Self-Supervised Video Representation Learning by Video Incoherence Detection

Haozhi Cao, Yuecong Xu, *Member, IEEE*, Kezhi Mao, *Member, IEEE*, Lihua Xie, *Fellow, IEEE*, Jianxiong Yin, Simon See, Qianwen Xu, *Member, IEEE*, and Jianfei Yang, *Member, IEEE*

***Abstract*—This article introduces a novel self-supervised method that leverages incoherence detection for video representation learning. It stems from the observation that the visual system of human beings can easily identify video incoherence based on their comprehensive understanding of videos. Specifically, we construct the incoherent clip by multiple subclips hierarchically sampled from the same raw video with various lengths of incoherence. The network is trained to learn the high-level representation by predicting the location and length of incoherence given the incoherent clip as input. Additionally, we introduce intravideo contrastive learning to maximize the mutual information between incoherent clips from the same raw video. We evaluate our proposed method through extensive experiments on action recognition and video retrieval using various backbone networks. Experiments show that our proposed method achieves remarkable performance across different backbone networks and different datasets compared to previous coherence-based methods.**

***Index Terms*—Action recognition, neural networks, self-supervised learning, video representation learning.**

I. Introduction

FULLY supervised learning has achieved great success in video representation learning during the past decade. However, its remarkable performance heavily relies on a large amount of labeled data, which requires considerable resources and time to annotate. Moreover, fully supervised methods are designed to extract task-specific representations, which limits their transferability and generalization capability. To address the aforementioned challenges, recent works have paid more attention to self-supervised learning, which aims to extract generalized representations from more accessible unlabeled data on the Internet.

The core of self-supervised methods is to design a pretext task that prompts the network to learn effective representations from unlabeled data. Existing self-supervised methods can be broadly categorized into two types: 1) dense prediction and 2) spatiotemporal reasoning. Dense prediction-based methods require the network to predict parts of low-level representations, such as future frames [1], [2] and optical flows [3]. Although these methods can achieve outstanding performance, they usually involve laborious hand-crafted features (e.g., optical flow [3]) or complicated computation process [4], [5], which are both time consuming and resource expensive. To improve the efficiency, recent spatiotemporal reasoning methods, such as clip order prediction [6], [7], [8], [9], video speed prediction [10], [11], [12], and spatiotemporal statistic prediction [13] tend to learn high-level spatiotemporal correlations in raw videos. Specifically, methods based on clip order prediction attempt to leverage video coherence for representation learning, where the supervision signal is generated from frame order disruption. In this article, we propose a novel task, named incoherence detection, to leverage video coherence for video representation learning from a new perspective.

Intuitively, our visual systems can effortlessly identify the incoherence of videos (e.g., loss of frames caused by connection latencies) by detecting abnormal motion based on our understanding of videos. In this case, the incoherence can be viewed as noise to motion information. Detecting incoherence requires the network to possess a comprehensive understanding of videos which motivates this article. Take Fig. 1 as an example: one can easily tell the difference between coherent and incoherent videos based on their understanding of the action "High Jump." Specifically, we can deduce that the athlete should be leaping over the bar in the next frame given previous frames (1 and 2). However, in frame (4), the athlete suddenly appears on the right side of the bar without the process of leaping, which is inconsistent with our deduction. This bi-directional reasoning of video contents could be an effective supervision signal for the network to learn the high-level representations of videos.

Inspired by the aforementioned observation, we propose a simple-yet-effective method named video incoherence detection (VID) for self-supervised video representation learning. During the self-supervised training process of VID, each training sample is generated as an incoherent clip, constructed

Manuscript received 3 August 2022; revised 18 December 2022 and 15 March 2023; accepted 2 April 2023. This work was supported in part by the National Research Foundation, Singapore under its Medium Sized Center for Advanced Robotics Technology Innovation, and in part by NTU Presidential Postdoctoral Fellowship, "Adaptive Multimodal Learning for Robust Sensing and Recognition in Smart Cities" Project Fund, Nanyang Technological University, Singapore. This article was recommended by Associate Editor G. C. Anagnostopoulos. *(Corresponding author: Jianfei Yang.)*

Haozhi Cao, Kezhi Mao, Lihua Xie, and Jianfei Yang are with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore (e-mail: haozhi001@e.ntu.edu.sg; ekzmao@ntu.edu.sg; elhxie@ntu.edu.sg; yang0478@e.ntu.edu.sg).

Yuecong Xu is with the Institute for Infocomm Research, A*STAR, Singapore (e-mail: xuyu0014@e.ntu.edu.sg).

Jianxiong Yin and Simon See are with NVIDIA AI Tech Center, Singapore (e-mail: jianxiongy@nvidia.com; ssee@nvidia.com).

Qianwen Xu is with the Department of Electric Power and Energy Systems, KTH Royal Institute of Technology, 10044 Stockholm, Sweden (e-mail: qianwenx@kth.se).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCYB.2023.3265393.

Digital Object Identifier 10.1109/TCYB.2023.3265393

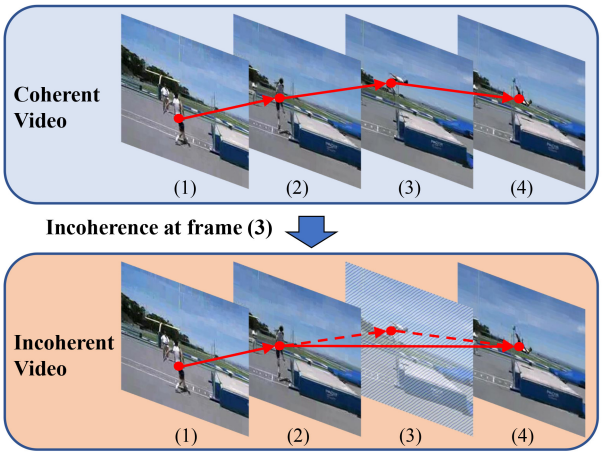


Fig. 1. Illustration about how video incoherence affects motion information. Both videos demonstrate the action "High Jump". The incoherence caused by losing frame (3) leads to a distortion of athlete motion between frame (2) and frame (4) in the incoherent video, which is incompatible with our understanding of the action "High Jump". This observation suggests that incoherence detection requires a comprehensive understanding of videos.

by multiple subclips from the same raw video. Specifically, subclips are hierarchically sampled from the raw video given the random incoherence location and length. The incoherent clip is then constructed by concatenating the subclips along the temporal dimension. Different from previous coherence-based methods [6], [7], [8], [9] which undermine temporal orders, VID preserves the sequential relationship of the raw video during the generation process. The network can therefore learn temporal representations for incoherence detection.

Given the incoherent clips as input, the network is trained to detect the incoherence by two novel pretext tasks that predict the location and length of incoherence, denoted as Incoherence location detection (LoD) and Incoherence length detection (LeD), respectively. In addition, intravideo contrastive learning (ICL) is adopted as an optimization objective to maximize the mutual information between different incoherent clips from the same raw video.

The main contributions of our work include

1) Motivated by the fact that detecting incoherence requires semantic understanding, we propose a simple-yet-effective self-supervised method for video representation learning, called VID, utilizing a single temporal transformation for video representation learning.
2) By detecting incoherence within videos, our proposed VID leverages spatiotemporal relationships for self-supervised learning from a different perspective while avoiding trivial solutions.
3) We evaluate our VID using three different backbones in two downstream tasks across three datasets. Extensive experiments show that our VID achieves the state-of-the-art (SOTA) performance on action recognition and video retrieval compared to previous coherence-based methods.

II. Literature Review

A. Self-Supervised Learning

Despite the success of fully supervised learning in the video domain [14], [15], [16], [17], [18], these methods require expensive human annotation to generate ground-truth labels, limiting their practicality. To alleviate the dependency on human annotation, previous works propose different learning strategies to extract video representations without labeling all training samples, including unsupervised learning [19], [20], semisupervised learning [21], [22], [23], domain adaptation [24], [25], [26], and self-supervised learning.

Specifically, self-supervised learning has attracted more and more attention in video representation learning, since it is tailored to leverage a large amount of unlabeled data from the Internet without requiring laborious human annotation. Different from semisupervised or domain adaptation methods which attempt to alleviate the labeling effort in the downstream tasks (e.g., [21] reduces the labeling effort in the video domains by transferring knowledge from the image domain), self-supervised methods aim to provide effective pretrained models without any human annotations during the pretraining stage. It builds upon some prior works, such as [27] which reveals the benefit of additional auxiliary during traditional fully supervised training, and [28] which explicitly introduces the dual-stage learning process: the network is purely trained on the pretext task and then transferred to the target problem. This dual-stage training procedure is now termed self-supervised learning and has been widely explored in images [29], [30], [31] or natural language [32], [33]. Early works have expanded self-supervised methods from other domains to videos, such as DPC [34] inspired by CPC [35] in the image domain and transformer-based methods [36], [37] inspired by BERT [32].

Recent self-supervised methods for video representation learning can be categorized into two types: 1) dense prediction and 2) spatiotemporal reasoning. Dense prediction methods [1], [2], [3], [4], [5], [34] require the network to predict the low-level information of videos. For instance, Vondrick et al. [1] and Srivastava et al. [2] proposed to learn video representations by predicting future frames whose foreground and background are generated from independent streams. To leverage multimodality video information, some previous works propose to generate supervision signals through the input of 3-D videos [3] or RGB-D data [38]. Another trend is to utilize estimated modalities (e.g., tracking trajectories [39] or optical flow [40]) that embed rich temporal relationships to obtain supervision signals so that the network can capture more accurate motion information during pretraining. Compared to spatiotemporal reasoning methods, dense prediction methods can usually achieve better performance due to their sophisticated training process [1], [2] and the utilization of multimodality [3], [38], [39], [40]. However, dense prediction methods inevitably utilize additional decoders to output low-level information in the pretraining stage, introducing extra computational complexity and inefficiency compared to spatiotemporal methods.

