This ICCV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

On Compositions of Transformations in Contrastive Self-Supervised Learning

Mandela Patrick[1,2*] Yuki M. Asano[2*] Polina Kuznetsova[1] Ruth Fong[2]
João F. Henriques[2] Geoffrey Zweig[1] Andrea Vedaldi[1,2]
[1] Facebook AI Research
mandelapatrick@facebook.com
[2] Visual Geometry Group, University of Oxford
yuki@robots.ox.ac.uk

Abstract

In the image domain, excellent representations can be learned by inducing invariance to content-preserving transformations via noise contrastive learning. In this paper, we generalize contrastive learning to a wider set of transformations, and their compositions, for which either invariance or distinctiveness is sought. We show that it is not immediately obvious how existing methods such as SimCLR can be extended to do so. Instead, we introduce a number of formal requirements that all contrastive formulations must satisfy, and propose a practical construction which satisfies these requirements. In order to maximise the reach of this analysis, we express all components of noise contrastive formulations as the choice of certain generalized transformations of the data (GDTs), including data sampling. We then consider videos as an example of data in which a large variety of transformations are applicable, accounting for the extra modalities – for which we analyze audio and text – and the dimension of time. We find that being invariant to certain transformations and distinctive to others is critical to learning effective video representations, improving the state-of-the-art for multiple benchmarks by a large margin, and even surpassing supervised pretraining. Code and pretrained models are available[1].

1. Introduction

Works such as MoCo [31], SimCLR [13], SwAV [12] and BYOL [25] have shown that it is possible to pre-train state-of-the-art image representations without the use of any manually-provided labels. Furthermore, many of these approaches use variants of noise contrastive learning [26, 27].

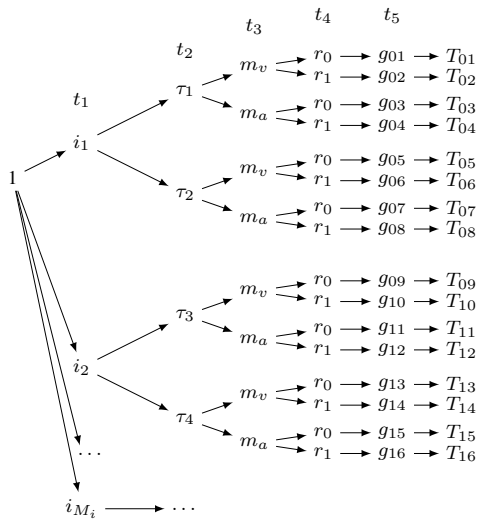

Fig. 1: Hierarchical sampling process of generalized data transformations (GDTs). Shown here are the five transformations analyzed for the audio-visual training case and their compositions: data-sampling (t_1), time-shift (t_2), modality splicing (t_3), time-reversal (t_4), and augmentation transformations, g (t_5) to learn video representations via noise contrastive learning.

Their idea is to learn a representation that is *invariant* to transformations that leave the meaning of an image unchanged (e.g. geometric distortion or cropping) and *distinctive* to changes that are likely to alter its meaning (e.g. replacing an image with another chosen at random).

These prior works have also shown that the choice of transformations is of primary importance for performance [12, 13]. This is not just a matter of selecting a

*Joint first authors
[1]https://github.com/facebookresearch/GDT

certain type of transformation, but also to specify how different transformations should be composed, and how these compositions should be sampled to from batches for training the model. So far, these choices have been mostly driven by intuition, with little formal understanding of why certain choices may be preferable, and how these choices can be generalized.

In this work, we answer some of these questions via a formal analysis of *composable transformations in contrastive learning*. Our analysis shows how invariance and distinctiveness to individual transformations can be meaningfully incorporated in the same learning formulation. It also provides some principles to guide the construction of the training batches. We interpret existing sampling schemes, such as the one in SimCLR, as special cases with certain potential advantages and disadvantages. We do so by showing how these constructions can be extended systematically to any composition of invariant and distinctive transformations.

Furthermore, we demonstrate the utility of our analysis by exploring contrastive methods for learning representations of video data. Compared to images, videos contain a time dimension and multiple modalities, which have been shown to provide effective learning cues; for instance [60] leverages multiple modalities, and [15, 41] the time dimension. We show how these effects can be incorporated in a uniform manner in contrastive learning by considering a suitable class of *generalized data transformations* (GDTs). GDTs capture standard augmentations, as well as temporal transformations, modality slicing and data sampling. The advantages of using GDTs is that they allow us to base the entire design of the learning formulation (e.g., how to write a coherent learning objective and how to sample batches) on a small number of design principles that our analysis has identified.

With this, we make some notable findings for contrastive video representation learning. First, we show that using this wider class of transformations greatly exceeds the performance that can be obtained by a vanilla applications of image-centric methods such as SimCLR to video data. By leveraging time and multiple modalities, we obtain large performance gains, almost *doubling* the performance. Second, we show that just learning representations that are invariant to more and more transformations is *not* optimal, at least when it comes to video data; instead, combining invariance to certain factors with distinctiveness to others performs better. To the best of our knowledge, this is the first time such an effect has been demonstrated in contrastive learning.

We also set the new state of the art in audio-visual representation learning, with both small and large video pretraining datasets on a variety of downstream tasks. In particular, we achieve 94.1% and 67.4% on the standardized UCF-101 [67] and HMDB-51 [42] action recognition benchmarks, when pretrained on HowTo100M [50], and 95.2% and 72.8% respectively when pretrained on IG65M [21].

2. Related work

Self-supervised learning from images and videos. A variety of pretext tasks have been proposed to learn representations from unlabelled **images**. Some tasks leverage the spatial context in images [17, 56] to train CNNs, while others create pseudo classification labels via artificial rotations [23], or clustering features [6, 10, 11, 12, 22, 37]. Colorization [83, 84], inpainting [62], solving jigsaw puzzles [57], as well as the contrastive methods detailed below, have been proposed for self-supervised image representation learning. Some of the tasks that use the space dimension of images have been extended to the space-time dimensions of **videos** by crafting equivalent tasks. These include jigsaw puzzles [40], and predicting rotations [38] or future frames [28]. Other tasks leverage the temporal dimension of videos to learn representations by predicting shuffled frames [53], the direction of time [76], motion [74], temporal ordering [43, 80], and playback speed [9, 14, 19]. These pretext-tasks can be framed as GDTs.

