Self-Supervised Learning of Video-Induced Visual Invariances

Michael Tschannen Josip Djolonga Marvin Ritter Aravindh Mahendran
Neil Houlsby Sylvain Gelly Mario Lucic

Google Research, Brain Team

Abstract

We propose a general framework for self-supervised learning of transferable visual representations based on Video-Induced Visual Invariances (VIVI). We consider the implicit hierarchy present in the videos and make use of (i) frame-level invariances (e.g. stability to color and contrast perturbations), (ii) shot/clip-level invariances (e.g. robustness to changes in object orientation and lighting conditions), and (iii) video-level invariances (semantic relationships of scenes across shots/clips), to define a holistic self-supervised loss. Training models using different variants of the proposed framework on videos from the YouTube-8M (YT8M) data set, we obtain state-of-the-art self-supervised transfer learning results on the 19 diverse downstream tasks of the Visual Task Adaptation Benchmark (VTAB), using only 1000 labels per task. We then show how to co-train our models jointly with labeled images, outperforming an ImageNet-pretrained ResNet-50 by 0.8 points with 10× fewer labeled images, as well as the previous best supervised model by 3.7 points using the full ImageNet data set.

METHOD	MEAN		NAT.	SPEC.	STR.
Ex-ImageNet	59.5		50.5	**81.4**	56.4
VIVI-Ex(4)	62.5	(+3.0)	55.9	80.9	59.1
VIVI-Ex(4)-Big	**63.3**	(+3.8)	**57.5**	81.0	**59.5**
Semi-Ex-10% [69]	65.3		**70.2**	81.9	52.7
VIVI-Ex(4)-Co(10%)	**67.2**	(+1.9)	63.3	**82.6**	**62.9**
Sup-100% [69]	66.4		73.5	82.5	52.1
Sup-Rot-100% [69]	68.0	(+1.6)	**73.6**	83.1	55.5
VIVI-Ex(4)-Co(100%)	69.4	(+3.0)	69.9	83.3	62.1
VIVI-Ex(4)-Co(100%)-Big	**71.7**	(+5.3)	72.5	**84.3**	**64.7**

Table 1: Mean testing accuracy and per-category mean accuracy for models fine-tuned on the 19 diverse downstream tasks (based on NATural, SPECialized, STRuctured data sets) from the VTAB benchmark [69][1], using only 1000 labels per task. The proposed unsupervised models (VIVI-Ex(4) / VIVI-Ex(4)-Big) trained on raw YT8M videos and variants co-trained with 10%/100% labeled ImageNet data (VIVI-Ex(4)-Co(10%) / VIVI-Ex(4)-Co(100%)), outperform the corresponding unsupervised (Ex-ImageNet), semi-supervised (Semi-Ex-10%) and fully supervised (Sup-100%, Sup-Rot-100%) baselines by a large margin.

1. Introduction

Supervised deep learning necessitates the collection and manual annotation of large amounts of data, which is often expensive, hard to scale, and may require domain expertise (e.g., in the context of medical data). Expensive data annotation hence presents a bottleneck which impedes the application of deep learning methods to diverse, previously under-explored problems. Learning *transferable visual representations*, namely representations obtained by training a model on one task (or collection of tasks) which can then be adapted to multiple unseen downstream tasks using *few samples*, is therefore a key research challenge [69].

An emerging body of work based on *self-supervision* has demonstrated that it is possible to learn such transferable visual representations. The idea is to carefully construct a *pretext* task which does not rely on manual annotation, yet encourages the model to extract useful features from the input. Videos are a promising data modality to design such pretexts tasks for as they capture variations of the instances over time which are not present in images. In addition, there is an abundance of videos available on the Internet covering almost any imaginable domain. As a result, and with the recent emergence of research video data sets [1, 59], videos have been investigated in the context of self-supervision (for example, [40, 64, 63, 28, 65, 73, 41, 51, 42, 3, 2]). We believe that a holistic approach which captures these diverse efforts can be coupled with image-based pretext tasks to further improve the performance of self-supervised models.

In this work we propose a novel, versatile video-based self-supervision framework for learning *image* representations. We divide a video data set into its natural hierarchy

[1] **We use Version 1 of the VTAB benchmark** (arXiv:1910.04867v1); please see Appendix E for Version 2 (arXiv:1910.04867v2) results.

of frames, shots, and videos. The intuition is that the model can leverage (1) the *frames* to learn to be robust to color perturbations or contrast changes, (2) the *shot* information to be robust to rigid and non-rigid transformations of objects in a scene, and that (3) explicitly accounting for the *video-level* context should encourage the model to capture semantic relationships of scenes across shots/clips. In contrast to individual frame, shot, or video-level self-supervision objectives, our holistic approach yields a representation that transfers better to a large set of downstream tasks. As an additional benefit, our approach does not need to pre-compute optical flow or motion segmentation masks, nor does it rely on object tracking.

