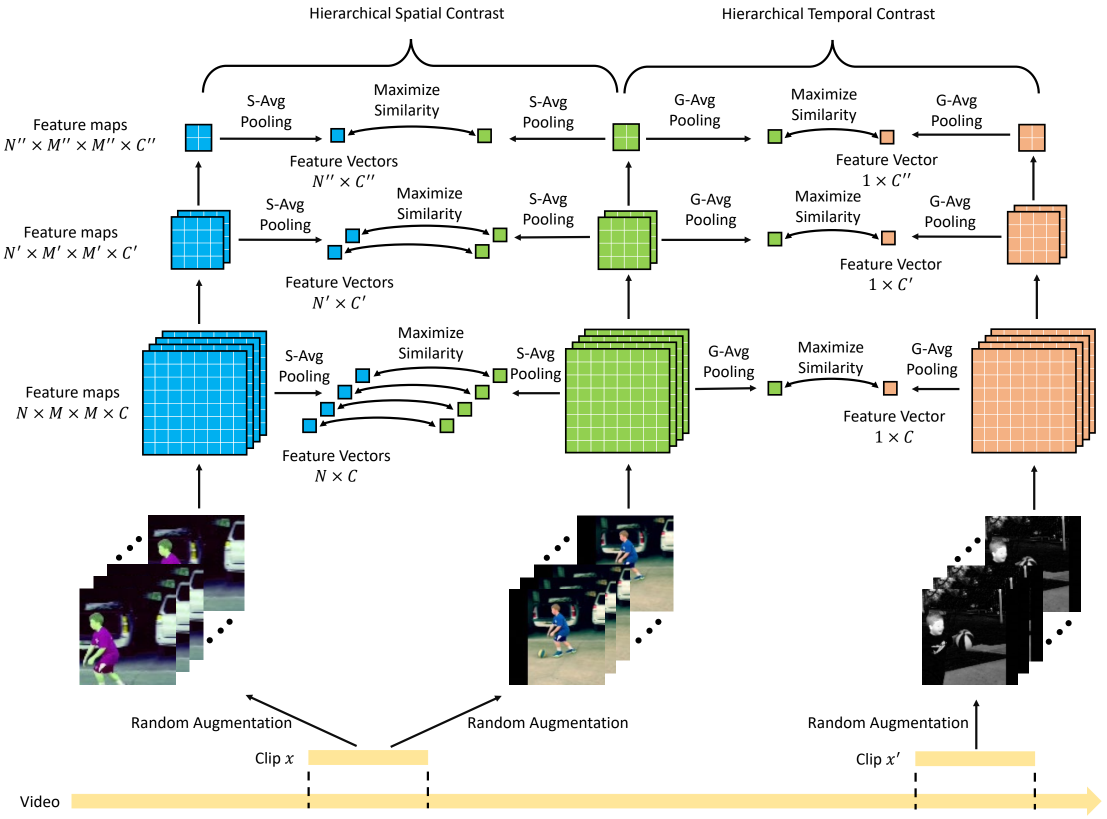

Figure 3: An example illustrating Hierarchical Contrast. Here we only show learning among variants from the same video and thus the network is maximizing similarity between all corresponding pairs. Otherwise if variants are from different videos, the network will minimize similarity. Feature maps at multiple scales are 2D global average pooled along the spatial dimension (*S-Avg Pooling*) in Hierarchical Spatial Contrast learning, and 3D global average pooled along the temporal and spatial dimensions (*G-Avg Pooling*) in Hierarchical Temporal Contrast learning. The channel dimension is omitted for clarity.

cheating via batch statistics [19, 21]; for C3D, an instance normalization layer is inserted after each convolution layer. For multi-scale learning, we consider feature maps from blocks $3, 4, 5$ (last 3 blocks). Instead of directly performing the matching pretext task with the feature vectors from pooling, we project each vector to a lower-dimension space with linear projections, following [73]. In particular, there are projections $g_o^k(\cdot)$, $g_s^k(\cdot)$, and $g_t^k(\cdot)$ for projecting $v_{o,i}^k$, $v_{s,i}^k$, and $v_{t,i}^k$, respectively, at each scale k, each implemented as a single fully-connected layer with linear activation projecting a vector into 128-d. (See [7] for more experiments and discussion about the importance of such linear projections.) The families of spatial augmentations Φ_o, Φ_s, and Φ_t contain the same set of transformations of random spatial cropping, scale jittering, horizontal flipping, color jittering, and channel replication [39]. We use $\tau = 0.07$ for computing the contrastive loss.