Instead of directly predicting low-level information, recent methods have proposed utilizing spatiotemporal reasoning to generate supervision signals based on correlations or characteristics of videos. In contrast to dense prediction methods, spatiotemporal reasoning methods explore various pretext tasks to generate effective supervision signals, such as temporal order prediction [6], [7], [8], [9], [41] and video speed

prediction [10], [11], [12]. Inspired by the sequential relationships of videos, previous works [7], [8], [9], [42] propose to predict or identify the correct frame order given clips shuffled along the temporal dimension. Further advance in this direction includes Xu et al. [6] who applied the order prediction method with 3-D-CNN and Kim et al. [41] who expanded the order prediction to the spatial dimension. On the other hand, recent methods [10], [11], [12] propose to extract effective representations by predicting the speed of videos. Specifically, Yao et al. [12] and Wang et al. [10] combined the speed prediction task with regeneration and contrastive learning, respectively. Jenni et al. [11] proposed to recognize various temporal transformations under different speeds and Chen et al. [43] achieved SOTA performance by reformulating the speed prediction task into a relative one. Inspired by humans' sensitivity toward incoherence in videos, we argue that VID requires a semantic understanding of video contents, which can be explored to learn effective video representations.

B. Contrastive Learning

Contrastive learning has demonstrated its effectiveness in self-supervised learning. In the image domain, multiple methods [29], [30], [31], [44] have been proposed to extract effective image representations by contrastive learning. Building upon this success, recent methods [10], [45], [46], [47], [48] explore different manners to adopt contrastive learning in the video domain. The core idea is to maximize the mutual information by contrasting positive pairs and negative samples. For instance, Wang et al. [10] and Dwibedi et al. [45] proposed to align spatiotemporal representations of the same action or same context. Yao et al. [48] conducted contrastive learning from spatial, spatiotemporal, and sequential perspectives, while Qian et al. [49] further simplified the spatiotemporal contrastive reasoning and achieved SOTA performance by utilizing carefully designed data augmentations and deeper networks. Building upon the success of MoCo [46], Pan et al. [50] integrated the momentum queue in [46] with temporal adversarial learning between the input and its augmented variant to extract temporal robust representations of input samples. Additionally, considering the degradation effect of the momentum queue, a temporal decay mechanism is designed to attend to more recent keys in the queue. Han et al. [51] further expanded the contrastive learning to a novel co-training scheme by co-training the network for each modality through contrastive loss, which leverages the complementary information from multiple modalities. Inspired by the success of BERT [32], VATT [52] is proposed to capture cross-modal video representations from four different modalities (raw videos, audios, and text) by utilizing multimodal contrastive loss with transformer-based networks. By combining temporal consistency with existing imaged-based contrastive learning methods, Feichtenhofer et al. [53] revealed the potentials of contrastive learning, achieving competitive or even superior performance compared to fully supervised learning.

Despite the outstanding performance methods based on pure contrastive learning, they usually demand a relatively large batch size [48], [49], [52] or specifically designed mechanisms (e.g., the memory bank [29], [54] or momentum queue [46]) to ensure a sufficient number of negative samples for each update. Therefore, the performance of contrastive loss might be limited when encountering situations with limited computational resources. Instead of using pure contrastive learning, we utilize ICL as an additional objective to maximize the mutual information between different incoherent clips from the same video. With the guidance of other training objectives, the improvement brought by contrastive loss is noticeable even without adopting a large batch size during the pretrained stage.

III. Proposed Methods

Coherence is one of the crucial properties of videos as natural videos are formed by sets of frames coherently observed. Our visual systems can easily identify incoherence caused by loss of frames within a video clip, which demonstrates that the detection of incoherence requires a semantic understanding of videos. This observation motivates us to develop a self-supervised method by leveraging incoherence detection for video representation learning.

In this work, we propose to extract effective spatiotemporal representations via VID based on a simple temporal transformation in a self-supervised manner. We first illustrate how to generate incoherent clips from raw videos. Based on these generated clips, LoD, LeD, and ICL are proposed for self-supervised learning. To clarify the whole learning procedure, we summarize the overall learning objective and framework of VID in Section III-C.

A. Generation of Incoherent Video Clips

To utilize VID, we first generate incoherent clips from raw videos. Given a raw video V, the incoherent clip $\mathcal{V}_{\text{inc}}$ is constructed by k subclips $\mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_k$ sampled from V, with a certain length of incoherence between each subclip. The location L_{loc} and length l_{inc} of incoherence are both randomly generated. The length of incoherence l_{inc} between subclips is limited within the range

$$l_{\text{inc}} \in \left[l_{\text{inc}}^{\min}, l_{\text{inc}}^{\min} + 1, \ldots, l_{\text{inc}}^{\max} \right] \tag{1}$$

where $l_{\text{inc}}^{\min}$ and $l_{\text{inc}}^{\max}$ are both hyperparameters indicating the upper and lower bounds of the incoherence length, respectively. The purposes of this constraint are twofold. First, this constraint determines the number of classes for our following self-supervised task LeD. Second, this constraint prevents the length of incoherence between subclips from being either too vague or too obvious, thereby avoiding learning trivial solutions. For simplicity, we illustrate how to generate $\mathcal{V}_{\text{inc}}$ by two subclips as an example in Fig. 2.

1) Selection of Incoherence Location: The incoherence location L_{loc} indicates the relative concatenation location between two subclips. Formally, given the desired length of the incoherent clip l_0, the location of incoherence L_{loc} is sampled as follows:

$$l_1 \in \{1, 2, \ldots, l_0 - 1\}, \quad l_2 = l_0 - l_1 \tag{2}$$

$$L_{\text{loc}} = l_1 - 1 \tag{3}$$

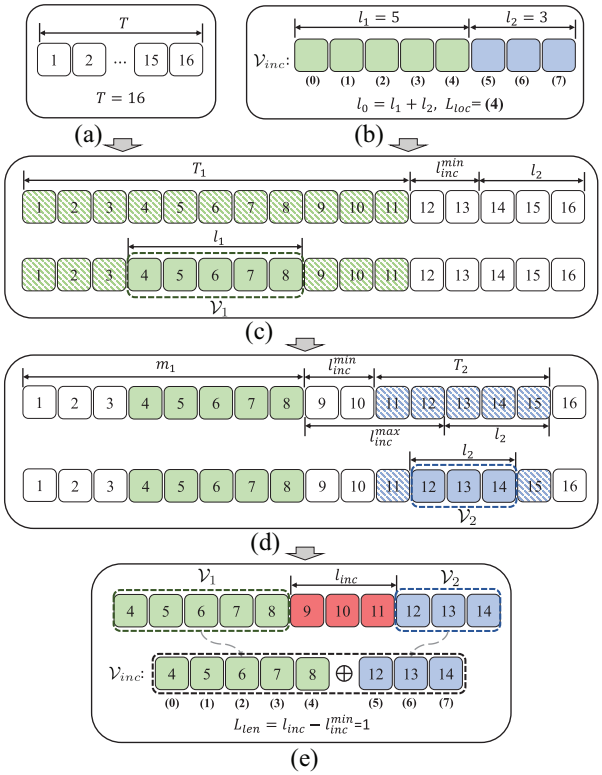


Fig. 2. Generation process of the incoherent clip $\mathcal{V}_{\text{inc}}$. Indices 1–16 in square boxes denote the frame indices in the raw video V while indices (0–7) denote the relative frame indices in $\mathcal{V}_{\text{inc}}$. Squares in shadow and color refer to the sample range and sampled frames for the corresponding subclip, respectively. $\mathcal{V}_{\text{inc}}$ is generated as the concatenation of $\mathcal{V}_1$ and $\mathcal{V}_2$ along the temporal dimension as shown in (e). (a) Raw video V. (b) Selected incoherence location L_{loc}. (c) Sampling of the first subclip $\mathcal{V}_1$. (d) Sampling of the second subclip $\mathcal{V}_2$. (e) Temporal concatenation.

where l_1 and l_2 are the length of $\mathcal{V}_1$ and $\mathcal{V}_2$ as illustrated in Fig. 2(b), where squares in different colors refer to allocated frame positions for different subclips in $\mathcal{V}_{\text{inc}}$. L_{loc} indicates the relative location of incoherence, where subclips concatenate and also the label for the following LoD task.

2) Hierarchical Selection of Subclips: Given the subclip lengths l_1 and l_2, subclips $\mathcal{V}_1$ and $\mathcal{V}_2$ are hierarchically sampled from the raw video V with incoherence between each other. The incoherent clip $\mathcal{V}_{\text{inc}}$ is generated as the temporal concatenation of $\mathcal{V}_1$, and $\mathcal{V}_2$. While previous works [10] propose to sample frames by looping over the raw video, this strategy is not compatible with our proposed VID since it could introduce unexpected incoherence when looping from the end to the start of the video. Instead, to preserve the sequential relationship of the raw video, we propose a hierarchical sampling strategy that maximizes the sample range of each subclip while satisfying the constraint in (1).

Illustrated as the upper row in Fig. 2(c), given the raw video V of the length T, the sample range T_1 of the first subclip $\mathcal{V}_1$ is determined by reserving sufficient frames for subsequent subclips. This allows the following subclip $\mathcal{V}_2$ to be sampled from the rest of the raw frames wherever the $\mathcal{V}_1$ locates in T_1, which preserves the sequential relationship and satisfies the constraint in (1). Formally, given l_2 and $l_{\text{inc}}^{\min}$, the sample range T_1 is computed as

$$t_1^{\min} = 1 \tag{4}$$

$$t_1^{\max} = T - l_{\text{inc}}^{\min} - l_2 \tag{5}$$

$$T_1 = \left\{ t_1^{\min}, t_1^{\min} + 1, \ldots, t_1^{\max} \right\} \tag{6}$$

where $t_1^{\min}$ and $t_1^{\max}$ are the lower and upper bound of the range T_1. Given the range T_1, $\mathcal{V}_1$ is uniformly sampled as $\mathcal{V}_1 \in T_1$ illustrated as the lower row in Fig. 2(c).