We train the proposed model on the YouTube-8M (YT8M) data set (without using video-level labels) and show that this approach leads to state-of-the-art self-supervised results on the 19 diverse downstream tasks of the Visual Task Adaptation Benchmark (VTAB) [69]. We then show how to co-train the model jointly with labeled images, outperforming an ImageNet-pretrained ResNet-50 with 10× fewer labeled images. We also investigate the robustness of our co-training models to natural perturbations as induced by the variations across nearby frames in videos [54]. In summary, our contributions are:

- We propose a versatile framework to learn image representations from *non-curated videos in the wild* by learning frame-, shot-, and video-level invariances.
- We train a variety of models on 3.7M videos from the YT8M data set and achieve a 3.8% absolute improvement over image/frame-based baselines across the 19 diverse tasks of the VTAB benchmark [69], which sets new state of the art among unsupervised methods.
- We augment the self-supervised learning (SSL) training framework with a supervised classification loss using data from ImageNet. The resulting models outperform an ImageNet-pretrained network using only 10% labeled ImageNet images (and no additional unlabeled ones), and achieve a new state of the art when co-trained with the full ImageNet data set, outperforming the best previous supervised result by 3.7 points.

2. Related work

Self-supervised learning of image representations SSL is an active topic of research in the computer vision community. Recent methods [67, 25, 4, 45, 24, 60] have advanced the state of the art in terms of learning representations that can linearly separate between the 1000 ImageNet categories [50]. Prior work has explored diverse self-supervision cues such as predicting the spatial-context [12], colorization [71], equivariance to transformations [18, 44]; alongside unsupervised techniques such as clustering [6, 72], generative modelling [14, 32], and exemplar learning [15]. We adopt some of these SSL losses in our framework at the frame-level.

Learning image representations from videos More relevant to our contribution is the body of literature on SSL of image representations from videos. The *temporal context* of frames in video data has been widely exploited. For example, [40, 35, 16, 5, 64] make use of the order in which frames appear in a video. Other forms of temporal context include its combination with spatial context [63], and the use of spatio-temporal co-occurrence statistics [28]. Orthogonal to these efforts, which attempt to be selective of the differences between frames, prior work along the lines of *slow feature analysis* [65, 73] also exploited videos as a means of learning invariant representations. Temporal coherence was exploited in a co-training setting by early work [41] on learning convolutional neural networks (CNNs) for visual object recognition and face recognition. Slow and steady feature analysis [30] attempts to learn representations that exhibit higher order temporal coherence. This object deformation signal can be separated from global camera motion by tracking objects using unsupervised methods. These tracked patches have been used to learn image representations [62]. Tracking may also be replaced by spatio-temporally matched region proposals [17]. Motivated by these works, we explore learning invariances from temporal information in video pixels.

Some of the earliest work making use of temporal consistency used *future frame prediction* [56] as a pretext task. A more challenging version of this task is single frame future synthesis. The ambiguity in single-frame prediction has been side-stepped via time-agnostic prediction [29], motion segmentation [47], cross-pixel matching [38], and by giving the model a motion cue as input [70]. The latter two require distilling the temporal information from video pixels into optical-flow fields.

Optical flow has been treated as a separate modality from the RGB pixels in a *multi-modal* setting [51, 60]. Beyond optical-flow, videos on the web are inherently multi-modal, as they contain audio and subtitles. Multi-modal learning methods that combine vision and audio [42, 9, 46, 3], and vision and text [57] achieve better performance than uni-modal baselines. In a robotics setting, RGB pixels may be considered together with ego-motion [2, 31]. Time-contrastive networks [53] consider two views of the same action to learn invariant representations.

Doersch et al. [13] show that motion-based SSL may be combined with other self-supervision cues namely exemplar, colorization, and spatial-context, to pre-train models that perform better than each of these cues individually. Taking inspiration from their success our framework presents a *synergistic combination* of SSL methods.