Multi-modal learning. Videos, unlike images, are a rich source of a variety of modalities such as speech, audio, and optical flow, and their correlation can be used as a supervisory signal. This idea has been present as early as 1994 [16]. Only recently, however, has multi-modal learning been used to successfully learn effective representations by leveraging the natural correspondence [2, 4, 5, 7, 54, 60] and synchronization [15, 41, 59] between the audio and visual streams. A number of recent papers have leveraged speech as a weak supervisory signal to train video representations [46, 49, 55, 68, 69] and recently [1], who use speech, audio and video. Other works incorporate optical flow and other modalities [29, 30, 64, 85] to learn representations. In CMC [70], representations are learned with different views such as different color channels or modalities to solely induce multi-view invariance. In contrast, our work extends this to and analyses multi-modal transformations and examines their utility as an invariant *or* distinctive learning signal.

Noise Contrastive Loss. Noise contrastive losses [26, 27] measure the similarity between sample pairs in a representational space and are at the core of several recent works on unsupervised feature learning. They yield good performance for learning image [13, 31, 33, 35, 45, 52, 58, 70, 71, 77] and video [3, 28, 34, 46, 49, 54, 66, 68, 82] representations, and circumvent the need to explicitly specify what information needs to be discarded via a designed task.

We leverage the noise contrastive loss as a learning framework to encourage the network to learn desired invariance and distinctiveness to data transformations. The GDT framework can be used to combine and extend many of these cues, contrastive or not, in a single noise contrastive formulation.

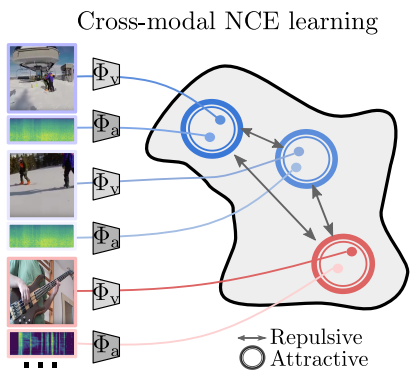

Fig. 2: **Example instantiation.** The embedding is learned via noise contrastive (NCE) learning. Here we show the case of audio-visual sample and time-shift distinctiveness: video-audio embeddings from the same video at the same time are pulled together, while audio-visual sample pairs from different videos and different starting times are pushed apart.

3. Method

We generalize contrastive methods such as CPC [58], PIRL [52], MoCo [31], SimCLR [13], and SwAV [12] to learn representations that can be invariant or distinctive to any number of transformations.

Given a collection x of data such as images or videos, we generate training samples

$$x(t_1, \dots, t_M) \in \mathcal{X}.$$

by applying a sequence of M transformations $T = (t_1, \dots, t_M)$ to the collection. We consider typical transformations such as data augmentations (e.g., randomly cropping an image). We also find it useful to express in the same manner other operations such as extracting a specific image or video from the collection or extracting a specific modality from a video. We call these *generalized data transformations* (GDTs).

To provide a concrete example, in a standard contrastive learning formulation such as SimCLR, the first transformation $t_1 = i \in \{1, \dots, |x|\}$ extracts an image x_i from the collection x and the second transformation $t_2 = g$ applies to it a random augmentation, so that we can write $x(t_1, t_2) = g(x_i)$. The goal is to learn a representation $\Phi : \mathcal{X} \to \mathbb{R}^d$ that identifies the image regardless of the augmentation; in other words, Φ should be invariant to the choice of t_2 and distinctive for the choice of t_1.

We wish to generalize this construction to richer data such as videos. Compared to images, videos contain multiple modalities and additional dimensions, which allows to consider qualitatively different transformations such as time shift, time reversal, and modality slicing. This generalization is however non-trivial. First, when considering $M > 2$ transformations, we have a choice of making the representation invariant or distinctive to each of them independently. For instance, video representations may benefit from being distinctive to time shift and/or time reversal rather than invariant to them. It is not immediately obvious how contrastive learning should be modified to incorporate these different choices. Another less apparent but important issue is how training data batches should be formed. Contrastive learning formulations minimize, in fact, a loss that involves comparing (contrasting) the representations of different samples, and is thus *not decomposable*. In practice, the loss is approximated by sampling batches of data, and how this is done has a major effect on the performance. In the previous example of SimCLR, if transformation (t_1, t_2) is included in the batch, so must be a complementary transformation (t_1, t_2') that differs only in the second factor $t_2 \neq t_2'$. This is required in order to learn the desired invariance. It also means that transformations in a batch cannot be sampled independently. A way to guarantee that both (t_1, t_2) and (t_1, t_2') are in the batch is to consider all possible combinations $\mathcal{T}_1 \times \mathcal{T}_2$ of two sets of transformations $\mathcal{T}_1$ and $\mathcal{T}_2$. However this is statistically inefficient because it applies the same augmentations $\mathcal{T}_2$ to all images in the batch. Instead, SimCLR samples at random $B/2$ images and then applies to them B independently sampled augmentations. This is better than the scheme above that would only use $B/|\mathcal{T}_1| = 2$ different augmentations. However, it is unclear how this strategy for sampling diverse transformations can be extended to $M > 2$ factors. This is studied next.