Subsequently, the range of the second subclip T_2 is determined by the sampled subclip $\mathcal{V}_1$ and the range of l_{inc} in (1). As shown in the upper row of Fig. 2(d), given the raw frame index of the last frame in $\mathcal{V}_1$ denoted as $m_1 = \max(\mathcal{V}_1)$, the sample range T_2 is computed as

$$t_2^{\min} = m_1 + l_{\text{inc}}^{\min} + 1 \tag{7}$$

$$t_2^{\max} = \min\left(m_1 + l_{\text{inc}}^{\max} + l_2, T\right) \tag{8}$$

$$T_2 = \left\{ t_2^{\min}, t_2^{\min} + 1, \ldots, t_2^{\max} \right\} \tag{9}$$

where $t_2^{\min}$ and $t_2^{\max}$ are the upper and lower bounds which ensure that l_{inc}, the length of incoherence between $\mathcal{V}_2$ and $\mathcal{V}_1$, always satisfies the constraint in (1). Similar to $\mathcal{V}_1$, the second clip $\mathcal{V}_2$ is uniformly sampled as $\mathcal{V}_2 \in T_2$, illustrated as the lower row of Fig. 2(d).

Given the subclips $\mathcal{V}_1$ and $\mathcal{V}_2$, the incoherent clip $\mathcal{V}_{\text{inc}}$ and its label L_{len} for Incoherence LeD task are generated as

$$\mathcal{V}_{\text{inc}} = \mathcal{V}_1 \oplus \mathcal{V}_2 \tag{10}$$

$$l_{\text{inc}} = \min(\mathcal{V}_2) - \max(\mathcal{V}_1) \tag{11}$$

$$L_{\text{len}} = l_{\text{inc}} - l_{\text{inc}}^{\min} \tag{12}$$

where $\oplus$ indicates the concatenation of two subclips $\mathcal{V}_1$ and $\mathcal{V}_2$ along the temporal dimension.

B. Optimization Objectives

We propose two novel self-supervised tasks, including Incoherence LoD and Incoherence LeD, to detect the incoherence in incoherent clips while maximizing the mutual information between different incoherent clips from the same raw video by ICL. Specifically, given an incoherent clip $\mathcal{V}_{\text{inc}}$, the high-level representation is first extracted as $h = f(\mathcal{V}_{\text{inc}})$, where $f(\cdot)$ denotes the encoder. Given the representation h, the optimization objectives of VID include LoD, LeD, and ICL.

1) Incoherence Location Detection : Given the high-level representation h and its incoherence location label L_{loc}, the network is required to predict the location of incoherence in $\mathcal{V}_{\text{inc}}$. This is mainly inspired by the sensitivity of human perception toward the loss of frames within video clips. By identifying the abnormal motion caused by incoherence, the network is driven to learn semantic representations of videos. The LoD task is formulated as a single-label classification problem. Given the representation h and label L_{loc}, the network is optimized by the cross-entropy loss formulated as

$$l_{\text{LoD}} = -\sum_{i=0}^{l_0-1} y_i^{\text{loc}} \log\left(\frac{\exp\left(z_i^{\text{loc}}\right)}{\sum_{j=0}^{l_0-1} \exp\left(z_j^{\text{loc}}\right)} \right) \tag{13}$$

where $z^{\text{loc}} \in \mathbb{R}^{l_0-1}$ is the output of fully connected layers $\phi^{\text{loc}}(\cdot)$ given the representation h as input. $y^{\text{loc}} \in \mathbb{R}^{l_0-1}$ is the one-hot label vector whose element at L_{loc} equals 1. In practice, given a mini-batch of representations Z^{loc}, (13) is applied to each representation in $Z^{\text{loc}} \in \mathbb{R}^{N\times(l_0-1)}$, where N denotes the batch size. The $\mathcal{L}_{\text{loc}}$ loss is then calculated as the average loss of representations in Z^{loc}.

2) Incoherence Length Detection: In addition to LoD, the network is required to predict the length of incoherence given the high-level representation h and its corresponding label L_{len} of (12). The proposed LeD task is designed as a regularization measure to avoid trivial learning. In some cases, incoherence may occur at the period when the distribution of low-level representation intensively changes (e.g., intensive movement of the camera or sudden changes in light conditions). This could cause a distinct difference in low-level representation between subclips of the incoherent clip $\mathcal{V}_{\text{inc}}$, leading to trivial learning when adopting LoD as the only optimization objective. Compared with LoD which extracts semantic representation, our proposed LeD can be regarded as a simple yet challenging task that requires the network to deduce the length of incoherence with respect to the raw video. In practice, although the accuracy of LeD is relatively low (with Top1 at about 25%), our ablation study shows that it can bring noticeable improvement as it can effectively prevent VID from learning trivial solutions based on low-level information.

Similar to LoD, the LeD task can also be formulated as a classification problem, where cross-entropy loss is utilized for optimization as

$$l_{\text{LeD}} = -\sum_{i=0}^{\Delta l_{\text{inc}}} y_i^{\text{len}} \log\left(\frac{\exp\left(z_i^{\text{len}}\right)}{\sum_{j=0}^{\Delta l_{\text{inc}}} \exp\left(z_j^{\text{len}}\right)}\right) \tag{14}$$

where $z^{\text{len}} \in \mathbb{R}^{l_0-1}$ is the output of fully connected layers $\phi^{\text{len}}(\cdot)$ given h as input. $y^{\text{len}} \in \mathbb{R}^{\Delta l_{\text{inc}}}$ is the one-hot label vector of incoherence length and Δl_{inc} is the difference between the upper bound and lower bound of incoherence length. Similar to LoD, (14) is also applied to all representations of the mini-batch $Z^{\text{len}} \in \mathbb{R}^{N\times\Delta l_{\text{inc}}}$, whose loss $\mathcal{L}_{\text{LeD}}$ is calculated as the average loss of Z^{len}.

3) Intravideo Contrastive Learning: Contrastive learning can effectively extract the mutual information between variously augmented samples from the same source. Recent works [10], [48] demonstrate its great potential, exceeding other self-supervised or even supervised methods when adopting a large batch size. In this work, we include ICL as an extra optimization objective to maximize the mutual information between different incoherent clips from the same video. This is inspired by the fact that human beings can correctly recognize video actions based on the mutual information shared across the video regardless of the location and length of incoherence. Although the motion of different incoherent clips is distorted in different manners, their representation should be homogeneous since they indicate the same video contents, which can therefore be leveraged as the supervision signal for video representation learning.

Formally, given a mini-batch of N raw videos $\mathbf{V} = \{V_1, V_2, \ldots, V_N\}$, two incoherent clips are randomly generated for each raw video $V_i \in \mathbf{V}$ as in Section III-A. The incoherent clips from the same raw video V_i are considered as the positive pair denoted as $\{\mathcal{V}_{\text{inc}}^i, \widetilde{\mathcal{V}}_{\text{inc}}^i\}$, while those from different raw videos are regarded as negative pairs denoted as $\{\mathcal{V}_{\text{inc}}^i, \mathcal{V}_{\text{inc}}^k \mid k \neq i\}$. Each incoherent clip $\mathcal{V}_{\text{inc}}^i$ is then fed to the network $f(\cdot)$, forming the high-level representation h_i. The representation h_i is subsequently passed to a fully connected layer $\phi^{cl}(\cdot)$ followed by a nonlinear ReLU activation, resulting in features z_i^{cl} as the input of our proposed ICL. Provided with features of the positive pair $\{z_i^{cl}, \widetilde{z}_i^{cl}\}$ and features of negative pairs $\{z_i^{cl}, z_k^{cl}\}, k \neq i$, the contrastive loss is computed as

$$\mathcal{L}_{\text{ICL}} = -\frac{1}{2N}\sum_{i=1}^{2N} \log\left(\frac{\exp\left(s\left(z_i^{cl}, \widetilde{z}_i^{cl}\right)\right)}{\exp\left(s\left(z_i^{cl}, \widetilde{z}_i^{cl}\right)\right) + \mathcal{D}(i)}\right) \tag{15}$$

$$\mathcal{D}(i) = \sum_{k\neq i} \exp\left(s\left(z_i^{cl}, z_k^{cl}\right)\right) \tag{16}$$

where $s(u, v) = u^\top v/\|u\|\|v\|$ indicates the similarity between feature u and v. $\mathcal{D}(i)$ is the summation of exponential similarity between features of negative pairs.

C. Network Structure and Training

The overall network structure is illustrated in Fig. 3. Given the unlabeled raw video, the incoherent clips are first generated as described in Section III-A, where each raw video randomly generates two different incoherent clips as shown in Fig. 3(a). The batch of incoherent clips is then fed to the encoder $f(\cdot)$ implemented as the 3-D CNN backbone. The ultimate optimization objective is formulated as

$$\mathcal{L} = \alpha\mathcal{L}_{\text{LoD}} + \beta\mathcal{L}_{\text{LeD}} + \lambda\mathcal{L}_{\text{ICL}} \tag{17}$$

where α, β, and λ are the coefficients of three loss terms from the subtasks, respectively.

IV. Experiments

In this section, we present thorough experiments to justify the effectiveness of our proposed VID. We first illustrate our experimental settings and subsequently justify our VID design through detailed ablation studies. Finally, VID is evaluated on two downstream tasks, including action recognition and video retrieval in comparison with SOTA methods.