Transferable representations Fine-tuning models pre-

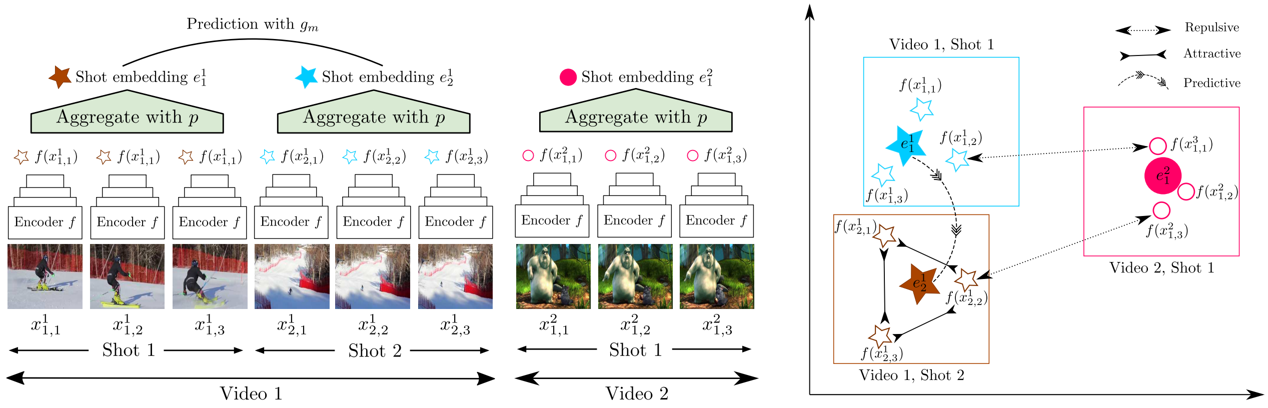

Figure 1: (**left**) Illustration of the frame-, shot-, and video-level encoding pipeline used in this work. Each frame $x_{k,\ell}^i$ is encoded using the frame encoder f. The frame embeddings $f(x_{k,\ell}^i)$ are then aggregated for each shot using a pooling function p to obtain shot embeddings e_k^i. Predictions on the video level are then computed using the prediction functions g_m. (**right**) Intuitively, we want to choose frame/shot- and video-level losses that embed frames from the same shot close to each other and frames from different shots or videos far apart, while encouraging shot embeddings from the same video to be predictive of each other using (simple) prediction functions.[2]

trained on ImageNet labels is a popular strategy for transferring representations to new tasks [26]. Kornblith et al. [34] show that better supervised models tend to transfer better when fine-tuned. Other supervised learning benchmarks focus on performance on multiple data sets, either via transfer learning, meta-learning, or multi-task learning [49, 61]. In the representation learning literature, models are usually evaluated in-domain, typically on ImageNet [70, and references therein]. However, self-supervised models are now performing well on tasks such as surface normal estimation, detection, and navigation [19]. The VTAB benchmark evaluates the transferability of representations beyond object classification in the natural image domain to many domains and task semantics such as counting and localization [69]. Similarly, recent developments in natural language processing (NLP) have lead to representations that transfer effectively to many diverse tasks [11].

3. Learning video-induced visual invariances

We start by giving an overview of the proposed framework in Sec. 3.1, and discuss frame/shot-level and video-level losses in detail in Sec. 3.2 and Sec. 3.3, respectively.

3.1. Overview

We consider a data set $\mathcal{X}$ containing N videos, each composed of multiple shots. For simplicity of exposition we assume that each video consists of K shots, and each shot has L frames. If we denote the ℓ-th frame in the k-th shot of video i by $x_{k,\ell}^i$, we can write the data set as $\mathcal{X} = \{x_{1:K,1:L}^i\}_{i=1}^N$. Our framework consists of a frame-encoder f, a frame embedding pooling function p, and one or multiple shot-level prediction functions g_m (see Fig. 1). The pooling function computes an embedding e_k^i of the k-th shot in video i by feeding each frame through the frame encoder and applying the pooling function,

$$e_k^i = p(f(x_{k,1}^i), \dots, f(x_{k,L}^i)).$$

The pooling function can have different forms, ranging from simple average pooling to attention pooling taking the values of the individual frame embeddings $f(x_{k,\ell}^i)$ into account. Shot-level prediction functions are trained to predict pretext (label-free) targets from shot embeddings.

We define a frame/shot-level loss and a video-level loss to learn invariances at different levels of abstraction. More specifically, the frame/shot-level loss takes the form

$$\mathcal{L}_{\mathrm{S}} = \sum_{i,k} L_{\mathrm{S}}(f(x_{k,1}^i), \dots, f(x_{k,L}^i); y_{k,1}^i, \dots, y_{k,L}^i),$$

where $y_{k,\ell}^i$ are shot-level pretext labels and L_{S} is a shot-level loss that can be instantiated as only acting on the frame level in the sense of L_{S} decomposing into a sum over the frames $\ell = 1, \dots, L$ (see Sec. 3.2 for concrete instantiations of the losses). The video-level loss is given by

$$\mathcal{L}_{\mathrm{V}} = \sum_{i,m} L_{\mathrm{V}}(g_m(e_1^i, \dots, e_K^i)); y_m^i), \tag{1}$$

where the y_m^i are video-level pretext labels and L_{V} is a video-level loss (see Sec. 3.3 for concrete losses). The total loss is then given by $\mathcal{L}_{\mathrm{SSL}} = \mathcal{L}_{\mathrm{S}} + \lambda \mathcal{L}_{\mathrm{V}}$, where $\lambda > 0$ balances the shot level and video level losses. $\mathcal{L}_{\mathrm{SSL}}$ is minimized jointly w.r.t. the parameters of f, p, and g_m.