3.1. Compositional contrastive learning

Given a batch $\mathcal{T}$ of data transformations, we consider the learning objective:

$$\mathcal{L}(\Phi; \mathcal{T}) = -\sum_{T, T' \in \mathcal{T}} c(T, T') w(T, T') \cdot \log \left(\frac{e^{\langle \Phi(x(T)), \Phi(x(T')) \rangle / \rho}}{\sum_{T'' \in \mathcal{T}} w(T, T'')\, e^{\langle \Phi(x(T)), \Phi(x(T'')) \rangle / \rho}} \right). \quad (1)$$

where $\rho > 0$ is a temperature parameter. The *contrast function* $c(T, T') \in \{0, 1\}$ has the following interpretation: when $c(T, T') = 1$, then the representations $\Phi(x(T))$ and $\Phi(x(T'))$ are pulled together (invariance), and when $c(T, T') = 0$ they are pushed apart (distinctiveness). For example, in SimCLR, we set $c(T, T') = c((i, g), (i', g')) = \delta_{i=i'}$ to push apart the representations of different images (i, i') while remaining invariant to a transformation pair (g, g'). The *weight function* w is a second binary function that focuses learning on more informative transformation pairs; for instance, SimCLR sets $w(T, T') = \delta_{T \neq T'}$ to

avoid focusing learning invariance to identical transformation $T = T'$ as this is trivially satisfied. Next, we provide a semi-formal analysis of this formulation, leaving the details to Appendix A.1.

Multiple invariances and distinctiveness. The key to extending eq. (1) to $M > 2$ transformation is to build the function $c(T, T')$. We do this one factor a time. If we wish the representation to be distinctive to factor t_m, we set $c(t_m, t'_m) = \delta_{t_m = t'_m}$. If we wish it to be invariant to it, we set $c(t_m, t'_m) = 1$. In lemmas 4 and 5 (Appendix A.1), we show that, given these choices, the only consistent definition of $c(T, T')$ is the product $\prod_{m=1}^{M} c(t_m, t'_m)$. The intuition is as follows: The representation Φ should distinguish samples $x(T)$ and $x(T')$ if, and only if, at least one of the distinctive factors in T and T' differs.

Forming a batch. Given c, the remaining challenge is to sample training batches $\mathcal{T}$ appropriately. We start by deriving some requirements for $\mathcal{T}$ and then develop a sampling scheme that satisfies them (none of these are guaranteed by sampling T and T' independently). (i) First, in order for eq. (1) not to be identically zero, $c(T, T')$ should be non-zero for at least *some* choices of T and T' in the batch. (ii) Furthermore, when $c(T, T') = 1$, this should not be for the trivial case $T = T'$ (the one that SimCLR discounts by setting $w(T, T') = 0$). Based on the discussion above, the condition $c(T, T') = 1 \wedge T \neq T'$ means that all distinctive factors in T and T' agree and that at least an invariant factor differs. (iii) Additionally, for the fraction in eq. (1) not to be a constant, if $c(T, T') = 1$, there should be another T'' in the batch such that $c(T, T'') = 0$. The latter means that at least one distinctive factor in T and T'' differs.

Short of considering all possible combinations of transformations (which, as explained above, can be statistically inefficient), we can sample a batch $\mathcal{T}$ that satisfies these constraints as follows. We describe this process for the case of $M = 3$ transformations, but note that it extends immediately to any M (this is done in Appendix A.1.1 and A.1.2). First, we sample K_1 versions of the first distinctive transformations t_1. Then, for each t_1, we sample K_2 transformations t_2, also distinctive. Finally, for each choice of (t_1, t_2), we sample K_3 invariant transformations t_3.[2] We thus obtain a batch of $|\mathcal{T}| = K_1 K_2 K_3$ transformations.

This scheme has several desirable properties. First, for every $T = (t_1, t_2, t_3)$, there is another $T' = (t_1, t_2, t'_3)$ that agrees on the distinctive factors and differs in the invariant one (properties (i) and (ii)). Second, there is a $T'' = (t_1, t'_2, t'_3)$ or $T'' = (t'_1, t'_2, t'_3)$ that differs in one or more distinctive factors (property (iii)). Third, the construction is balanced, in the sense that the number of transformations that share a particular factor value t_m is the same for all values of t_m (this number is $|\mathcal{T}|/(K_1 \cdots K_m)$). Furthermore, SimCLR is obtained as a special case. Please see lemmas 6 and 7 (Appendix A.1) for an in-depth discussion.

Limitations. Despite the benefits, this scheme has also some limitations. The main issue is that a difference in factor t_m generally implies a difference in all subsequent factors as well, meaning that the representation may be unable to observe and thus learn to discriminate changes in all individual factors. In Appendix A.1.3, we show why this is unlikely to be an issue for the practical cases considered here and in the literature. However, we also suggest other practical cases where this *can* be a significant issue, affecting even methods such as SimCLR.

3.2. Properties of Generalized Data Transformations

In this section, we show that GDT's batch sampling strategy is statistically more efficient than naively-sampled pairs for contrastive learning. We do this by showing that GDT's objective has the same mean but a lower variance than sampling batches with eq. 1 directly, which would either enumerate all possible pairs of transformations (which is prohibitively expensive) or subsample it by sampling transformations independently. We assume that the distinctive transformations are injective. This must be approximately true, otherwise it is impossible for any method to be distinctive to such transformations. In fact, we can prove the following result:

Theorem 1. *Given a set of transformations $\mathcal{T}$, of which the distinctive transformations are injective, GDT is an unbiased estimate $\hat{\mathcal{L}}$ of the generalized contrastive loss (eq. 1), i.e. $\mathbb{E}[\hat{\mathcal{L}}] = \mathcal{L}(\Phi; \mathcal{T})$. Furthermore, consider a batch of sampled compositions of M transformations, with size $\prod_j^M K_j$, where K_m is the number of samples for the mth transformation. Define $K_I = \prod_{j \in I} K_j$ and $K_V = \prod_{j \in V} K_j$, where I and V are the subsets of indices corresponding to invariant and distinctive transformations, respectively. Denote by $\mathcal{L}_{jj'}$ and $\sigma^2_{jj'}$ the mean and variance of the partial sum of the objective (eq. 1) on the set $\mathcal{X}_j \times \mathcal{X}_{j'}$, with $\mathcal{X}_j = \{x(T^I, T^V_j) : T^I \in \mathcal{T}_I\}$, i.e. the sample pairs corresponding to distinctive transformations with indices j and j'. Then, the variance of the GDT estimate is*

$$\mathbb{V}[\hat{\mathcal{L}}] = \frac{1}{K_V^4 K_I^2} \sum_{jj'}^{K_V, K_V} \sigma^2_{jj'}.$$