A. Experimental Settings

1) Datasets: We evaluate our VID across three action-recognition datasets, including UCF101 [54], HMDB51 [55], and Kinetics-400 [56]. UCF101 is a widely used video dataset for action recognition, which contains 13 320 videos with 101 action categories. HMDB51 is a relatively smaller yet challenging dataset for action recognition, including about 7 000 videos with 51 action classes. Both UCF101 and HMDB51 are divided into three training and testing splits. Kinetics-400, denoted as K-400, is a large dataset for action recognition, containing about 304 000 videos with 400 action classes collected from the online video platform YouTube. Same as

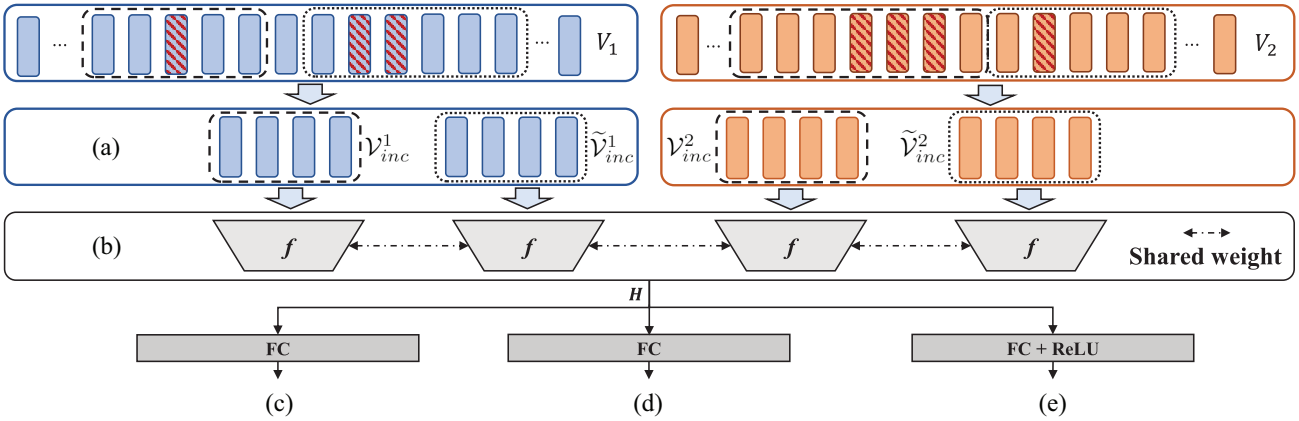


Fig. 3. Structure of our proposed VID method. The first row indicates two raw videos V_1 and V_2, each of which generates two incoherent clips, respectively. The generated incoherent clips are then fed into a single 3-D CNN backbone. The extracted high-level representation H is subsequently passed to three different linear or nonlinear layers to perform three different subtasks, including Incoherence LoD, Incoherence LeD, and ICL. (a) Generation of incoherent clips. (b) 3-D CNN backbone. (c) Incoherence location: Z^{loc}. (d) Incoherence length: Z^{len}. (e) ICL: Z^{cl}.

the setting of prior work [10], [11], we utilize the training split of Kinetics-400 and the training split 1 of UCF101 for self-supervised pretraining.

2) Backbone Networks: To fairly compare our proposed method with others [6], [10], we evaluate our proposed VID with three different 3-D CNN networks in our experiments, including C3D [57], R3D [58], and R(2+1)D [59]. The aforementioned backbones have been widely used to evaluate self-supervised methods in previous research [6], [10], [12], [60]. Specifically, C3D [57] is constructed by direct extending 2-D CNN kernels to 3-D, while R3D [58] introduces the residual connections from 2-D CNNs to 3-D CNNs. Following previous works [11], [12], [41], we utilize R3D-18 which is the 18-layer variant of R3D. R(2+1)D [59] proposes to replace the traditional 3-D kernel with the combination of a 2-D kernel and a 1-D kernel for spatial and temporal feature extraction, respectively. In this work, we mainly conduct our experiments with the 18-layer variant of R(2+1)D thanks to its superior performance compared to others.

3) Augmentation and Other Details: Following the setting of prior work [10], [11], each incoherent clip includes 16 frames. The frame interval for subclip sampling is 1 (i.e., the same raw frame interval we used to sample subclips $\mathcal{V}_1$ and $\mathcal{V}_2$ in Fig. 2) and the range of incoherence length is set as $l_{\text{inc}} \in \{3, 4, \ldots, 10\}$. When pretraining on UCF101, we follow the epoch setting in [10], [61] which increases the epoch size from 9k to 90k (i.e., equivalent to 180 raw epochs) and include color jittering along the temporal dimension. Such epoch setting is sufficient to reach convergence on both UCF101 and K400 when training with VID (e.g., LoD can achieve Top1 of more than 90%). Frames are resized to 128×171 and then randomly cropped to 112×112. The whole input clip is subsequently flipped horizontally with a probability of 50%. The network is trained with a batch size of 30. The stochastic gradient descent [62] is utilized for optimization with the weight decay set to 0.005 and the momentum set to 0.9. The learning rate is initialized as 0.001 and divided by 10 every six epochs with a total training epoch of 18. For the coefficient settings, here we empirically set coefficients of subtasks α, β, and λ to 1, 0.1, and 0.1 following the same settings in [10]. We conduct our experiments utilizing PyTorch [63] with two NVIDIA Tesla P100.

TABLE I
ABLATION STUDY OF RANGE OF INCOHERENCE LENGTHS, JITTERING, AND THE HIERARCHICAL SAMPLING METHOD

Method	Jittering	Range of l_{inc}	UCF101(%)
Random	✓	-	56.7
VID	✓	[2, 10]	76.7
	✓	[4, 10]	77.3
	✓	[5, 10]	76.9
	✓	[3, 6]	76.8
	✓	[3, 8]	77.5
	✓	[3, 12]	77.1
	✓	[3, 14]	76.7
	✓	[3, 10]	**78.1**
	✗	[3, 10]	76.3

B. Ablation Studies

In this section, we justify the design of our proposed VID by ablation studies. We first illustrate the optimal range of the incoherence length and the necessity of the hierarchical sampling process. Subsequently, different combinations of our proposed pretext tasks are evaluated to justify our proposed methods. Our proposed incoherence detection is additionally evaluated with various backbones compared to previous coherence-based methods. All our ablation studies are conducted with R(2+1)D [59] pretrained on UCF101, unless otherwise specified.

1) Range of the Incoherence Length: We first explore the best range of incoherence length l_{inc}. Illustrated in Table I, the experiments are conducted by changing either the lower bound $l^{\text{min}}_{\text{inc}}$ or the upper bound $l^{\text{max}}_{\text{inc}}$. As $l^{\text{min}}_{\text{inc}}$ increases from 2, the performance of VID improves and peaks at 78.1% with $l^{\text{min}}_{\text{inc}} = 3$, while the performance begins to decline when $l^{\text{min}}_{\text{inc}}$ further expands. When $l^{\text{min}}_{\text{inc}}$ is smaller than 3, the incoherence between subclips is too difficult for the network to identify.

TABLE II
ABLATION STUDY OF HIERARCHICAL SAMPLING. HERE, DISABLING HIERARCHICAL OPTION REFERS TO LOOP-OVER SAMPLING

Backbone	Hierarchical	UCF101(%)	HMDB51(%)
C3D	✗	69.1	35.4
	✓	**70.2**	**37.7**
R3D	✗	70.6	34.7
	✓	**73.6**	**38.0**
R(2+1)D	✗	75.3	37.9
	✓	**78.5**	**41.5**

TABLE III
ABLATION STUDY OF DIFFERENT SUBTASKS

Sub-tasks	LoD / α	LeD / β	ICL / λ	UCF101(%)
Random Init	-	-	-	56.7
LoD	1	-	-	75.4
LeD	-	1	-	70.9
ICL	-	-	1	72.1
LoD+LeD	1	0.1	-	77.3
LoD+ICL	1	-	0.1	76.9
LeD+ICL	-	1	0.1	71.8
LoD+LeD+ICL	1	0.1	0.1	**78.1**

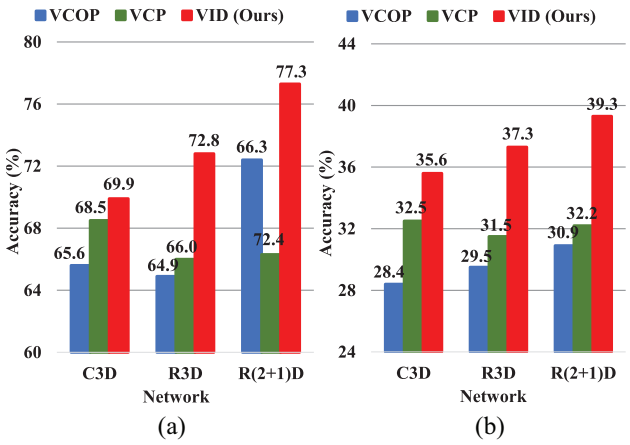


Fig. 4. Comparison with coherence-based methods. Even without ICL, our VID outperforms previous coherence-based methods by more than 1.4% on (a) UCF101, while this improvement further expands to more than 3.1% when testing on (b) HMDB51. This performance improvement is consistent across three different backbones.

The further increase of the lower bound degenerates the variety of incoherence length, leading to a drop in performance. Similar to the lower bound, when the upper bound of incoherence length grows from $l_{\text{inc}}^{\max} = 6$, the performance of VID rises consistently from 76.8% and then reaches a climax when $l_{\text{inc}}^{\max} = 10$, whereas a deteriorated performance can be observed as upper bound further expands, dropping from 78.1% with $l_{\text{inc}}^{\max} = 10$ to 76.7% with $l_{\text{inc}}^{\max} = 14$. As $l_{\text{inc}}^{\max}$ increases, the sample range of incoherence becomes more abundant, while the incoherence becomes too obvious when $l_{\text{inc}}^{\max} > 10$. This observation indicates that an inappropriate range of l_{inc} can result in too vague or too obvious incoherence, which leads to inferior performance. We thus set the range of l_{inc} as $[3, 10]$ in the following experiments.

2) Hierarchical Sampling Versus Loop-Over: As mentioned in Section III-A, the generation of incoherent video clips is specifically designed to avoid undesired incoherence caused by conventional loop-over sampling. Here, we justify the necessity of our proposed hierarchical sampling method in comparison with the loop-over sampling method. The loop-over sampling randomly selects the start frame of incoherent clips across the raw video without any constraint and will loop to the beginning of the raw video if the desired frame exceeds the length of the video. As shown in Table II, noticeable improvements can be observed across three different backbones on UCF101 and HMDB51 when adopting hierarchical sampling. Specifically, our hierarchical method achieves an improvement of more than 1.0% on all backbones when evaluated on UCF101 for action recognition. Such performance gap further expands to about 3.0% when adopting more competitive backbones, including R3D and R(2+1)D. When testing with HMDB51, the performance improvements brought by hierarchical sampling become more significant on all backbones, each of which exceeds 2.0% compared to the loop-over strategy. Such observation proves the effectiveness and necessity of our proposed hierarchical sampling in VID.

3) Different Subtasks: We further evaluate the performance of different subtasks. As shown in Table III, when utilizing a single subtask, networks pretrained with any subtask significantly exceed random initialization with a relative improvement of more than 25.0%. The network with LoD obtains the highest performance of 75.4% on UCF101, which justifies the dominant effect of LoD in VID. The network with ICL also achieves a competitive performance of 72.1%. However, when optimizing with only LeD, the network is required to directly predict the incoherence length without locating it. Therefore, the network cannot fully leverage incoherence detection in videos, leading to an inferior performance of 70.9%.