[2] Video credit: https://vimeo.com/362621732 and https://en.wikipedia.org/wiki/Big_Buck_Bunny.

Co-training with labeled images We also consider the case where one has access to a limited number of labeled images in addition to the video data. Combining image-based SSL losses with a supervised loss applied to a subset of the images was studied previously by [68]. They found that this approach leads to a state-of-the-art semi-supervised models, and improves the performance of supervised models when all images are labeled. Here, we consider the related setup where the SSL loss is computed on video data, and the supervised loss is based on image data from a different data set. Specifically, we additionally apply f followed by a linear classifier to mini-batches of labeled images and compute the cross-entropy loss $\mathcal{L}_{\text{SUP}}$ between the predictions and the image labels. The total loss is then computed as $\mathcal{L}_{\text{SSL}} + \gamma\mathcal{L}_{\text{SUP}}$, where $\gamma > 0$ balances the contributions of the self-supervised and supervised loss terms.

Relation to prior work We are not aware of prior work using the natural hierarchy of frame, shot, and video-level invariances in videos for self-supervised image representation learning. Further, our approach is geared towards reducing the need for curated datasets and expensive labeling procedures. In contrast, many existing methods for learning image representations from video data often rely on short curated videos consisting of single clips, or even the treat training set as a bag of frames [7, 13].

3.2. Learning shot-level invariances

To define the frame/shot-level loss $\mathcal{L}_{\text{S}}$, we propose to build on any SSL loss designed for images, such as classifying exemplars [15], solving jigsaw puzzles of image patches [43], or rotation prediction [18]. For learning shot-induced invariances, one can take two approaches:

(i) apply the image-based SSL loss independently to each frame so that the shot-induced invariances are learned implicitly through the combination of pooling function and video-level prediction task, or

(ii) explicitly ensure that the embeddings of the frames from the same shot are similar by adding a triplet or a contrastive loss to the image-based SSL loss.

In this work, in the spirit of approach (i) we consider SSL by rotation prediction [18] without additional explicit shot-level loss. To explore approach (ii) we rely on a variant of exemplar SSL [15], where each image is associated with a different class, and a feature extractor is trained to classify each image into its own class after heavily augmenting it (random cropping, rotation, contrast, and color shifts). Following [12, 33], to scale this approach to hundreds of millions of images (frames), we employ a triplet loss [52] encouraging augmentations of the same image to be close and augmentations of different images to be far apart. To learn invariances from different frames of the same shot, rather than picking a random frame from the shot and applying M random augmentations to it, we pick M *consecutive frames* from the same shot and augment each frame once. As a result, our feature extractor learns both the invariances induced by temporal variation in video as well as those induced by the data augmentation.

3.3. Learning video-level invariances

In contrast to action recognition networks, which learn video representations that have to be discriminative w.r.t. changes between frames, our framework targets learning representations that are invariant to such changes. Nevertheless, discriminative tasks useful for learning representations for action recognition, such as predicting whether a sequence of frames is played forward or backward [64], verifying whether the frames are ordered or shuffled [40], or predicting features corresponding to future frames [21], can be useful to learn abstract transferable representations when applied to sensibly chosen *groups of aggregated frames*. Following this intuition, our framework allows to apply any of these tasks to shot embeddings, rather than individual frame embeddings. Despite being discriminative at the video level, these tasks encourage the representation to be invariant to all except those cues that are necessary for the pretext task; and hence indirectly induce invariances. For example, determining whether a sequence of shot embeddings is played forward or backward requires understanding of the high-level semantics of the scene and objects in each shot. Similarly, predicting future shot embeddings from the past ones encourages learning an abstract summary of each shot. In this work we will explore these two approaches.

For shot order prediction, we randomly reverse the order of the shot embeddings and train a prediction function g to predict the shot order from concatenated shot embeddings, i.e., L_{V} in (1) is the cross-entropy loss and y^i_m is 1 if the sequence of shot embeddings is reversed and 0 otherwise. To train g to predict future shot embeddings, we rely on noise-contrastive estimation [20]. Specifically, we use the embeddings of the shots $e^i_1, \ldots, e^i_k$ to obtain a prediction $\hat{e}^i_{k+m}$ of the embedding e^i_{k+m} of the shot m steps in the future. Then, $\mathcal{L}_{\text{V}}$ should quantify the quality of the prediction, which we accomplish using the InfoNCE loss [45]

$$\mathcal{L}_{\text{NCE}} = -\frac{1}{N}\sum_i \log \frac{e^{g(\hat{e}^i_{k+m}, e^i_{k+m})}}{\frac{1}{N}\sum_j e^{g(\hat{e}^i_{k+m}, e^j_{k+m})}}, \qquad (2)$$

where g is trained to assign high scores to pairs of shot embeddings from the same video, and low values to embeddings computed from different videos.[3] Note that the terms in (2) can, up to an additive constant, be seen as the cross-entropy loss of an N-class classification problem where the correct label is i, so that we could reformulate the loss in the form (1) using class labels y^i.