The naive estimate's variance, on the other hand, is

$$\mathbb{V}[\hat{\mathcal{L}}_d] = \frac{1}{K_V^2 K_I^2} \sum_{jj'}^{K_V, K_V} \sigma^2_{jj'} + \frac{1}{K_V^2} \sum_{jj'}^{K_V, K_V} \left(\mathcal{L}_{jj'} - \mathcal{L}\right)^2,$$

which is larger by a multiplicative factor of K_V^2 and a further additive factor. *Proof:* See Appendix A. □

[2] Note that the sampling ordering is arbitrary; in particular, it needs not to be the same as the ordering in which transformations are applied to the data.

This states that sampling data with GDT yields reduced variance, resulting in higher-quality gradients for learning the *same* objective (since the estimate is unbiased), which is reflected empirically in our strong performance on numerous datasets and benchmarks. We note that this may apply to other methods built on the same sampling strategy but which compose transformations in different ways than GDT, as long as the requirements (i-iii) for forming a batch (sec. 3.1) are satisfied.

3.3. Application to video data

As a concrete instantiation of our framework, we consider video data and transformations of the type $T = (t_1, t_2, t_3, t_4, t_5) = (i, \tau, m, r, g)$, as shown in fig. 1, as follows. The first component i **selects** a video in the dataset. We sample $K_i \gg 2$ indices/videos and assume distinctiveness, so that $c(i, i') = \delta_{i=i'}$. The second component τ contrasts different **temporal shifts**. We sample $K_\tau = 2$ different values of a delay τ uniformly at random, extracting a 1s clip $x_{i\tau}$ starting at time τ. For this contrast, we will test the distinctiveness and invariance hypotheses, as [41] indicate that the former may be preferable. The third component m contrasts **modalities**, projecting the video $x_{i\tau}$ to either its visual or audio component $m(x_{i\tau})$. We assume invariance $c(m, m') = 1$ and always sample two such transformations m_v and m_a to extract both modalities, so $K_m = 2$. The fourth component r contrasts **time reversal** [63, 76], which has not previously been explored in a contrastive or cross-modal setting. This is given by a transformation $r \in \mathcal{R} = \{r_0, r_1\}$, where r_0 is the identity and r_1 flips the time dimension of its input tensor, so $K_r = 2$. The final component g applies a spatial and aural **augmentation** $x(T) = g(r(m(x_{i\tau})))$, also normalizing the data. We assume invariance $c(g, g') = 1$ and pick $K_g = 1$, i.e. augment each datum at this level in the sampling hierarchy. These choices lead to $K = K_i K_\tau K_m K_r K_g = 8K_i$ transformations T in the batch $\mathcal{T}$ (in ablations, we also test a subset of these choices).

While we focus on modality splitting, time reversal and shift, note that we could use any transformation that can yield a useful learning signal, such as speed [9, 14, 18, 36, 75, 81] and temporal ordering [20, 43, 53, 80].

Modality splitting. The modality splitting transformation m is useful to capture correlation between modalities [4, 7, 41, 60, 76]. Modality splitting means that the nature of the sample $x(i, \tau, m, r, g)$ is either a sequence of frames ($m = m_v$) or a sound ($m = m_a$). Formally, this means that $x(i, \tau, m, r, g)$ is an element of the direct sum $\mathcal{X}_v \oplus \mathcal{X}_a$ of visual and audio signals; likewise g, r and Φ are defined on this direct sum. In practice, this means that the transformation g comprises a pair of augmentations (g_v, g_a), where $g_v(v)$ extracts a fixed-size tensor by resizing to a fixed resolution of a random spatial crop of the input video v, and $g_a(a)$ extracts a spectrogram representation of the audio signal followed by SpecAugment [61] with frequency and time masking. Likewise, $\Phi = (\Phi_v, \Phi_a)$ comprises a pair of neural networks, one for each modality, both valued in $\mathbb{R}^d$ (refer to Appendix A.3.4 for architectural details). In the Appendix A.3.1, we show that modality splitting is key for performance; thus, we extend SimCLR weight function w to focus learning on only cross-modal pairs: $w(T, T) = \delta_{i \neq i} \cdot \delta_{m \neq m'}$.

3.4. Discussion: utility of GDT

With our framework, we can now generalize current state of the art contrastive learning approaches such as SimCLR in a systematic and practical manner. The theory above and in Appendix A.1 tells us what is the meaning of composing transformations, how a batch should be sampled and why, how this can be achieved by using a hierarchical sampling scheme that extends SimCLR, and what are the limitations of doing so. A particular benefit is to allow to specify individually, for each transformation, if invariance or distinctiveness is sought, whereas previous works lack this distinction and largely considered learning only invariances (SimCLR [13], AVID [54]), or distinctiveness (AoT [76]) to all factors. This property allows the flexible utilization of dataset specific transformations in the case of prior knowledge, or, as we have shown in this study, the exploration of useful signals by enumeration. Finding the best transformation signals can even be further optimized by methods such as Bayesian optimization. Finally, compared to a direct application of previous state-of-the-art methods image-based methods such as SimCLR [13], PIRL [52], and MoCo [31], we can also seamlessly incorporate important cues such as cross-modal correlation, greatly improving downstream performance (see table A.1).

4. Experiments

We compare self-supervised methods on pretraining audio-visual representations. Quality is assessed based on how well the pretrained representation transfers to downstream tasks. We conduct a study on video-audio, as well as video-text unsupervised representation learning to show the generality of our framework and then compare our best setup to the state of the art.