In terms of arbitrary pairs of subtasks, the LoD-based pairs (LoD+LeD and LoD+ICL) surpass the single LoD with noticeable margins of more than 2.0%. This justifies the effectiveness of LeD and ICL as additional objectives. On the other hand, an inferior performance of the LeD-based pair (LeD+ICL) can be observed, mainly due to the absence of LoD. Compared with LeD, the additional ICL of LeD+ICL slightly improves the performance by 0.9% since ICL relies less on the identification of incoherence locations. When compared with ICL, however, the performance of LeD+ICL marginally decreases by 0.3%, mainly because the network is not able to identify the incoherence without LoD, not to mention deducting the length of incoherence (i.e., LeD). This observation reveals the important role of LoD in VID, which also inspires us to maintain the dominant coefficient of LoD in the following experiments.

4) Comparison With Coherence-Based Methods: To justify the effectiveness of incoherence detection, we evaluate our VID without additional ICL subtask compared to previous coherence-based methods [6], [60] utilizing order prediction. Illustrated in Fig. 4, our VID outperforms previous coherence-based methods across various backbones and datasets. On UCF101, VID exceeds the previous VCOP [6] and VCP [60] by 4.3%–7.9% and 1.4%–11.0%, respectively. On HMDB51,

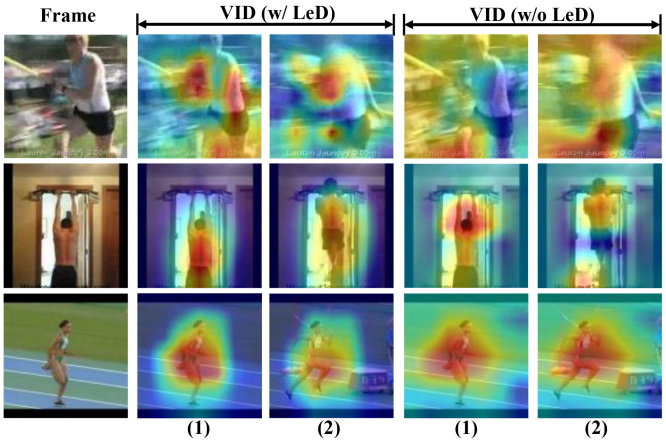


Fig. 5. Visualization of the heat map. The indices below indicate the corresponding subclips to which the frame belongs. The first row indicates that using LeD leads to a more stable concentration on motion areas, while the one without LeD may focus on the dynamic background instead.

the proposed VID surpasses VCOP [6] and VCP [60] by over 10% relatively. The improvement indicates that incoherence detection requires a more comprehensive understanding of videos compared to frame-order reasoning.

5) Visualization Comparison: We visualize heat maps of extracted representation to justify our proposed VID as shown in Figs. 5 and 6. Here, heat maps are generated based on Grad-CAM [67] utilizing the representations from the last convolution layer. To validate the regularization effect of LeD, we present corresponding heat maps from the network pretrained with or without LeD illustrated in Fig. 5. As shown in the last two rows, when there is a subtle difference between scenes of subclips, networks pretrained with or without LeD both focus on the actors to detect abnormal motion caused by incoherence, which proves that incoherence detection requires motion understanding. When scenes change intensively as shown in the first row, however, the network without LeD is distracted by the dynamic background. Yet the network with LeD maintains its concentration on motion areas, which justifies that the utilization of LeD increases the robustness of VID toward the severe changes of low-level information, which can therefore avoid trivial learning.

To justify the overall effectiveness of our proposed VID, more heat maps are presented in Fig. 6. Each row is augmented frames extracted from the test sample of UCF101 [54]. The first column is the original frame representing the input sample and the following columns are heat maps visualized by Grad-CAM [67] of different subclips. The heat maps provided in Fig. 6 justify our assumption that incoherence detection requires an understanding of motion in videos. For example, given the golf swing in the first row, the network pretrained with our VID concentrates on the upper body of the actor to detect the movements for incoherence detection. Additionally, when there are intensive changes between scenes of different subclip, our VID can maintain its concentration on the motion areas to detect incoherence. For instance, given input indicating horse racing in the last row, the network pretrained with our VID continuously focuses on the horses and riders regardless of the intensive changes of backgrounds.

C. Evaluation of Self-Supervised Representation

1) Action Recognition: To verify the effectiveness of our proposed VID, we evaluate our VID with different backbones on action recognition, which is a primary downstream task adopted in prior works [10], [11], [12]. For action recognition, the network is initialized with the pretrained weights while the fully connected layer is randomly initialized. During the fine-tuning stage, the training split 1 of UCF101 and HMDB51 are applied to fine-tune all parameters. The whole network is trained using the cross-entropy loss with an initial learning rate of 0.003. Other augmentations and parameter settings are the same as in the pretraining stage. For testing, following the same evaluation protocol of previous works [10], [12], we uniformly sample ten clips from each video followed by a center crop. The final predictions for each video are the average result of all sampled clips.

As shown in Table IV, our proposed VID achieves SOTA results, outperforming all previous spatiotemporal reasoning methods built upon a single data transformation. For C3D, while achieving a competitive performance on UCF101 with a marginal gap of 0.7% compared to DBA [42], our VID surpasses the performance of DBA [42] on HMDB51 for more than 3.5%. For R3D, VID exceeds PRP [12] and ST-Puzzle [41], which are the previous SOTA method on UCF101 and HMDB51, with noticeable margins of 7.1% on UCF101 and 5.4% on HMDB51. With R(2+1)D pretrained on UCF101, VID achieves competitive performance with a minor improvement of 0.3% on UCF101 compared to the previous SOTA method STS [40]. It is worth mentioning that STS [40] utilizes RGB and estimated optical flow as an additional modality during the pretrained stage, while our VID achieves better performance utilizing pure RGB. When pretrained on Kinetics-400, the margins of improvement further expand to 0.7% and 1.4%, respectively. The noticeable performance improvements justify that VID can learn more abundant spatiotemporal representation compared to previous single-transformation methods.

In Table IV, we additionally include the results of RTT [11] which assembles multiple transformations, leading to superior performance compared to single-transformation methods. Nevertheless, for C3D, VID is the only single-transformation method that outperforms RTT by 0.3% on UCF101 and provides competitive performance on HMDB51. It is possible to further improve the performance of ensemble-based methods by including our VID, while we mainly focus on leveraging video coherence by using a single temporal transformation in this work. Moreover, some previous SOTA methods that utilize different evaluation protocols are also illustrated at the bottom section of Table IV, where MemDPC [5] utilizes a deeper backbone R2D3D-34 and CAVP [65] adopts higher input resolutions and a more complex network. While CSJ [64] outperforms our VID on HMDB51 when pretrained on K400 with a backbone of the same depth, our VID significantly surpasses its performance when pretrained on UCF101, with a performance gap of 7.4% and 5.5% on UCF101 and HMDB51, respectively. Moreover, when compared to some previous methods based on multimodalities (e.g., STS [40]

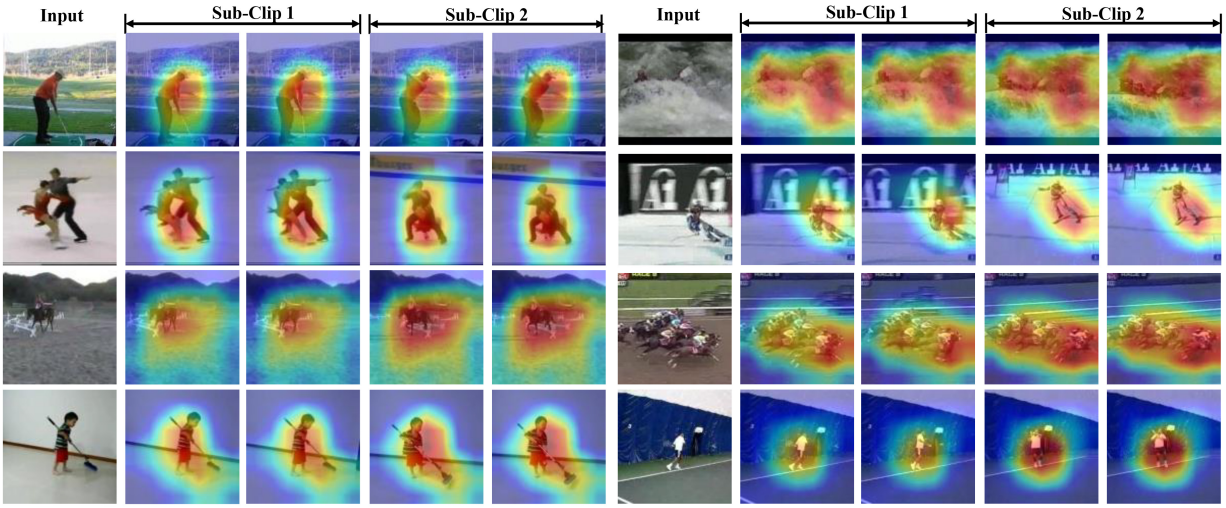


Fig. 6. Heat map visualization of our VID. The column "Input" contains original frames from the test samples. The following frames are sampled from different subclips. Our visualization shows that the network concentrates on the motion areas rather than backgrounds when pretrained with our VID.