[3] In practice, we use all shot embeddings from the other videos, not only those at time step $k + m$, which is known to improve performance [45].

4. Experimental setup

Our experiments encompass two training phases, which we refer to as *upstream* and *downstream*. First, in the upstream phase, we train our models on video (and image) data using the methods proposed in the previous section. Then, we fine-tune those trained models on a set of downstream problems in the second phase. We focus on the challenging scenario in which the downstream data is limited, and use only 1000 examples for each downstream task [69].

Upstream training We train on the videos in the YT8M data set [1], which consists of millions of YouTube video IDs with over 3800 visual entities. We downloaded approximately 4.7M of these videos sampled at 1 Hz and split them into a training set of 3.7M and a testing set of 1M videos. We further split the videos into shots using a simple strategy based on color histograms, similarly to [39] (see Table 5 in the supplementary material for data set statistics). No other pre-processing or filtering is performed as we target learning from real-world videos in the wild. We also present results of several baseline approaches applied to a data set obtained by selecting a single random frame from each video, which we refer to as YT8M frames.

Furthermore, in the co-training experiments we also use (a class-balanced fraction of) the ImageNet (ILSVRC-2012) training set [10].

Downstream evaluation To evaluate the learned representations, we use the data sets and follow the protocol of the **VTAB Version 1** (arXiv:1910.04867v1) [69].[4] This protocol consists of 19 tasks categorized into three groups as follows (details and references are in the appendix).

- *Natural* — Six classical image classification problems on natural images (data sets: Caltech101, CIFAR-100, DTD, Flowers102, Pets, Sun397 and SVHN).
- *Specialized* — Image classification on data captured using specialist equipment, from the remote-sensing (data sets: Resisc45, EuroSAT) and medical (data sets: Patch Camelyon, Diabetic Retinopathy) domains.
- *Structured* — Eight tasks to predict properties of the objects appearing in an image (how many there are, their relative position and distance), on both rendered (Clevr, dSprites, SmallNORB, DMLab) and real (KITTI) data.

For each of these 19 tasks and each model that we propose, we launch a sweep over 4 hyper-parameters (learning rates and schedules, as in the lightweight mode of [69]). Then, we choose the models that had the best validation accuracy when averaged over these 19 tasks. These best-performing models are then re-trained for each data set on 1000 random points from the union of the train and validation set and evaluated on the testing set. To account for the randomness coming from the initialization of the fresh classification head and the order in which the data appears, we repeated this evaluation scheme three times and report the median test set accuracy (following [69]).

Architectures and training details The frame encoder f is modeled using the ResNet-50 v2 [23] architecture with BatchNorm [27]. We also investigated the effect of model capacity by widening the network by a factor of three. To avoid mismatch in batch statistics between the two data sources, in the co-training experiments we replace BatchNorm with GroupNorm [66] and also standardize [48] the weights of the convolutions. We construct mini-batches by sampling either 2 or 4 consecutive shots from each video (dropping those videos with fewer shots), and randomly select 8 consecutive frames for exemplar-based shot-level SSL and 4 consecutive frames rotation-based frame-level SSL. For the $\mathcal{L}_{\text{NCE}}$ loss, when we sample 2 shots, we predict the embedding of one from the embedding of the other one using a multilayer perceptron (MLP), i.e., the function g in (2) has the form $g(e, e') = \phi_1(e)^\top \phi_2(e')$, where ϕ_1, ϕ_2 are MLPs with a single hidden layer with 256 units. In the experiments with 4 shots, we use a Long Short-Term Memory (LSTM) prediction function with 256 hidden units to predict every shot embedding from the previous ones. We use temporal order prediction only together with exemplar-based SSL and for data with 2 shots per video, relying on a single-hidden-layer MLP with 512 hidden units as prediction function. Throughout, we rely on (parameter-free) average pooling for p. For both frame and shot-level SSL approaches we use the augmentation mechanism from [58]. For models co-trained with a supervised loss based on a fraction of ImageNet we additionally use the same HSV-space color randomization as [68].

We also perform experiments where we replace the augmentation mechanism from [58] with AutoAugment (AA), which is an augmentation policy learned using a reinforcement learning algorithm from the full ImageNet data set. While this can cause *label leakage* when applied to unsupervised methods, we investigate it to understand how these automatically learned invariances compare to those induced by shot-based augmentation which are label-free.