We perform self-supervised training on Kinetics-400 [31], which has 400 action classes and over 400 videos per class. We resize frames, preserving aspect ratio, so that the shorter side is 128 pixels. Each mini-batch contains 128 clips from 128 videos and each clip consists of 16 randomly-cropped continuous frames of shape $128 \times 128 \times 3$. After spatial and temporal augmentations, three sets of clip variants are obtained, each of shape $128 \times 16 \times 112 \times 112 \times 3$. Before being fed to the encoder, these clips are rescaled following [6] so pixel values are between -1 and 1.

Our model is implemented with Tensorflow [1] and Keras [9]. We use stochastic gradient descent with learning rate 0.1, momentum 0.9, decay 0.0001, and $L2$ regularizer $5e^{-5}$. HDC is trained as a whole towards minimizing a novel compound contrastive loss, namely HD-NCE,

$$L = \sum_k (\alpha_k \cdot L_s^k + \beta_k \cdot L_t^k), \tag{6}$$

where L_s^k and L_t^k are in Eqs. 4 and 5, and $k = 3, 4, 5$ as we use features from blocks $3, 4, 5$ of the encoder.

Significance of instance-wise consistency. α_k and β_k represent the significance of instance-wise consistency in each layer, essentially weighting how much the corresponding subtask will contribute to our main goal. We simply tested (1) $\alpha_3 = \beta_3 = \alpha_4 = \beta_4 = \alpha_5 = \beta_5 = 1.0$, and (2)

$\alpha_3 = \beta_3 = 0.25$, $\alpha_4 = \beta_4 = 0.5$ and $\alpha_5 = \beta_5 = 1.0$, and the second set worked better (see Section 4.1).

3.4. Difference from Recent Work

Some work [75,76] also explores hierarchical contrastive learning in videos. However, while HDC matches variants obtained with carefully designed augmentations, [75] brings closer features from the slow and fast streams of a Slowfast network [12], and [76] predicts future motion patterns. Moreover, our work introduces spatial-temporal learning decoupling, which is another novel component.

Another similar method is CVRL [48], which is similar in spirit to the variant of our model trained with only Temporal Contrast incorporated in the last layer. Our HDC decouples the learning into separate subtasks emphasizing spatial and temporal features and further performs the learning hierarchically. As later experiments (Sec. 4.1 and Tab. 1) show, HDC achieves significant improvement over the CVRL-alike variant (the second row in Tab. 1).

4. Experiments

We follow a common protocol [45] to evaluate the effectiveness of our HDC by using the learned representations as initialization and fine-tuning on the downstream task of action recognition on UCF101 [52] and HMDB51 [36]. UCF101 [52] consists of 13,320 videos and 101 classes of human action. It has three train/test splits with a split ratio of about 7:3. HMDB51 [36] is another widely-used action recognition dataset containing 6,766 videos and 51 classes. It also has three splits with a similar split ratio as UCF101. In our ablation studies, if not explicitly mentioned, we report Top-1 accuracy on only UCF101 split 1. When comparing our method with other state-of-the-art, results averaged on three splits of UCF101 and HMDB51 are reported.

In fine-tuning, we use the same network (C3D, 3D-ResNet18, or R(2+1)D-10) as we did in self-supervised learning. A dropout layer [53] of ratio 0.5 is added after global average pooling, followed by a single fully-connected layer and softmax activation for classification. The instance normalization layers are kept as they are. Blocks 1-5 are initialized with the learned weights by self-supervised training on Kinetics-400. The last layer is randomly initialized. During fine-tuning, we use stochastic gradient descent with learning rate 0.01, momentum 0.9, decay 0.0001, and $L2$ regularizer $5e^{-5}$. Each mini-batch contains 32 clips, each with 16 continuous frames randomly cropped to $128 \times 128 \times 3$. Augmentations including random spatial cropping, scale jittering, and horizontal flipping are applied, resulting in an input of shape $32 \times 16 \times 112 \times 112 \times 3$.