Self-supervised pretraining. For pretraining, we consider two standard pretraining datasets: Kinetics-400 [39] and HT100M [50] and use R(2+1)D-18 [72] and a 2D ResNet [32] as encoders (see Appendix for further details). We also explore how our algorithm scales to even larger, less-curated datasets and train on IG65M [21] as done in XDC [2].

Downstream tasks. To assess the pretrained representation f_v, we consider standard action recognition benchmark datasets, UCF-101 [67] and HMDB-51 [42]. We test the performance of our pretrained models on the tasks of finetuning

Table 1: **Learning hypothesis ablation.** Results on action classification performance on HMDB-51 is shown for finetuning accuracy (Acc) and frozen retrieval (recall@1) after pretraining on Kinetics-400 for 50 epochs. GDT can leverage signals from both invariance and stronger distinctiveness transformation signals. We consider data-sampling (DS), time-reversal (TR) and time-shifting (TS).

	DS	TR	TS	Mod.	**Acc.**	**R@1**
SimCLR-like: DS-distinctiveness only						
(a)	**d**	·	·	V	44.6	11.8
(b)	**d**	**i**	·	V	36.9	13.3
(c)	**d**	·	**i**	V	35.9	15.3
(d)	**d**	**i**	**i**	V	37.8	13.9
Cross-modal						
(e)	**d**	·	·	AV	52.4	21.8
(f)	**d**	**i**	·	AV	58.8	22.6
(g)	**d**	·	**i**	AV	57.4	23.5
(h)	**d**	**i**	**i**	AV	59.9	24.8
Cross-modal +1 distinctive factor						
(i)	**d**	**d**	·	AV	57.8	26.1
(j)	**d**	·	**d**	AV	58.7	22.1
(k)	**d**	**d**	**i**	AV	61.1	25.4
(l)	**d**	**i**	**d**	AV	**61.4**	**27.1**
Cross-modal + 2 distinctive factors						
(m)	**d**	**d**	**d**	AV	57.2	20.5

the pretrained representation, conducting few-shot learning and video action retrieval. The full details are given in the Appendix.

4.1. Analysis of generalized data transformations

In this section, we conduct an extensive study on each parameter of the GDT transformation studied here, $T = (i, \tau, m, r, g)$, and evaluate the performance by finetuning our network and conducting video retrieval on the HMDB-51 action recognition benchmark.

SimCLR-like baseline. First, we use the framework to test a direct extension of SimCLR to video data, as shown in Table 1(a)-(d). By this, we mean utilizing only the visual modality (V), and only invariance to transformations, which is standard in all recent self-supervised methods [13, 31, 77]. For this, we consider GDTs of the type $T = (i, m, \tau, r, g)$ described above and set $K_i = 512$ (the largest we can fit in our setup). In row (a), we pick only the video modality ($m = m_v$ so $K_m = 1$). We also sample a single shift τ (so $K_\tau = 1$), which results in data augmentation but does not learn shift invariance, and no time reversal $r = 1$ (so $K_r = 1$) — these are denoted with a $\cdot$ in the table. However, we do sample two visual augmentations g ($K_g = 2$), emulating SimCLR and learning invariance to that factor.

Table 2: **GDT on video-text HT100M dataset**. We also find the positive effect of including more modalities and find non-trivial combinations of beneficial transformations previously unexplored.

	DS	TR	TS	Mod.	**Acc**
SimCLR-like					
(a)	**d**	·	·	V	36.1
Video-text cross-modal					
(b)	**d**	·	·	VT	59.2
(c)	**d**	**d**	·	VT	61.5
(d)	**d**	·	**d**	VT	62.9
(e)	**d**	**d**	**i**	VT	63.8
(f)	**d**	**i**	**d**	VT	**64.4**
(g)	**d**	**d**	**d**	VT	**64.4**

We also set all transformation components to invariance ($c(t_m, t'_m) = 1$) except the first that does sample selection. In row (b-d) we also experiment with adding invariance to *time shift* (TS) and *time reversal* (TR), by setting $K_\tau = 2$ and $K_r = 2$. We find that doing so consistently degrades the finetuning accuracy performance, but increases the retrieval performance somewhat, indicating that the model is not able to fully leverage these augmentation signals in a meaningful way.

Cross-modal learning. Next, in rows (e-h) we repeat this experiment, but using both audio-visual modalities (AV) by setting $K_m = 2$. In this case, as explained above, we set the weight w to only consider cross-modal interactions and set $K_g = 1$. We note two facts: First, the performance increases substantially (+7.8% (e) vs (a-d)). Second, now TS and TR invariance leads to significant improvements (up to +7.5%).

Invariance vs distinctiveness. Next, in rows (i-l), we explore the effect of being invariant or distinctive to individual transformations, which is unique to our method. Comparing row (h) to rows (k) and (l), we see that switching to distinctiveness to one of TS or TR further improves performance (up to +1.5%). On the other hand, 'ignoring' either ($\cdot$ symbols in lines (g) and (j)) is worse than learning invariance ((h) and (l)), with a difference of around 2.5%. Finally, in row (m) we find that being distinctive for both TS and TR at the same time is worse, suggesting that a mix of distinctiveness and invariance is preferable. This is particularly true for the retrieval metric (column R@1).

4.2. Textual modality

In table 2, we demonstrate the generality of our approach by using ASR captions as an alternative modality (instead of audio) for the HowTo100M dataset [50]. For the text encoder, we use a simple Word2Vec [51] embedding with a MLP (further details are provided in the Appendix). Com-

paring table 2(a) with (b), we find that switching from SimCLR to a cross-modal baseline increases performance by more than +22%. Furthermore, we find gains of 3.7% when switching from data-sampling distinctiveness only (row (b)) to incorporating further distinctivenesses (rows c-d). Finally, we find that – as in the video-audio case – combining time-shift distinctiveness with time-reversal invariance leads to particularly strong representations (row (f)), yielding benefits of over +5% compared to data-sampling distinctiveness alone. Compared to video-audio learning (table 1(m)), we find the case of distinctive-only for video-text learning (table 2(g)) to be highly competitive, highlighting the need to explore the set of possible transformation signals to achieve the best downstream performance.