In all cases we choose the batch size such that the product of the number of videos and the number of shots is 2048, i.e., $NK = 2048$. We train all unsupervised models for 120k iterations, using stochastic gradient descent (SGD) with a learning rate of 0.8 and momentum 0.9, multiplying the learning rate by 0.1 after 90k and 110k iterations. The co-trained models are trained for 100k iterations, and the schedule as well as the batch size is chosen depending on the amount of labeled data used. For the weight λ (and γ for co-trained models) we sweep over at most four different

[4] Please see Appendix E for Version 2 (arXiv:1910.04867v2) results; the relative improvements of our methods over baselines and the conclusions are similar.

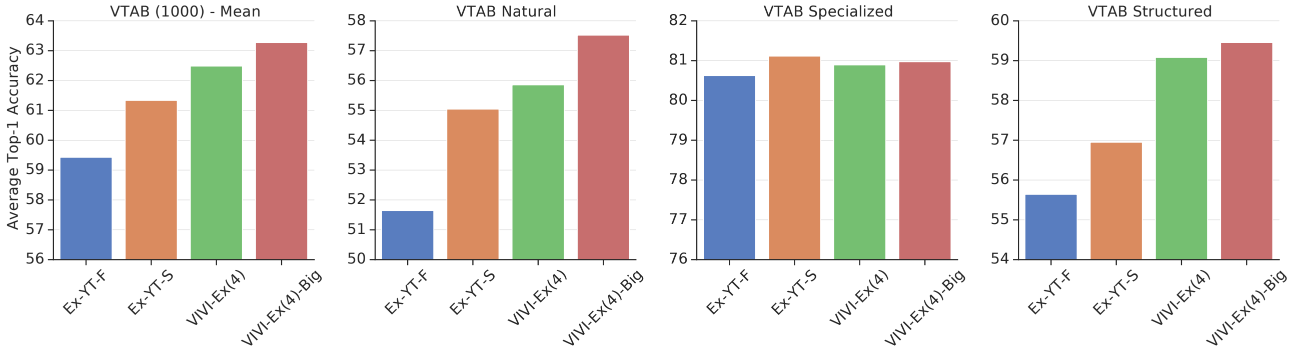

Figure 2: VTAB 1000 example mean score and per-category mean score of exemplar SSL from YT8M frames (Ex-YT-F), with additional shot-level self-supervision (Ex-YT-S), the proposed method with InfoNCE video-level prediction across 4 shots (VIVI-Ex(4)) and additionally 3×wider architecture (VIVI-Ex(4)-Big). Both shot and video-level losses improve the overall score, with the gains coming mostly from higher mean accuracy on the natural and structured subsets.

values. A complete description of all hyper-parameters and architectures can be found in the appendix.

Baselines We train a rotation and exemplar baseline model on ImageNet and a data set obtained by sampling one frame from each video in our training set (YT8M frames). We use the same training protocol as [33] for the respective methods except that we increase the batch size to 2048 and stretch the schedule to 120k iterations to be comparable to our methods. Furthermore, for the exemplar-based model we ablate the video-level prediction task, which amounts to treating the shots independently and only using the frames from the same shot as exemplars. In addition, we consider 3 baselines from [69]: A standard ResNet-50 v2 pretrained on ImageNet (achieving top-1/top-5 accuracy of 75.5%/92.6% on the ImageNet validation set), the exemplar model trained on ImageNet with 10% class-balanced labeled data from [68] (Semi-Ex-10%), which achieves state-of-the-art semi-supervised accuracy on ImageNet, and the rotation model trained on ImageNet with all labels [68] (Sup-Rot-100%).

We further compare against three prior works that learn image representations from video data: The motion segmentation (MS) [47] and the multi-task SSL (MT-SSL) models from [13], and the transitive invariance (TI) model from [63]. MS learns representations based on a foreground-background segmentation pretext task. The segmentation maps are derived using an off-the-shelf offline video segmentation algorithm. MT-SSL combines MS and three other self supervision objectives to train a multi-task network. Its representation derives from a combination of colorization, spatial context, and motion segmentation cues. The MS and MT-SSL models fine-tuned in this evaluation have a ResNet-101 [22] architecture up to the third block. TI builds a graph combining intra-instance and inter-instance edges and exploits transitivity to learn invariant representations. The intra-instance edges are obtained by tracking patches in videos. We fine-tune their publicly available pre-trained VGG-16 [55] checkpoint. We refer the reader to the supplementary material for implementation details regarding the evaluation of these baselines.

5. Results

In this section we focus on the low sample-size regime, i.e., when each downstream data set consists of 1000 samples, and discuss the performance on the full data sets in the supplementary material (Table 4). In brief, the ranking of the methods according to the VTAB mean score using all examples is similar to the ranking according to the VTAB 1000 example mean score. Further, here we only present the best configuration (w.r.t. the number of shots K and choice of prediction function) for each of our VIVI learning approaches, and defer the results for other configurations to the supplementary material (Table 4). We also present an evaluation of the proposed methods on object detection in the supplementary material.