TABLE IV
PERFORMANCE OF ACTION RECOGNITION COMPARED TO PREVIOUS METHODS. THE FORMER FIGURE OF INPUT SIZE INDICATES THE INPUT FRAME NUMBER, WHILE THE LATTER REPRESENT THE SPATIAL RESOLUTION. RTT [11] IS A SOTA METHOD UTILIZING MULTIPLE DATA TRANSFORMATIONS

Method	Experimental Settings				Downstream Tasks	
	Network	Input Size	Pre-trained	Single-Mod	UCF101(%)	HMDB51(%)
VCOP [6]	C3D	$16 * 112$	UCF101	✓	65.6	28.4
VCP [60]	C3D	$16 * 112$	UCF101	✓	68.5	32.5
PRP [12]	C3D	$16 * 112$	UCF101	✓	69.1	34.5
PMAS [13]	C3D	$16 * 112$	K-400	✓	58.8	32.6
DBA [42]	C3D	$16 * 112$	UCF101	✓	**70.9**	34.2
VID (Ours)	C3D	$16 * 112$	UCF101	✓	70.2	**37.7**
VID (Ours)	C3D	$16 * 112$	K400	✓	70.4	37.1
VCOP [6]	R3D	$16 * 112$	UCF101	✓	64.9	29.5
VCP [60]	R3D	$16 * 112$	UCF101	✓	66.0	31.5
ST-puzzle [41]	R3D	$16 * 112$	K-400	✓	65.8	33.7
PRP [12]	R3D	$16 * 112$	UCF101	✓	66.5	29.7
VID (Ours)	R3D	$16 * 112$	UCF101	✓	73.6	38.0
VID (Ours)	R3D	$16 * 112$	K400	✓	**73.9**	**38.4**
VCOP [6]	R(2+1)D	$16 * 112$	UCF101	✓	72.4	30.9
VCP [60]	R(2+1)D	$16 * 112$	UCF101	✓	66.3	32.2
PRP [12]	R(2+1)D	$16 * 112$	UCF101	✓	72.1	35.0
PP [10]	R(2+1)D	$16 * 112$	K-400	✓	77.1	36.6
STS [40]	R(2+1)D	$16 * 112$	UCF101	✗	77.8	40.1
VID (Ours)	R(2+1)D	$16 * 112$	UCF101	✓	78.1	40.1
VID (Ours)	R(2+1)D	$16 * 112$	K-400	✓	**78.5**	**41.5**
RTT [11]	C3D	$16 * 112$	K-600	✓	69.9	39.6
RTT [11]	R3D	$16 * 112$	UCF101	✓	77.3	47.5
RTT [11]	R(2+1)D	$16 * 112$	UCF101	✓	81.6	46.4
MemDPC [5]	R2D3D-34	$40 * 112$	K-400	✓	78.1	41.2
CSJ [64]	R2D3D-18	$16 * 112$	UCF101	✓	70.4	36.0
CSJ [64]	R2D3D-18	$16 * 112$	K-400	✓	76.2	46.7
CAVP [65]	I3D	$16 * 224$	K-400	✓	73.6	46.1
DSM [66]	I3D	$64 * 224$	K-400	✗	74.8	52.5

and DSM [66] utilizing additional trajectories and optical flow, respectively), our proposed VID achieves competitive performance with only RGB input. Specifically, despite utilizing a much smaller input resolution and pure RGB input only, our VID surpasses DSM [66] with a noticeable gap of 3.7% when evaluated on UCF101.

2) Video Retrieval: We further evaluate our VID on another downstream task of nearest-neighbor video retrieval, which evaluates the quality of features extracted by the self-supervised pretrained model. To make a fair comparison, our evaluation follows the protocol of previous SOTA methods [10], [12], where all models are pretrained on UCF101. Given ten 16-frame clips sampled from each video, their features are extracted from the last pooling layer of the pretrained backbone model. During the inference stage, frames of each clip are first resized to 128×171 and then centrally cropped to 112×112. Clips in the testing split are utilized to query the Top k nearest samples based on their corresponding features. Here, we consider k equal to 1, 5, 10, 20, and 50.

As shown in Table V, VID outperforms all previous spatiotemporal reasoning on most evaluation metrics of UCF101 and HMDB51 across all backbones. When evaluated on

TABLE V
PERFORMANCE OF VIDEO RETRIEVAL ON UCF101 AND HMDB51.∗: R3D HERE, REFERS TO R2D3D-18

Method	UCF101					HMDB51				
	Top1	Top5	Top10	Top20	Top50	Top1	Top5	Top10	Top20	Top50
C3D (VCOP [6])	12.5	29.0	39.0	50.6	66.9	7.4	22.6	34.4	48.5	70.1
C3D (VCP [60])	17.3	31.5	42.0	52.6	67.7	7.8	23.8	35.3	49.3	71.6
C3D (PRP [12])	23.2	38.1	46.0	55.7	68.4	10.5	27.2	40.4	56.2	75.9
C3D (PP [10])	20.0	37.4	46.9	58.5	73.1	8.0	25.2	37.8	54.4	77.5
C3D (DBA [42])	18.6	37.3	47.8	60.1	75.7	8.8	29.5	**43.5**	**59.4**	**79.6**
C3D (DSM [66])	16.8	33.4	43.4	54.6	70.7	8.2	25.9	38.1	52.0	75.0
C3D (Ours)	**26.9**	**43.6**	**53.6**	**63.8**	**78.2**	**11.6**	**29.6**	43.3	58.4	77.3
R3D (VCOP [6])	14.1	30.3	40.4	51.1	66.5	6.7	22.9	34.4	48.8	68.9
R3D (VCP [60])	18.6	33.6	42.5	53.5	68.1	7.6	24.4	36.3	53.6	76.4
R3D (PRP [12])	22.8	38.5	46.7	55.2	69.1	8.2	25.8	38.5	53.3	75.9
R3D (PP [10])	19.9	36.2	46.1	55.6	69.8	8.2	24.2	37.3	53.3	74.5
R3D* (CSJ [64])	21.5	40.5	53.4	**64.9**	-	-	-	-	-	-
R3D* (MemDPC [5])	20.2	40.4	52.4	64.7	-	7.7	25.7	40.6	57.7	-
R3D (Ours)	**26.4**	**44.5**	**54.1**	63.9	**78.2**	**11.2**	**32.2**	**45.4**	**59.8**	**79.2**
R(2+1)D (VCOP [6])	10.7	25.9	35.4	47.3	63.9	5.7	19.5	30.7	45.8	67.0
R(2+1)D (VCP [60])	19.9	33.7	42.0	50.5	64.4	6.7	21.5	32.7	49.2	73.3
R(2+1)D (PRP [12])	20.3	34.0	41.9	51.7	64.2	8.2	25.3	36.2	51.0	73.0
R(2+1)D (PP [10])	17.9	34.3	44.6	55.5	72.0	10.1	24.6	37.6	54.4	**77.1**
R(2+1)D (Ours)	**22.0**	**40.4**	**51.2**	**61.8**	**74.7**	**10.4**	**27.9**	**42.7**	**58.1**	76.7

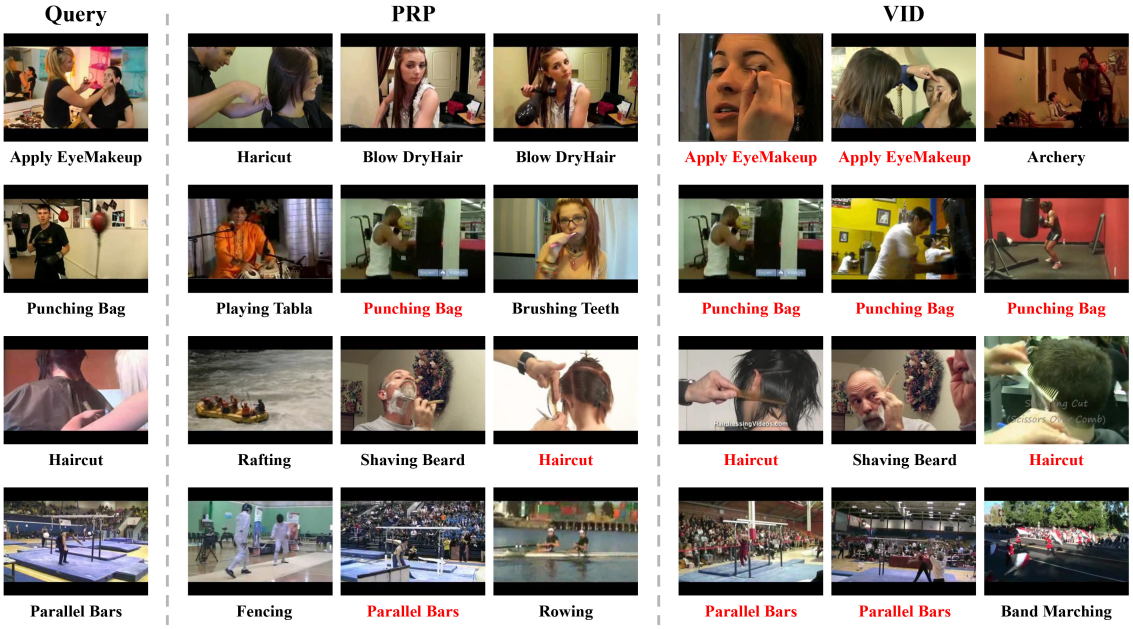


Fig. 7. Visualization of video retrieval results of our VID and previous PRP [12]. The figures in the first column are queries. For each query, we present the Top3 retrieval results of our VID and the previous SOTA PRP [12]. Action classes in red represent the correct retrieval results. Compared to PRP [12], the network pretrained with our VID can retrieve more samples from query action categories.

UCF101 with the same backbone, our VID achieves better performance compared to previous methods on all evaluation metrics. Specifically, VID surpasses the previous SOTA method PRP [12] by at least 1.7% for Top1 accuracy on UCF101, while the improvement further increases to 3.7% when both methods adopt C3D as their backbones. More specifically, VID surpasses one of the previous SOTA methods PRP [12] by at least 1.7% for Top1 accuracy on UCF101, while the improvement further increases to 3.7% when both methods adopting C3D as their backbones. Particularly, although the performance of VID on Top20 is outperformed by CSJ [64] with a gap of 1.0% when adopting C3D, our VID still surpasses CSJ across all other evaluation matrices on UCF101. When tested on HMDB51, the proposed VID similarly exceeds previous methods on most evaluation metrics. Compared to PRP [12] and PP [10], our method achieves better performance with a margin of 0.3%–3.6% for Top1 accuracy across three different backbones. The performance of improvement further expands to more than 2.4% when k varies from 5 to 20. It is worth noting that though our VID is surpassed by DBA [42] on the Top 10–50 matrices of HMDB51 with C3D, the performance gap is relatively trivial compared to our improvement on the Top1 and the Top5. Furthermore, our VID achieves superior performance with noticeable improvement across all evaluation matrices on UCF101. The noticeable overall improvement further justifies that our VID extracts more effective spatiotemporal representation for downstream tasks compared to previous methods.