Intuition. While we only analyse a subset of possible transformations for video data, we nevertheless find consistent signals across both video-audio and video-text learning: Inclusion of further distinctivenesses to TS and TR always improve upon the basecase and the best setup is achieved for TS distinctiveness and TR invariance. One explanation for this might be that there is useful signal in both of these transformations that are not captured by previous "augmentation-only" naive noise-contrastive formulations. For example, for time-shift (TS), the model profits from having to differentiate different points in time, e.g. between an athlete running vs an athlete landing in a sandpit, which could be both in the same video. This intuitively serves as a hard negative for the model, increasing its discriminative power. For time reversal (TR), many actions depicted such as moving an object are inherently invariant to reversing time, as shown in [65], therefore yielding a gain when used as an augmentation. In [76], they show that humans have a 20% error-rate when classifying a video's direction of time in Kinetics-400, thus demonstrating that Kinetics-400 has subsets of videos that look realistic even when reversed. These findings that additional distinctiveness combined with invariances improve video representation learning are noteworthy, as they contradict results from the image self-supervised learning domain, where learning pretext-invariance can lead to more transferable representations [52]. Even when compared to previous self-supervised learning approaches for video-data, such as predicting the arrow of time [76], our method yields new insights by showing that a unique combination of distinctivenesses and invariances performs best, at least on the training sets considered. Combining these points, the strong performance of GDT is founded in its ability to leverage highly informative, yet "free", signal that we have from construction of the transformations.

4.3. Qualitative analysis

Here, we study what effect the different transformations we let our model be invariant and distinctive to have on our learned representations. For this, we compare against the SimCLR baseline of Tab.1(a) and compare the average standard deviation of the normalized features for 10 time-shifted clips per video for 3000 randomly selected Kinetics-400 validation set.

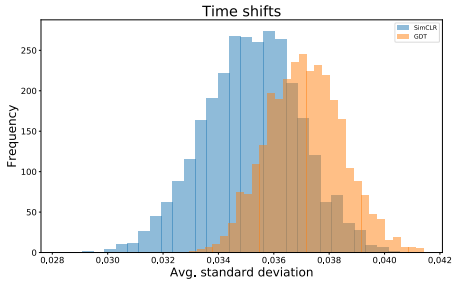

Fig. 3: Learning distinctiveness to time-shifts: our GDT model from Tab.1(j) is able to differentiate features from the same video at different times better than a simple SimCLR variant (Tab.1(a)).

Table 3: **Video retrieval and Few Shot Learning.** Retrieval accuracy in (%) via nearest neighbors at various levels of neighborhood sizes and few shot learning accuracy (%) via a k-nearest neighbor on frozen representations.

		HMDB		UCF	
		1	5	1	5
kNN	Random	3.0	3.5	2.3	4.6
	3DRot [38]	–	–	15.0	31.5
	GDT (ours)	**14.3**	**15.4**	**26.7**	**44.6**
Retrieval	SP-Net [9]	-	-	13.0	28.1
	VCP [14]	7.6	24.4	18.6	33.6
	M-DPC [29]	7.7	25.7	20.2	40.4
	VSP [14]	10.3	26.6	24.6	41.9
	CoCLR [30]	23.2	43.2	53.3	69.4
	SeLaVi [5]	24.8	47.6	52.0	68.6
	GDT (ours)	**26.1**	**51.7**	**62.8**	**79.0**

4.4. Comparison to the state of the art

Given one of our best learning setups from Sec. 4.1 (row (l)), we train for longer and compare our feature representations to the state of the art on standard downstream benchmarks.

4.4.1 Downstream benchmarks

For **few-shot classification**, as shown in table 3, we significantly beat the 3D-Rotnet [38] baseline on UCF-101 by more than 10% on average for each shot with our Kinetics-400 pretrained model.

Table 4: **State-of-the-art on video action recognition with full-finetuning.** Self- and fully-supervisedly trained methods on UCF-101 and HMDB-51 benchmarks.

Method	Data	Top-1 Acc%	
		HMDB	UCF
Supervised [79]	K-400+IN	75.9	96.8
Supervised [2]	K-400	65.1	94.2
AoT [76]	K-400	-	79.4
MultiSensory [59]	K-400	-	82.1
SeLaVi [5]	K-400	47.1	84.2
PEMT [44]	K-400	–	85.2
XDC [2]	K-400	52.6	86.8
AV Sync+RotNet [78]	K-400	54.6	87.0
CoCLR [30]	K-400	54.6	87.9
SeCo [82]	K-400	55.6	88.3
AVTS [41]	K-400	56.9	85.8
CPD [46]	K-400	57.7	88.7
AVID [54]	K-400	60.8	87.5
CM-ACC [47]	K-400	61.8	90.2
GLCM [48]	K-400	61.9	91.2
GDT (ours)	K-400	**62.3**	**90.9**
MIL-NCE [49]	HT100M	61.0	91.3
GDT (ours)	HT100M	**67.4**	**94.1**
XDC [2]	IG65M	68.9	**95.5**
GDT (ours)	IG65M	**72.8**	95.2

For **video retrieval**, we report recall at 1 and 5 retrieved samples for split-1 of the HMDB-51 and UCF-101 datasets in table 3.Using our model trained on Kinetics-400, GDT significantly beats all other self-supervised methods. In particular, we outperform CoCLR [30], a recent state-of-the-art self-supervised method, that uses optical flow as another view to mine hard positive to improve instance discrimination learning for video representations. Moreover, we surpass SeLaVi, an audio-visual clustering and representation learning method, by 2% and 10% on average on recall at 1 and 5 for HMDB-51 and UCF-101.