5.1. Self-supervised learning

Exemplar Fig. 2 shows the results for our models and the exemplar-based baselines. The baseline trained on YT8M frames only (Ex-YT-F), without leveraging any temporal information, achieves a mean VTAB 1000 example score of 59.4%. Exploiting the temporal variations within shots to create exemplars (Ex-YT-S) increases that score by about 1.9 points. Further, adding the video-level prediction loss on top adds another 1.2 points. It hence appears that leveraging both shot- and video-level invariances using our approach leads to significant gains over just using frames. In addition, increasing the model capacity (using a 3×wider model) leads to another increase by 0.8 points. Note that this model is only 2.0 points behind the semi-supervised model from [68] (Semi-Ex-10%) which uses 128k labeled images from ImageNet for training (cf. Table 1). The gains mostly come from improvements on the natural and struc-

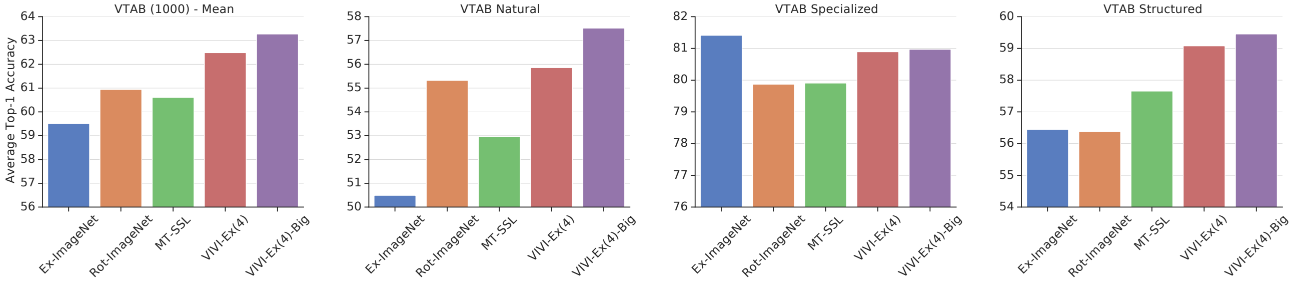

Figure 3: Comparison of the VTAB 1000 example mean score of the proposed method with exemplar frame/shot-level SSL and InfoNCE video-level prediction across 4 shots (VIVI-Ex(4), and with a 3× wider architecture (VIVI-Ex(4)-Big)), with ImageNet-based exemplar (Ex-ImageNet) and rotation (Rot-ImageNet) baselines, as well as the multi-task SSL model from [13]. Our models outperform all baselines on average, and in particular on the structured data sets.

tured data sets, whereas video-level losses do not notably improve the score on the specialized data sets (see Fig. 2). We observed the largest gains when using $\mathcal{L}_{\text{NCE}}$ with $K = 4$ shots and more modest improvements for $\mathcal{L}_{\text{NCE}}$ and temporal order prediction with $K = 2$ shots.

Rotation Similarly to the exemplar experiments, we observe gains of 2.0 points in the mean VTAB 1000 example score over the frame-based baseline (Rot-YT-F) when using a video-level prediction task (VIVI-Rot in Table 2). The gains are smaller for $K = 2$ than for $K = 4$ shots when combined with $\mathcal{L}_{\text{NCE}}$, and temporal order prediction was not effective when combined with rotation prediction as frame-level loss for both $K \in \{2, 4\}$. We emphasize that the frame encoder trained via rotation SSL on YT8M frames performs considerably worse than the same model trained on ImageNet. This is not surprising as ImageNet images are carefully cropped and the data has a balanced class distribution. By contrast, frames sampled from YT8M are less balanced in terms of content and arguably provide many shortcuts for the rotation task such as black borders, overlaid logos, frames with text, or lack of orientation cues.

Effect of AutoAugment (AA) Table 2 shows the effect of using AA [8] instead of the augmentation mechanism from [58]. The effect is strongest on the frame-based baselines, increasing the VTAB 1000-example score by at least 2, and weakest on models involving shot- and video-level losses, where the increase is between 0.5 and 1.5 points. Hence, the invariances induced by AA are, to some degree, complementary to the proposed shot- and video-level losses. However, note that AA is trained on labeled ImageNet images, which might introduce label leakage. Hence, methods relying on AA should not be considered fully unsupervised.

	Exemplar				Rotation	
	YT-F	YT-S	VIVI(4)	VIVI(4)-Big	YT-F	VIVI
w/o AA	59.4	61.3	62.5	63.3	56.9	58.9
AA	61.8	62.8	63.0	64.4	58.9	59.9

Table 2: Effect of replacing the data augmentation mechanism from [58] with AA. Video-induced invariances learned by our method are complementary to AA in the sense that applying AA to different variants of our method consistently leads to improvements.