During testing, each video is divided into non-overlapping 16-frame clips. A center crop and four corner crops are taken for each clip [69]. The class score for each video is obtained by averaging over all crops and clips.

SC	TC	α_k	β_k	Top-1 Acc
5	-	1	-	61.3
-	5	-	1	62.7
5	5	1	1	65.5
4, 5	4, 5	1, 1	1, 1	66.9
4, 5	4, 5	0.5, 1	0.5, 1	67.8
3, 4, 5	3, 4, 5	1, 1, 1	1, 1, 1	66.8
Training from scratch		-	-	43.5
3, 4, 5	3, 4, 5	0.25, 0.5, 1	0.25, 0.5, 1	**69.0**

Table 1: Ablation study of decoupled contrast and hierarchical contrast. We use 3D-ResNet18 as the backbone, pretrain on Kinetics-400 and report top-1 accuracy on UCF101 split 1. SC and TC indicate the scale at which we perform Spatial Contrast (SC) or Temporal Contrast (TC) self-supervised learning – i.e., the index of the block we take features from. α_k and β_k are defined in Eq. 6 (listed in order of scale).

4.1. Ablation Studies

We conduct ablation experiments to analyze our design choices.

Decoupled Contrast. We first evaluate the effectiveness of decomposing the video representation learning task into subtasks of Spatial and Temporal Contrast and performing joint learning with them in Tab. 1. The first three rows present the results with only Spatial Contrast learning, with only Temporal Contrast learning, and with both Spatial Contrast and Temporal Contrast learning. Hierarchical Contrast learning is not involved and the learning is based on the features of the last (5th) block. We see that, first, Temporal Contrast achieves slightly better performance than Spatial Contrast, perhaps because temporal semantics are more important in video learning. Then, by incorporating both Spatial and Temporal Contrast, performance improves significantly. This indicates that Spatial Contrast and Temporal Contrast learn complementary features, and validates our hypothesis that applying different augmentations as a way of regularization guides the network to learn different features. Furthermore, the nearest neighbor retrieval results in Fig. 1 and Fig. 4 show that Spatial Contrast and Temporal Contrast focus on spatial and temporal features respectively.

We note that Spatial Contrast in and of itself can be viewed as directly migrating the basic contrastive learning model from the image to video domain, while Temporal Contrast learning adapts to this new domain by further applying temporal augmentations to create the sample variant. Thus the results suggest that traditional contrastive learning models should be further adapted to learn a good embedding space for video, and we present one possible solution.

color jittering	channel replication	flipping	scale jittering	Top-1 Acc
✗				65.4
	✗			61.0
✗	✗			49.4
		✗		68.0
			✗	64.7
Our full model (HDC)				**69.0**

Table 2: Ablation study of different spatial augmentations. We use our full HDC model with 3D-ResNet18 as the backbone, pretrain on Kinetics-400, and show top-1 accuracy on UCF101 split 1. An ✗ indicates the transformation is not used during pretraining to generate augmented variants.

Hierarchical Contrast. Comparing rows 3 with 4 or 6 of Tab. 1, we observe that Hierarchical Contrast learning at more scales yields better performance. This suggests that instance-level invariance widely exists for mid-level and high-level features from previous layers, and can be used to capture multi-scale semantics. However, rows 4 and 6 show that simply adding more scales does not consistently bring improvement. We argue that this is because instance-level invariance may be weaker for mid-level features, and adding more scales while giving them the same weight of significance will distract the network and harm the learning. We discuss this below.