For **video action recognition**, we finetune our GDT pretrained network for UCF-101 and HMDB-51 video classification, and compare against state-of-the-art self-supervised methods in table 4. When pretrained on the Kinetics datasets, we find that our GDT pretrained model achieves very good results, outperforming all recent methods. In particular, we outperform audio-visual pretraining methods, AVTS [41], SeLaVi [5] and XDC [2], by large margins using the same architecture (R(2+1)D-18) and dataset (Kinetics-400), showing the effectiveness of our GDT pre-training approach. We also surpass AVID [54], the state-of-the-art audio-visual representation learning method, by 1.5% on HMDB-51 and 3.8% on UCF-101. AVID uses a variant of the pre-training scheme of our baseline approach that extends noise contrastive learning to the audio-visual domain as in Table 1, row (e). However, while AVID simply encodes sample distinctiveness and invariance to modality in its visual representations, we are able to encode invariances and distinctiveness to additional transformations, which significantly improves our performance. Our approach is also more sample efficient, as we are able to achieve our results with 300 less epochs of training. Finally, when pretrained on HT100M, we achieve strong gains of $+6.4\%$ on HMDB-51 and $+2.8\%$ on UCF-101 compared to the state-of-the-art video text method, MIL-NCE [49]. Similar to AVID, MIL-NCE uses a variant of the baseline cross-modal contrastive framework to learn representations, while we are able to improve upon this baseline by learning invariance and distinctiveness to additional transformations such as time reversal and time shift. Moreover, with HT100M pre-training, we outperform the Kinetics supervised baseline using the same architecture when finetuned on HMDB-51 (67.4 vs 65.1) and are on par for UCF-101 (94.1 vs 94.2). We further show the scalability and flexibility of our GDT framework by pretraining on the IG65M dataset [21]. With this, our visual feature representation sets a new state of the art among all self-supervised methods, particularly by a margin of $>4\%$ on the HMDB-51 dataset. On UCF-101, we set similar state-of-the-art performance with XDC. Along with XDC, we beat the Kinetics supervised pretraining baseline using the same architecture and finetuning protocol.

5. Conclusion

We introduced the framework of Generalized Data Transformations (GDTs), which allows one to capture, in a single noise-contrastive objective, cues used in several prior contrastive and non-contrastive learning formulations, as well as easily incorporate new ones. The framework shows how new meaningful compositions of transformations can be obtained, encoding valuable invariance and distinctiveness that we want our representations to learn. Following this methodology, we achieved state-of-the-art results for self-supervised pretraining on standard downstream video action recognition benchmarks, even surpassing supervised pretraining. Overall, our method significantly increases the expressiveness of contrastive learning for self-supervision, making it a flexible tool for many multi-modal settings, where a large pool of transformations exist and an optimal combination is sought.

References

[1] Jean-Baptiste Alayrac, Adrià Recasens, Rosalia Schneider, Relja Arandjelović, Jason Ramapuram, Jeffrey De Fauw, Lucas Smaira, Sander Dieleman, and Andrew Zisserman. Self-supervised multimodal versatile networks. In *NeurIPS*, 2020.

[2] Humam Alwassel, Bruno Korbar, Dhruv Mahajan, Lorenzo Torresani, Bernard Ghanem, and Du Tran. Self-supervised learning by cross-modal audio-video clustering. In *NeurIPS*, 2020.

[3] Ankesh Anand, Evan Racah, Sherjil Ozair, Yoshua Bengio, Marc-Alexandre Côté, and R Devon Hjelm. Unsupervised state representation learning in atari. 2019.

[4] Relja Arandjelovic and Andrew Zisserman. Look, listen and learn. In *ICCV*, 2017.

[5] Yuki M Asano, Mandela Patrick, Christian Rupprecht, and Andrea Vedaldi. Labelling unlabelled videos from scratch with multi-modal self-supervision. *NeurIPS*, 2020.

[6] Yuki M Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *ICLR*, 2020.

[7] Yusuf Aytar, Carl Vondrick, and Antonio Torralba. Soundnet: Learning sound representations from unlabeled video. In *NeurIPS*, 2016.

[8] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. In *NeurIPS*, 2019.

[9] 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 *CVPR*, 2020.

[10] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018.

[11] Mathilde Caron, Piotr Bojanowski, Julien Mairal, and Armand Joulin. Unsupervised pre-training of image features on non-curated data. In *ICCV*, 2019.

[12] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *NeurIPS*, 2020.

[13] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

[14] 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.

[15] Joon Son Chung and Andrew Zisserman. Out of time: automated lip sync in the wild. In *Workshop on Multi-view Lip-reading, ACCV*, 2016.

[16] Virginia R. de Sa. Learning classification with unlabeled data. In *NeurIPS*, 1994.

[17] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *ICCV*, 2015.

[18] Dave Epstein, Boyuan Chen, and Carl Vondrick. Oops! predicting unintentional action in video. In *CVPR*, 2020.

[19] Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould. Self-supervised video representation learning with odd-one-out networks. In *Proc. CVPR*, 2017.

[20] Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould. Self-supervised video representation learning with odd-one-out networks. In *CVPR*, 2017.

[21] Deepti Ghadiyaram, Du Tran, and Dhruv Mahajan. Large-scale weakly-supervised pre-training for video action recognition. In *CVPR*, 2019.

[22] Spyros Gidaris, Andrei Bursuc, Nikos Komodakis, Patrick Pérez, and Matthieu Cord. Learning representations by predicting bags of visual words. In *CVPR*, 2020.

[23] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *ICLR*, 2018.

[24] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch SGD: training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.

[25] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. In *NeurIPS*, 2020.

[26] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *AISTATS*, 2010.

[27] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, 2006.

[28] Tengda Han, Weidi Xie, and Andrew Zisserman. Video representation learning by dense predictive coding. In *ICCV Workshops*, 2019.

[29] Tengda Han, Weidi Xie, and Andrew Zisserman. Memory-augmented dense predictive coding for video representation learning. In *ECCV*, 2020.

[30] Tengda Han, Weidi Xie, and Andrew Zisserman. Self-supervised co-training for video representation learning. In *NeurIPS*, 2020.

[31] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.

[32] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[33] 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.