We further present multiple examples of video retrieval results in comparison to the previous SOTA method PRP [12] as a qualitative study. Both our VID and PRP [12] are evaluated with R3D-18. Illustrated in Fig. 7, our proposed VID provide more reasonable results compared to previous PRP [12]. For example, given the query of applying eye makeup as shown in the first row, our VID retrieves two samples of the same action classes as the query among Top3, while PRP retrieves samples that belong to similar-yet-incorrect actions, such as haircut and blow-dry hair. The retrieval results indicate that the network pretrained with our VID obtains a more comprehensive understanding of videos compared to previous methods.

V. Conclusion and Future Works

In this article, we propose a novel self-supervised method based on VID for video representation learning. The incoherent clip is generated as the concatenation of subclips sampled from the same video with incoherence between each other. By detecting the location and length of incoherence, the network can extract effective spatiotemporal features. The ICL is developed to maximize the mutual information between subclips from the same raw video. Extensive experiments show that VID achieves SOTA performance with significant margins compared to previous methods. The proposed VID reveals a new perspective to leverage video coherence for video representation learning.

Despite the improvement of VID compared to previous methods, there is still room for further improvement. While this article proposes to generate each incoherent clip with two subclips, it is possible to further elaborate this method by using more subclips. Since VID is delicately designed to avoid loop-back sampling, the number of input frames is restricted by the minimum frames of samples in the dataset as well as the incoherence length. One of the main challenges is that the lower bound of the raw frame number that satisfies our requirements is increasing when we introduce more subclips. In the future, the performance of VID can be further advanced by introducing more flexible frame sampling methods or using datasets with more raw video frames.

References

[1] C. Vondrick, H. Pirsiavash, and A. Torralba, "Generating videos with scene dynamics," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst.*, 2016, pp. 613–621.

[2] N. Srivastava, E. Mansimov, and R. Salakhutdinov, "Unsupervised learning of video representations using LSTMs," in *Proc. 32nd Int. Conf. Int. Conf. Mach. Learn.*, vol. 37, 2015, pp. 843–852.

[3] C. Gan, B. Gong, K. Liu, H. Su, and L. J. Guibas, "Geometry guided convolutional neural networks for self-supervised video representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 5589–5597.

[4] Y. Tian, Z. Che, W. Bao, G. Zhai, and Z. Gao, "Self-supervised motion representation via scattering local motion cues," in *Proc. 16th Eur. Conf. Comput. Vis. (ECCV)*, Glasgow, U.K., 2020, pp. 71–89.

[5] T. Han, W. Xie, and A. Zisserman, "Memory-augmented dense predictive coding for video representation learning," in *Proc. 16th Eur. Conf. Comput. Vis. (ECCV)*, Glasgow, U.K., 2020, pp. 312–329.

[6] D. Xu, J. Xiao, Z. Zhao, J. Shao, D. Xie, and Y. Zhuang, "Self-supervised spatiotemporal learning via video clip order prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 10334–10343.

[7] B. Fernando, H. Bilen, E. Gavves, and S. Gould, "Self-supervised video representation learning with odd-one-out networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3636–3645.

[8] I. Misra, C. L. Zitnick, and M. Hebert, "Shuffle and learn: Unsupervised learning using temporal order verification," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 527–544.

[9] H.-Y. Lee, J.-B. Huang, M. Singh, and M.-H. Yang, "Unsupervised representation learning by sorting sequences," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 667–676.

[10] J. Wang, J. Jiao, and Y.-H. Liu, "Self-supervised video representation learning by pace prediction," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 504–521.

[11] S. Jenni, G. Meishvili, and P. Favaro, "Video representation learning by recognizing temporal transformations," in *Proc. 16th Eur. Conf. Comput. Vis*, Aug. 2020, pp. 425–442.

[12] Y. Yao, C. Liu, D. Luo, Y. Zhou, and Q. Ye, "Video playback rate perception for self-supervised spatio-temporal representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1–10.

[13] J. Wang, J. Jiao, L. Bao, S. He, Y. Liu, and W. Liu, "Self-supervised spatio-temporal representation learning for videos by predicting motion and appearance statistics," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4006–4015.

[14] L. Shao, X. Zhen, D. Tao, and X. Li, "Spatio-temporal Laplacian pyramid coding for action recognition," *IEEE Trans. Cybern.*, vol. 44, no. 6, pp. 817–827, Jun. 2014.

[15] L. Liu, L. Shao, X. Li, and K. Lu, "Learning spatio-temporal representations for action recognition: A genetic programming approach," *IEEE Trans. Cybern.*, vol. 46, no. 1, pp. 158–170, Jan. 2016.

[16] Y. Bin, Y. Yang, F. Shen, N. Xie, H. T. Shen, and X. Li, "Describing video with attention-based bidirectional LSTM," *IEEE Trans. Cybern.*, vol. 49, no. 7, pp. 2631–2641, Jul. 2019.

[17] B. Sheng, P. Li, R. Ali, and C. P. Chen, "Improving video temporal consistency via broad learning system," *IEEE Trans. Cybern.*, vol. 52, no. 7, pp. 6662–6675, Jul. 2021.

[18] J. Bai et al., "Two-stream spatial–temporal graph convolutional networks for driver drowsiness detection," *IEEE Trans. Cybern.*, vol. 52, no. 12, pp. 13821–13833, Dec. 2022.

[19] N. Behrmann, J. Gall, and M. Noroozi, "Unsupervised video representation learning by bidirectional feature prediction," in *Proc. IEEE/CVF Winter Conf. Appl. of Comput. Vis.*, 2021, pp. 1670–1679.

[20] C.-Y. Zhang, Y.-Y. Xiao, J.-C. Lin, C. L. P. Chen, W. Liu, and Y.-H. Tong, "3-D deconvolutional networks for the unsupervised representation learning of human motions," *IEEE Trans. Cybern.*, vol. 52, no. 1, pp. 398–410, Jan. 2022.

[21] J. Zhang, Y. Han, J. Tang, Q. Hu, and J. Jiang, "Semi-supervised image-to-video adaptation for video action recognition," *IEEE Trans. Cybern.*, vol. 47, no. 4, pp. 960–973, Apr. 2017.

[22] M. Luo, X. Chang, L. Nie, Y. Yang, A. G. Hauptmann, and Q. Zheng, "An adaptive semisupervised feature analysis for video semantic recognition," *IEEE Trans. Cybern.*, vol. 48, no. 2, pp. 648–660, Feb. 2018.

[23] C. Huang et al., "Weakly supervised video anomaly detection via self-guided temporal discriminative transformer," *IEEE Trans. Cybern.*, early access, Dec. 29, 2022, doi: 10.1109/TCYB.2022.3227044.

[24] Z. Gao, Y. Zhao, H. Zhang, D. Chen, A.-A. Liu, and S. Chen, "A novel multiple-view adversarial learning network for unsupervised domain adaptation action recognition," *IEEE Trans. Cybern.*, vol. 52, no. 12, pp. 13197–13211, Dec. 2022.

[25] Y. Xu, J. Yang, H. Cao, K. Wu, M. Wu, and Z. Chen, "Source-free video domain adaptation by learning temporal consistency for action recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 147–164.

[26] Y. Xu, H. Cao, Z. Chen, X. Li, L. Xie, and J. Yan, "Video unsupervised domain adaptation with deep learning: A comprehensive survey," 2022, *arXiv:2211.10412*.

[27] R. Caruana and V. de, "Promoting poor features to supervisors: Some inputs work better as outputs," in *Advances in Neural Information Processing Systems*, vol. 9, M. C. Mozer, M. Jordan, and T. Petsche, Eds. Cambridge, MA, USA: MIT Press, 1997. [Online]. Available: https://proceedings.neurips.cc/paper/1996/file/6c14da109e294d1e8155be8aa4b1ce8e-Paper.pdf

[28] R. K. Ando, T. Zhang, and P. Bartlett, "A framework for learning predictive structures from multiple tasks and unlabeled data," *J. Mach. Learn. Res.*, vol. 6, no. 11, pp. 1817–1853, 2005.

[29] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3733–3742.

[30] R. D. Hjelm et al., "Learning deep representations by mutual information estimation and maximization," in *Proc. 7th Int. Conf. Learn. Represent.*, 2019, pp. 1–24. [Online]. Available: https://openreview.net/forum?id=Bklr3j0cKX

[31] I. Misra and L. V. D. Maaten, "Self-supervised learning of pretext-invariant representations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1–14.

[32] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.

[33] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "Albert: A lite bert for self-supervised learning of language representations," 2019, *arXiv:1909.11942*.

[34] T. Han, W. Xie, and A. Zisserman, "Video representation learning by dense predictive coding," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops*, 2019, pp. 1–10.

[35] A. V. D. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018, *arXiv:1807.03748*.

[36] C. Sun, A. Myers, C. Vondrick, K. Murphy, and C. Schmid, "VideoBERT: A joint model for video and language representation learning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 7464–7473.

[37] C. Sun, F. Baradel, K. Murphy, and C. Schmid, "Learning video representations using contrastive bidirectional transformer," 2019, *arXiv:1906.05743*.

[38] Z. Luo, B. Peng, D.-A. Huang, A. Alahi, and L. Fei-Fei, "Unsupervised learning of long-term motion dynamics for videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2203–2212.

[39] G. Wang, Y. Zhou, C. Luo, W. Xie, W. Zeng, and Z. Xiong, "Unsupervised visual representation learning by tracking patches in video," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 2563–2572.

[40] J. Wang, J. Jiao, L. Bao, S. He, W. Liu, and Y.-H. Liu, "Self-supervised video representation learning by uncovering spatio-temporal statistics," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 7, pp. 3791–3806, Jul. 2022.

[41] D. Kim, D. Cho, and I. S. Kweon, "Self-supervised video representation learning with space-time cubic puzzles," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, Jul. 2019, pp. 8545–8552. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/4873

[42] Z. Wang, C. Hou, G. Yue, and Q. Yang, "Dynamic-boosting attention for self-supervised video representation learning," *Appl. Intell.*, vol. 52, no. 3, pp. 3143–3155, 2022.