Significance of Instance-level Invariance at Different Scales. Zeiler et al. [79] showed that early layers of neural networks learn low-level features which change a lot due to augmentations. As we perform Hierarchical Contrast learning at more scales, those mid-level features may not share the same level of invariance against augmentations as the last layer's features. We use α_k and β_k to model the significance of instance-level invariance at different scales. We conducted experiments with different values of α_k and β_k, and the results (row 4 vs row 5, row 6 vs the last row in Tab. 1) show that smaller weights for lower levels of the hierarchy yields better performance than assigning 1's. This suggests that significance decreases at lower layers, which is consistent with [79]. We did not exhaustively tune α_k and β_k, so better performance is likely possible through tuning.

Spatial augmentation ablations. Tab. 2 shows ablation results of using different spatial augmentations. We find that channel replication is crucial for the model to learn good features, perhaps because it is a non-linear projection of RGB channels which can effectively prevent the network from learning trivial solutions based on color distribution [39]. As another way to prevent such trivial solutions, color jittering uses a linear function and thus is less effective. However, when neither color jittering nor channel replication is used, the accuracy drops greatly, indicating the network may suffer from the trivial solution.

We note that, as shown in [7], there are other spatial augmentations which can further improve the performance of contrastive learning. However, the purpose of this paper is not to exhaustively explore effects of different augmentations. By applying a set of simple augmentations, we show the generalizability of our HDC.

Self-supervised Learning		Accuracy (%)	
Method	Architecture	HMDB51	UCF101
Random Initialization [45]	CaffeNet	13.3	38.6
Shuffle & Learn [45]	CaffeNet	18.1	50.2
Büchler et al. [5]	CaffeNet	25.0	58.6
Random Initialization [39]	VGG-M-2048	18.3	51.1
OPN [39]	VGG-M-2048	23.8	59.8
Random Initialization [74]	R3D-18	21.5	54.4
Clip Order [74]	R3D-18	29.5	64.9
VCP [41]	R3D-18	31.5	66.0
Random Initialization [33]	R3D-18+1	19.4	44.7
CCL [33]	R3D-18+1	37.8	69.4
Random Initialization [15]	2D3D-ResNet18	17.1	46.5
DPC [15]	2D3D-ResNet18	34.5	68.2
Random Initialization [4]	I3D	**29.6**	47.9
SpeedNet [4]	I3D	**43.7**	66.7
Random Initialization [18]	3D-ResNet18	17.1 (19.1)	42.4 (43.8)
3D-RotNet [30]	3D-ResNet18	33.7	62.9
3D-ST-Puzzle [32]	3D-ResNet18	33.7	65.8
Our method (HDC†)	**3D-ResNet18**	**38.4**	**69.8**
Our method (HDC)	**3D-ResNet18**	**38.1**	**68.5**
Random Initialization [74]	C3D	23.2 (22.5)	61.6 (51.4)
Motion & Appearance [67]	C3D	33.4	61.2
Clip Order [74]	C3D	28.4	65.6
VCP [41]	C3D	32.5	68.5
Cho et al. [8]	C3D	34.3	70.4
Our method (HDC†)	**C3D**	**39.5**	**73.1**
Our method (HDC)	**C3D**	**39.3**	**72.3**
Random Initialization [74]	R(2+1)D-10	22.0	56.2
Clip Order [74]	R(2+1)D-10	30.9	72.4
VCP [41]	R(2+1)D-10	32.2	66.3
Cho et al. [8]	R(2+1)D-10	36.8	74.8
PacePrediction [68]	R(2+1)D-10	36.6	**77.1**
Our method (HDC†)	**R(2+1)D-10**	**40.0**	**76.8**
Our method (HDC)	**R(2+1)D-10**	**39.8**	**76.2**

Table 3: Top-1 accuracy averaged on 3 splits of UCF101 and HMDB51. Parentheses show accuracy obtained with our own implementation. HDC with † is pretrained by decreasing learning rate more slowly.

Spatial augmentations in Temporal Contrast. We verify the importance of applying spatial augmentations after temporal random cropping in Temporal Contrast. We use our full model of HDC with C3D as the backbone, pretrain on Kinetics-400, and report average accuracy on 3 splits of UCF101 and HMDB51. It achieves 72.3% on UCF101 and 39.3% on HMDB51 when spatial augmentations are used in Temporal Contrast versus only 68.9% on UCF101 and 38.0% on HMDB51 when spatial augmentations are not used. This supports our design choice of having spatial augmentations in Temporal Contrast to obtain variants whose spatial context varies as much as possible.