[34] R Devon Hjelm and Philip Bachman. Representation learning with video deep infomax, 2020.

[35] 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. In *ICLR*, 2019.
[36] Simon Jenni, Givi Meishvili, and Paolo Favaro. Video representation learning by recognizing temporal transformations. In *ECCV*, 2020.
[37] Xu Ji, João F. Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation, 2018.
[38] Longlong Jing and Yingli Tian. Self-supervised spatiotemporal feature learning by video geometric transformations. *arXiv preprint arXiv:1811.11387*, 2018.
[39] 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.
[40] Dahun Kim, Donghyeon Cho, and In So Kweon. Self-supervised video representation learning with space-time cubic puzzles. In *AAAI*, 2019.
[41] Bruno Korbar, Du Tran, and Lorenzo Torresani. Cooperative learning of audio and video models from self-supervised synchronization. In *NeurIPS*, 2018.
[42] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. HMDB: a large video database for human motion recognition. In *ICCV*, 2011.
[43] Hsin-Ying Lee, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Unsupervised representation learning by sorting sequences. In *ICCV*, 2017.
[44] Sangho Lee, Youngjae Yu, Gunhee Kim, Thomas Breuel, Jan Kautz, and Yale Song. Parameter efficient multimodal transformers for video representation learning. In *ICLR*, 2021.
[45] Junnan Li, Pan Zhou, Caiming Xiong, and Steven Hoi. Prototypical contrastive learning of unsupervised representations. In *ICLR*, 2021.
[46] Tianhao Li and Limin Wang. Learning spatiotemporal features via video and text pair discrimination. *arXiv preprint arXiv:2001.05691*, 2020.
[47] Shuang Ma, Zhaoyang Zeng, Daniel McDuff, and Yale Song. Active contrastive learning of audio-visual video representations. In *ICLR*, 2021.
[48] Shuang Ma, Zhaoyang Zeng, Daniel McDuff, and Yale Song. Contrastive self-supervised learning of global-local audio-visual representations, 2021.
[49] 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 *CVPR*, 2020.
[50] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *ICCV*, 2019.
[51] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.
[52] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *CVPR*, 2020.
[53] Ishan Misra, C Lawrence Zitnick, and Martial Hebert. Shuffle and learn: unsupervised learning using temporal order verification. In *ECCV*, 2016.
[54] Pedro Morgado, Nuno Vasconcelos, and Ishan Misra. Audio-visual instance discrimination with cross-modal agreement. 2020.
[55] Arsha Nagrani, Chen Sun, David Ross, Rahul Sukthankar, Cordelia Schmid, and Andrew Zisserman. Speech2action: Cross-modal supervision for action recognition. In *CVPR*, 2020.
[56] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*, 2016.
[57] Mehdi Noroozi, Hamed Pirsiavash, and Paolo Favaro. Representation learning by learning to count. In *ICCV*, 2017.
[58] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.
[59] Andrew Owens and Alexei A Efros. Audio-visual scene analysis with self-supervised multisensory features. In *ECCV*, 2018.
[60] Andrew Owens, Jiajun Wu, Josh H McDermott, William T Freeman, and Antonio Torralba. Ambient sound provides supervision for visual learning. In *ECCV*, 2016.
[61] Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. Specaugment: A simple data augmentation method for automatic speech recognition. In *INTERSPEECH*, 2019.
[62] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016.
[63] Lyndsey C Pickup, Zheng Pan, Donglai Wei, YiChang Shih, Changshui Zhang, Andrew Zisserman, Bernhard Scholkopf, and William T Freeman. Seeing the arrow of time. In *CVPR*, 2014.
[64] AJ Piergiovanni, Anelia Angelova, and Michael S. Ryoo. Evolving losses for unsupervised video representation learning. In *CVPR*, 2020.
[65] Will Price and Dima Damen. Retro-Actions: Learning 'close' by time-reversing 'open' videos. 2019.
[66] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *NeurIPS*, 2016.
[67] Khurram Soomro, Amir Roshan Zamir, and Mubarak

Shah. UCF101: A dataset of 101 human action classes from videos in the wild. In *CRCV-TR-12-01*, 2012.

[68] Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. Contrastive bidirectional transformer for temporal representation learning. *arXiv preprint arXiv:1906.05743*, 2019.

[69] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *ICCV*, 2019.

[70] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. 2020.

[71] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning. *arXiv preprint arXiv:2005.10243*, 2020.

[72] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, 2018.

[73] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *JMLR*, 9(2579-2605):85, 2008.

[74] 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 *CVPR*, 2019.

[75] Jiangliu Wang, Jianbo Jiao, and Yun-Hui Liu. Self-supervised video representation learning by pace prediction. In *ECCV*, 2020.

[76] Donglai Wei, Joseph J Lim, Andrew Zisserman, and William T Freeman. Learning and using the arrow of time. In *CVPR*, 2018.

[77] Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, 2018.

[78] Fanyi Xiao, Yong Jae Lee, Kristen Grauman, Jitendra Malik, and Christoph Feichtenhofer. Audiovisual slowfast networks for video recognition. *arXiv preprint arXiv:2001.08740*, 2020.

[79] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *ECCV*, 2018.

[80] Dejing Xu, Jun Xiao, Zhou Zhao, Jian Shao, Di Xie, and Yueting Zhuang. Self-supervised spatiotemporal learning via video clip order prediction. In *CVPR*, 2019.

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

[82] Ting Yao, Yiheng Zhang, Zhaofan Qiu, Yingwei Pan, and Tao Mei. Seco: Exploring sequence supervision for unsupervised representation learning. In *AAAI*, 2021.

[83] Richard Zhang, Phillip Isola, and Alexei A. Efros. Colorful image colorization. In *Proc. ECCV*, 2016.

[84] Richard Zhang, Phillip Isola, and Alexei A Efros. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In *CVPR*, 2017.

[85] Hang Zhao, Chuang Gan, Wei-Chiu Ma, and Antonio Torralba. The sound of motions. In *ICCV*, 2019.