[43] P. Chen et al., "RSPNet: Relative speed perception for unsupervised video representation learning," in *Proc. AAAI Conf. Artif. Intell.*, vol. 1, 2021, pp. 1–10.

[44] P. Bachman, R. D. Hjelm, and W. Buchwalter, "Learning representations by maximizing mutual information across views," in *Advances in Neural Information Processing Systems*, vol. 32, H. Wallach, H. Larochelle, A. Beygelzimer, F. D' Alché-Buc, E. Fox, and R. Garnett, Eds. Red Hook, NY, USA: Curran Assoc., Inc., 2019. [Online]. Available: https://proceedings.neurips.cc/paper/2019/file/ddf354219aac374f1d40b7e760ee5bb7-Paper.pdf

[45] D. Dwibedi, Y. Aytar, J. Tompson, P. Sermanet, and A. Zisserman, "Temporal cycle-consistency learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 1801–1810.

[46] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 9729–9738.

[47] G. Lorre, J. Rabarisoa, A. Orcesi, S. Ainouz, and S. Canu, "Temporal contrastive pretraining for video action recognition," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2020, pp. 662–670.

[48] T. Yao, Y. Zhang, Z. Qiu, Y. Pan, and T. Mei, "SeCo: Exploring sequence supervision for unsupervised representation learning," in *Proc. 35th AAAI Conf. Artif. Intell.*, 2021, pp. 1–9.

[49] R. Qian et al., "Spatiotemporal contrastive video representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 6964–6974.

[50] T. Pan, Y. Song, T. Yang, W. Jiang, and W. Liu, "VideoMoCo: Contrastive video representation learning with temporally adversarial examples," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 11205–11214.

[51] T. Han, W. Xie, and A. Zisserman, "Self-supervised co-training for video representation learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 5679–5690.

[52] H. Akbari et al., "VATT: Transformers for multimodal self-supervised learning from raw video, audio and text," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 1–20.

[53] C. Feichtenhofer, H. Fan, B. Xiong, R. Girshick, and K. He, "A large-scale study on unsupervised spatiotemporal representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 3299–3309.

[54] K. Soomro, A. R. Zamir, and M. Shah, "UCF101: A dataset of 101 human actions classes from videos in the wild," 2012, *arXiv:1212.0402*.

[55] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "HMDB51: A large video database for human motion recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2556–2563.

[56] W. Kay et al., "The kinetics human action video dataset," 2017, *arXiv:1705.06950*.

[57] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2015, pp. 4489–4497.

[58] K. Hara, H. Kataoka, and Y. Satoh, "Can spatiotemporal 3D CNNs retrace the history of 2D CNNs and ImageNet?" in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6546–6555.

[59] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6450–6459.

[60] D. Luo et al., "Video cloze procedure for self-supervised spatio-temporal learning," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, 2020, pp. 11701–11708.

[61] H. Alwassel, D. Mahajan, B. Korbar, L. Torresani, B. Ghanem, and D. Tran, "Self-supervised learning by cross-modal audio-video clustering," 2019, *arXiv:1911.12667*.

[62] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proc. COMPSTAT*, 2010, pp. 177–186.

[63] A. Paszke et al., "Automatic differentiation in PyTorch," in *Proc. NIPS-W*, 2017, pp. 1–4.

[64] Y. Huo et al., "Self-supervised video representation learning with constrained spatiotemporal jigsaw," in *Proc. 30th Int. Joint Conf. Artif. Intell.*, Aug. 2021, pp. 751–757. [Online]. Available: https://doi.org/10.24963/ijcai.2021/104

[65] Y. Lin, J. Wang, M. Zhang, and A. J. Ma, "Learning spatio-temporal representation by channel aliasing video perception," in *Proc. 29th ACM Int. Conf. Multimedia*, 2021, pp. 2317–2325.

[66] J. Wang et al., "Enhancing unsupervised video representation learning by decoupling the scene and the motion," in *Proc. AAAI Conf. Artif. Intell.*, vol. 35, 2021, pp. 10129–10137.

[67] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 618–626.

Haozhi Cao received the B.Eng. degree from the School of Electrical Engineering and Automation, Wuhan University, Wuhan, China, in 2019, and the M.Eng. degree from the School of Electrical and Electronic Engineering, Nanyang Technological University (NTU), Singapore, in 2021, where he is currently pursuing the Ph.D. degree.

He is also working as a Research Associate with the Centre for Advanced Robotics Technology, NTU. His research interests include deep learning with applications in video understanding, transfer learning, and multimodal learning.

Yuecong Xu (Member, IEEE) received the B.Eng. degree from the School of Electrical and Electronic Engineering, Nanyang Technological University (NTU), Singapore, in 2017, and the Ph.D. degree from NTU in 2021.

He was the receiver of the Nanyang President's Graduate Scholarship. He was the Co-Organizer of the UG2+ Challenge for Computational Photography and Visual Recognition, held in conjunction with CVPR 2021 and CVPR 2022. He is currently a Research Scientist with the Institute for Infocomm Research, A*STAR, Singapore, and a Lecturer with NTU. His research focuses on video understanding and analysis based on deep learning and transfer learning.

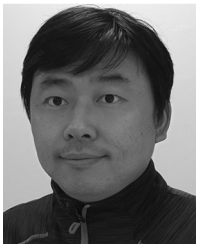
Kezhi Mao (Member, IEEE) received the B.Eng. degree from Jinan University, Jinan, China, in 1989, the M.Eng. degree from Northeastern University, Shenyang, China, in 1992, and the Ph.D. degree from the University of Sheffield, Sheffield, U.K., in 1998.

He was a Lecturer with Northeastern University from March 1992 to May 1995, a Research Associate with the University of Sheffield from April 1998 to September 1998, a Research Fellow with the Nanyang Technological University, Singapore, from September 1998 to May 2001, and an Assistant Professor with the School of Electrical and Electronic Engineering, Nanyang Technological University, from June 2001 to September 2005. He has been an Associate Professor since October 2005. His areas of interests include computational intelligence, pattern recognition, text mining, and knowledge extraction, cognitive science, and big data and text analytics.

Lihua Xie (Fellow, IEEE) received the B.E. and M.E. degrees in electrical engineering from the Nanjing University of Science and Technology, Nanjing, China, in 1983 and 1986, respectively, and the Ph.D. degree in electrical engineering from the University of Newcastle, Callaghan, NSW, Australia, in 1992.

Since 1992, he has been with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, where he is currently a Professor and served as the Head of Division of Control and Instrumentation from July 2011 to June 2014. He held teaching appointments with the Department of Automatic Control, Nanjing University of Science and Technology from 1986 to 1989, and Changjiang Visiting Professorship with South China University of Technology, Guangzhou, China, from 2006 to 2011. His research interests include robust control and estimation, networked control systems, multiagent control, and unmanned systems.

Prof. Xie has served as an Editor for IET Book Series in Control and an Associate Editor of a number of journals, including IEEE TRANSACTIONS ON AUTOMATIC CONTROL, *Automatica*, IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY, and IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS-II. He is a Fellow of the Academy of Engineering Singapore, IFAC, and Chinese Automation Association.



Jianxiong (Terry) Yin received the bachelor's from South China University of Technology, Guangzhou, China, 2009, and the master's degree from Yonsei University, Seoul, South Korea, in 2012 respectively.

He is currently a Senior Deep Learning Solutions Architect with NVIDIA AI Technology Center, Singapore. He was a Researcher with Nanyang Technological University (NTU), Singapore, from 2012 to 2016, during which he received ASEAN ICT Awards Gold Award, Data center Dynamics Award, ACM SIGCOMM 2013 travel grant and GTC 2015 presenter grant for his research work in cloud data center energy optimization, data center digital twin research, and deep learning compute system architecture.

Simon See received the Ph.D. degree in applied mathematics/engineering from the University of Salford, Salford, U.K.

He is currently the Senior Director and the Chief Solutions Architect with NVIDIA Asia–Pacific Professional Solution Group. He is also a Professor with Shanghai Jiaotong University, Shanghai, China, and Mahindra University, Hyderabad, India. He is also the Chief Scientific Computing Advisor to BGI Group, Shenzhen, China. Prior to NVIDIA, he was with the DSO National Laboratory, IBM, SGI, and Sun Microsystems. His research interests are computer architecture and systems, simulation, and applied mathematics.

Qianwen Xu (Member, IEEE) received the B.Sc. degree in electrical engineering from Tianjin University, Tianjin, China, in 2014, and the Ph.D. degree in electrical engineering from Nanyang Technological University, Singapore, in 2018.

During 2018–2020, she was a Postdoctoral Research Fellow with Aalborg University, Aalborg, Denmark, a Visiting Researcher with Imperial College London, London, U.K., and a Wallenberg-NTU Presidential Postdoctoral Fellow with Nanyang Technological University, Singapore. She is currently an Assistant Professor with the Department of Electric Power and Energy Systems, KTH Royal Institute of Technology, Stockholm, Sweden. Her research interests include advanced control, optimization, and AI application for microgrid and smart grid.

Dr. Xu was a recipient of Humboldt Research Fellowship, Excellent Doctorate Research Work in Nanyang Technological University, Best Paper Award in IEEE PEDG 2020, and Nordic Energy Award 2022. She serves as the Vice Chair for the IEEE Power and Energy Society and Power Electronics Society, Sweden Chapter, and an Associate Editor for the IEEE TRANSACTIONS ON SMART GRID and IEEE JOURNAL OF EMERGING AND SELECTED TOPICS IN POWER ELECTRONICS.

Jianfei Yang (Member, IEEE) received the B.Eng. degree from the School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China, in 2016, and the Ph.D. degree from Nanyang Technological University (NTU), Singapore, in 2021.

He used to work as a Senior Research Engineer with the University of California at Berkeley, Berkeley, CA, USA. He is currently a Presidential Postdoctoral Research Fellow and an Independent Principal Investigator with NTU. His research focuses on Artificial Intelligence of Things, such as wireless sensing and computer vision based on deep learning and transfer learning.

Dr. Yang received the Best Ph.D. Thesis Award from NTU. He won many International AI challenges in computer vision and interdisciplinary research fields.