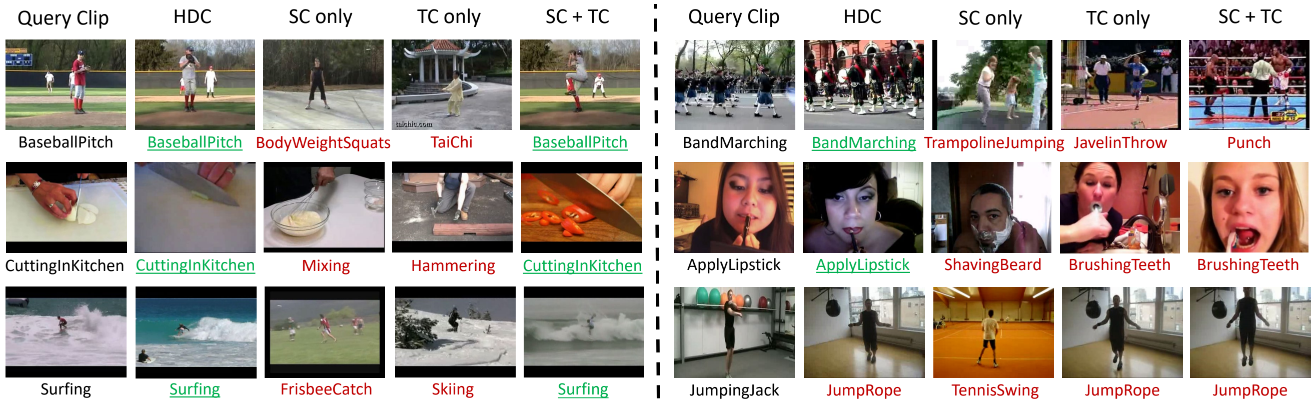

Figure 4: Sample results of nearest neighbor retrieval on UCF101 split 1, showing the top-1 retrieved clip from our full model (*HDC*), and Spatial Contrast (*SC*), Temporal Contrast (*TC*), and Joint Spatial-Temporal Contrast (*SC + TC*) models. Below each clip is its action category, with green underlined text indicating correct retrievals and red text indicating incorrect. Our proposed HDC achieved better results because of the ability to capture both spatial and temporal features at multiple scales.

Methods	Top1	Top5	Top10	Top20	Top50
OPN [39]	19.9	28.7	34.0	40.6	51.6
Büchler et al. [5]	25.7	36.2	42.2	49.2	59.5
Random Initialized C3D	16.7	27.5	33.7	41.4	53.0
Clip Order (C3D) [74]	12.5	29.0	39.0	50.6	66.9
VCP (C3D) [41]	17.3	31.5	42.0	52.6	67.7
CCL (R3D-18+1) [33]	22.0	39.1	44.6	56.3	70.8
Our HDC (C3D)	**33.9**	**49.6**	**55.7**	**61.6**	**69.9**

Table 4: Nearest neighbor retrieval results on UCF101.

Methods	Top1	Top5	Top10	Top20	Top50
Random Initialized C3D	7.4	20.5	31.9	44.5	66.3
Clip Order (C3D) [74]	7.4	22.6	34.4	48.5	70.1
VCP (C3D) [41]	7.8	23.8	35.3	**49.3**	**71.6**
Our HDC (C3D)	**14.6**	**28.8**	**36.1**	**44.8**	**57.9**

Table 5: Nearest neighbor retrieval results on HMDB51.

4.2. Comparison with the State-of-the-art

For fairness, we compare with methods under similar settings. There are other methods that achieve great success. But they adopt much more advanced backbones [10,48,77], require extra preprocessing [16,17,72], or needs additional information to prepare the input [2,34,43,55].

We report top-1 accuracy averaged over 3 splits of UCF101 and HMDB51 in Tab. 3. HDC achieves better or comparable performance on both datasets. We note that: 1) PacePrediction [68] shows slightly better results on UCF101, perhaps because its best performance is achieved by further adapting contrastive laerning in addition to pace prediction; 2) SpeedNet [4] benefits from the backbone (29.6% when trained from scratch) and thus has better performance on HMDB51.

By comparing HDC variants using different backbones, we observe that a more advanced backbone always leads to better performance, suggesting that HDC will be able to further benefit from future advances in network architectures. We also find that decreasing the learning rate more slowly is beneficial, perhaps because it allows better optimization.

4.3. Nearest Neighbor Retrieval

We follow [74] and perform nearest neighbor retrieval experiments. As shown in Tables 4 and 5, our method significantly outperforms other methods on both UCF101 and HMDB51. This implies that we learn better features, and explains the good performance on downstream tasks.

As shown in Fig. 1 and Fig. 4, qualitative results further support our idea of manipulating augmentations to guide the network to learn different features and the benefit of hierarchical learning. For example, in the first row of the right column in Fig. 4, HDC succeeds in retrieving a clip of the correct action while the other variants fail: SC focuses more on spatial information of people outdoors, TC pays more attention to the actions of people walking, and ST+TC fails perhaps because it does not learn features at different hierarchies, although it successfully captured both spatial and temporal information of multiple moving people.

5. Conclusion

We considered the problem of unsupervised video representation learning, and introduced Hierarchically Decoupled Spatial-Temporal Contrast (HDC). By decomposing the target into subtasks emphasizing different features and performing learning in a hierarchical manner, HDC is able to capture both rich spatial and temporal semantics at multiple scales. Extensive experiments of action recognition and nearest neighbor retrieval on UCF101 and HMDB51 using 3 different backbones suggest the potential of manipulating augmentations as regularization and demonstrate the state-of-the-art performance of HDC.

References

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek G. Murray, Benoit Steiner, Paul Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. Tensorflow: A system for large-scale machine learning. In *USENIX Conference on Operating Systems Design and Implementation*, pages 265–283, 2016.

[2] Humam Alwassel, Dhruv Mahajan, Lorenzo Torresani, Bernard Ghanem, and Du Tran. Self-supervised learning by cross-modal audio-video clustering. *arXiv preprint arXiv:1911.12667*, 2019.

[3] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. In *Advances in Neural Information Processing Systems*, pages 15509–15519, 2019.

[4] Sagie Benaim, Ariel Ephrat, Oran Lang, Inbar Mosseri, William T Freeman, Michael Rubinstein, Michal Irani, and Tali Dekel. Speednet: Learning the speediness in videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9922–9931, 2020.

[5] Uta Buchler, Biagio Brattoli, and Bjorn Ommer. Improving spatiotemporal self-supervision by deep reinforcement learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 770–786, 2018.

[6] 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.

[7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.

[8] Hyeon Cho, Taehoon Kim, Hyung Jin Chang, and Wonjun Hwang. Self-supervised spatio-temporal representation learning using variable playback speed prediction. *arXiv preprint arXiv:2003.02692*, 2020.

[9] François Chollet, JJ Allaire, et al. R interface to keras. `https://github.com/rstudio/keras`, 2017.

[10] Ali Diba, Vivek Sharma, et al. Dynamonet: Dynamic action and motion network. In *ICCV*.

[11] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1422–1430, 2015.

[12] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6202–6211, 2019.

[13] Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould. Self-supervised video representation learning with odd-one-out networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3636–3645, 2017.

[14] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE, 2006.

[15] Tengda Han, Weidi Xie, and Andrew Zisserman. Video representation learning by dense predictive coding. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[16] Tengda Han, Weidi Xie, and Andrew Zisserman. Memory-augmented dense predictive coding for video representation learning. In *European Conference on Computer Vision*, 2020.

[17] Tengda Han, Weidi Xie, and Andrew Zisserman. Self-supervised co-training for video representation learning. *arXiv preprint arXiv:2010.09709*, 2020.

[18] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6546–6555, 2018.

[19] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. *arXiv preprint arXiv:1911.05722*, 2019.

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[21] Olivier J Hénaff, Ali Razavi, Carl Doersch, SM Eslami, and Aaron van den Oord. Data-efficient image recognition with contrastive predictive coding. *arXiv preprint arXiv:1905.09272*, 2019.

[22] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[23] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.

[24] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.

[25] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*, 2018.

[26] Noureldien Hussein, Efstratios Gavves, and Arnold WM Smeulders. Timeception for complex action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 254–263, 2019.

[27] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning (ICML)*, 2015.

[28] Phillip Isola, Daniel Zoran, Dilip Krishnan, and Edward H Adelson. Learning visual groups from co-occurrences in space and time. *arXiv preprint arXiv:1511.06811*, 2015.

[29] Dinesh Jayaraman and Kristen Grauman. Slow and steady feature analysis: higher order temporal coherence in video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3852–3861, 2016.

[30] Longlong Jing and Yingli Tian. Self-supervised spatiotemporal feature learning by video geometric transformations. *CoRR*, abs/1811.11387, 2018.

[31] 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.

[32] Dahun Kim, Donghyeon Cho, and In So Kweon. Self-supervised video representation learning with space-time cubic puzzles. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8545–8552, 2019.

[33] Quan Kong, Wenpeng Wei, Ziwei Deng, Tomoaki Yoshinaga, and Tomokazu Murakami. Cycle-contrast for self-supervised video representation learning. In *Advances in Neural Information Processing Systems*, 2020.

[34] Bruno Korbar, Du Tran, and Lorenzo Torresani. Cooperative learning of audio and video models from self-supervised synchronization. In *Advances in Neural Information Processing Systems*, pages 7763–7774, 2018.

[35] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[36] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *2011 International Conference on Computer Vision*, pages 2556–2563. IEEE, 2011.

[37] Quoc V Le, Will Y Zou, Serena Y Yeung, and Andrew Y Ng. Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. In *CVPR 2011*, pages 3361–3368. IEEE, 2011.

[38] Honglak Lee, Alexis Battle, Rajat Raina, and Andrew Y Ng. Efficient sparse coding algorithms. In *Advances in neural information processing systems*, pages 801–808, 2007.

[39] Hsin-Ying Lee, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Unsupervised representation learning by sorting sequences. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 667–676, 2017.

[40] William Lotter, Gabriel Kreiman, and David Cox. Deep predictive coding networks for video prediction and unsupervised learning. *arXiv preprint arXiv:1605.08104*, 2016.

[41] Dezhao Luo, Chang Liu, Yu Zhou, Dongbao Yang, Can Ma, Qixiang Ye, and Weiping Wang. Video cloze procedure for self-supervised spatio-temporal learning. *arXiv preprint arXiv:2001.00294*, 2020.

[42] Michael Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. *arXiv preprint arXiv:1511.05440*, 2015.

[43] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9879–9889, 2020.

[44] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. *arXiv preprint arXiv:1912.01991*, 2019.

[45] Ishan Misra, C Lawrence Zitnick, and Martial Hebert. Shuffle and learn: unsupervised learning using temporal order verification. In *European Conference on Computer Vision*, pages 527–544. Springer, 2016.

[46] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[47] AJ Piergiovanni, Anelia Angelova, and Michael S Ryoo. Evolving losses for unsupervised video representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 133–142, 2020.

[48] Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, Huisheng Wang, Serge Belongie, and Yin Cui. Spatiotemporal contrastive video representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6964–6974, 2021.

[49] Rajat Raina, Alexis Battle, Honglak Lee, Benjamin Packer, and Andrew Y Ng. Self-taught learning: transfer learning from unlabeled data. In *Proceedings of the 24th international conference on Machine learning*, pages 759–766, 2007.

[50] Marc'Aurelio Ranzato, Fu Jie Huang, Y-Lan Boureau, and Yann LeCun. Unsupervised learning of invariant feature hierarchies with applications to object recognition. In *2007 IEEE conference on computer vision and pattern recognition*, pages 1–8. IEEE, 2007.

[51] Adrià Recasens, Pauline Luc, Jean-Baptiste Alayrac, Luyu Wang, Florian Strub, Corentin Tallec, Mateusz Malinowski, Viorica Patraucean, Florent Altché, Michal Valko, et al. Broaden your views for self-supervised video learning. *arXiv preprint arXiv:2103.16559*, 2021.

[52] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

[53] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[54] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhudinov. Unsupervised learning of video representations using lstms. In *International conference on machine learning*, pages 843–852, 2015.

[55] Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. Learning video representations using contrastive bidirectional transformer. *arXiv preprint arXiv:1906.05743*, 2019.

[56] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[57] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *arXiv preprint arXiv:1906.05849*, 2019.

[58] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015.

[59] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6450–6459, 2018.

[60] Michael Tschannen, Josip Djolonga, Marvin Ritter, Aravindh Mahendran, Neil Houlsby, Sylvain Gelly, and Mario Lucic. Self-supervised learning of video-induced visual invariances. *arXiv preprint arXiv:1912.02783*, 2019.

[61] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.

[62] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103, 2008.

[63] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Anticipating visual representations from unlabeled video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 98–106, 2016.

[64] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Generating videos with scene dynamics. In *Advances in neural information processing systems*, pages 613–621, 2016.

[65] Carl Vondrick, Abhinav Shrivastava, Alireza Fathi, Sergio Guadarrama, and Kevin Murphy. Tracking emerges by colorizing videos. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 391–408, 2018.

[66] Jue Wang, Gedas Bertasius, Du Tran, and Lorenzo Torresani. Long-short temporal contrastive learning of video transformers. *arXiv preprint arXiv:2106.09212*, 2021.

[67] Jiangliu Wang, Jianbo Jiao, Linchao Bao, Shengfeng He, Yunhui Liu, and Wei Liu. Self-supervised spatio-temporal representation learning for videos by predicting motion and appearance statistics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4006–4015, 2019.

[68] Jiangliu Wang, Jianbo Jiao, and Yun-Hui Liu. Self-supervised video representation learning by pace prediction. In *European Conference on Computer Vision*, pages 504–521. Springer, 2020.

[69] Limin Wang, Yuanjun Xiong, Zhe Wang, and Yu Qiao. Towards good practices for very deep two-stream convnets. *arXiv preprint arXiv:1507.02159*, 2015.

[70] Xiaolong Wang and Abhinav Gupta. Unsupervised learning of visual representations using videos. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2794–2802, 2015.

[71] Xiaolong Wang, Allan Jabri, and Alexei A Efros. Learning correspondence from the cycle-consistency of time. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2566–2576, 2019.

[72] Donglai Wei, Joseph J Lim, Andrew Zisserman, and William T Freeman. Learning and using the arrow of time. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8052–8060, 2018.

[73] Zhirong Wu, Yuanjun Xiong, Stella Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance-level discrimination. *arXiv preprint arXiv:1805.01978*, 2018.

[74] Dejing Xu, Jun Xiao, Zhou Zhao, Jian Shao, Di Xie, and Yueting Zhuang. Self-supervised spatiotemporal learning via video clip order prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10334–10343, 2019.

[75] Ceyuan Yang, Yinghao Xu, Bo Dai, and Bolei Zhou. Video representation learning with visual tempo consistency. *arXiv preprint arXiv:2006.15489*, 2020.

[76] Xitong Yang, Xiaodong Yang, Sifei Liu, Deqing Sun, Larry Davis, and Jan Kautz. Hierarchical contrastive motion learning for video action recognition. *arXiv preprint arXiv:2007.10321*, 2020.

[77] Ting Yao, Yiheng Zhang, et al. Seco: Exploring sequence supervision for unsupervised representation learning. *arXiv*, 2020.

[78] Mang Ye, Xu Zhang, Pong C Yuen, and Shih-Fu Chang. Unsupervised embedding learning via invariant and spreading instance feature. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6210–6219, 2019.

[79] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[80] Xu Zhang, Felix X Yu, Sanjiv Kumar, and Shih-Fu Chang. Learning spread-out local feature descriptors. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4595–4603, 2017.

[81] Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba. Temporal relational reasoning in videos. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 803–818, 2018.

[82] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017.