Comparison with related work Fig. 3 presents a summary of the comparison with baselines. We omit MS and TI as they obtain a VTAB 1000 example mean score comparable to relative patch location prediction [12] and jigsaw [43] SSL trained on ImageNet. These two methods have a significantly lower VTAB 1000 example score than the MT-SSL model, as well as rotation and exemplar SSL. Our VIVI models clearly outperform both the ImageNet baseline and the MT-SSL model. The score obtained by MT-SSL is comparable to that obtained by rotation-based SSL trained on ImageNet, which in turn scores 1.4 points higher than exemplar-based SSL. Both our models and MT-SSL significantly outperform rotation and exemplar-based SSL on the structured data sets, whereas the ImageNet-based exemplar baseline obtains the highest mean score on the specialized data sets.

5.2. Co-training with ImageNet

In Table 1 we compare the scores obtained by our exemplar-based co-training models with the baselines from [69]. Our model with frame/shot-level and video-level losses and a wider architecture (VIVI-Ex(4)-Big) reduces the gap between exemplar trained on ImageNet and the strong Semi-Ex-10% semi-supervised baseline model by more than a factor of 2. Moreover, our model co-trained with 10% labeled ImageNet examples (class-balanced, no additional unlabeled ImageNet examples are used) outperforms both the Semi-Ex-10% baseline and the ImageNet pre-trained ResNet-50 on the VTAB 1000 examples mean score. Using the entire labeled ImageNet training set for co-training yields an increase of 2.1 points. Finally, scaling up the architecture and applying AA to pre-process the ImageNet data adds 2.3 points, leading to a clear new state

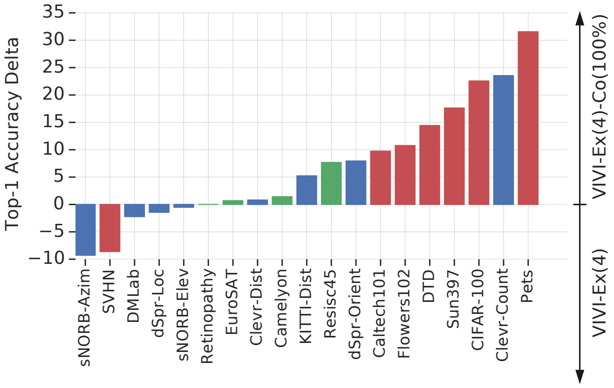

Figure 4: Per-data set comparison of our exemplar-based unsupervised model (VIVI-Ex(4)) and its counterpart co-trained with the full ImageNet data set (VIVI-Ex(4)-Co(100%)). The accuracy on most of the natural (red) and specialized (green) data sets improves, with the largest improvements observed on the latter, while the accuracy decreases for about half of the structured data sets (blue).

Model Type	Accuracy Original	Accuracy Perturbed	Δ
ImageNet	68.0 [65.2, 70.7]	49.9 [46.9, 52.9]	18.1
VIVI-Ex(4)-Co(100%)	62.2 [59.3, 65.1]	46.3 [43.3, 49.2]	15.9

Table 3: ImageNet-Vid-Robust evaluation: We evaluate our VIVI-Ex(4)-Co(100%) model (co-trained using all labeled images available in the ImageNet training set), on the ImageNet-Vid-Robust benchmark [54]. *Accuracy original* is the top-1 accuracy measured on "anchor" frames. *Accuracy perturbed* is the PM-10 accuracy from the benchmark. It is the worst case accuracy defined over neighbouring 20 frames [54] around each "anchor" frame. Δ is the absolute difference between these two. On this benchmark, lower difference is better. The small text in gray corresponds to the Clopper-Pearson confidence interval.

of the art on the VTAB benchmark. The largest gains from using (a subset of) ImageNet can generally be observed on the natural data sets, whereas the gains on the specialized and structured data sets are significantly lower. This result is not surprising given that many data sets in the natural category are semantically similar to ImageNet. Fig. 4 shows the per-data set increase/decrease in the VTAB 1000 example score when adding a classification loss computed on the entire ImageNet data set to VIVI-Ex(4).

Robustness to video perturbations Our co-trained models are trained to both recognize 1000 ImageNet categories and be invariant to deformations found in video data. We therefore expect model predictions to be stable across neighbouring frames in a video. To measure if this is indeed the case, we evaluate our VIVI-Ex(4)-Co(100%) model on the *ImageNet-Vid-Robust* [54] benchmark. This benchmark measures the drop in accuracy under a stricter definition of the 0-1 loss using videos from the ImageNet-Vid data set [50]. Given a set of frames, the prediction on an "anchor" frame is considered correct only if *all* neighboring frames are predicted correctly. Intuitively, the drop in performance going from standard top-1 accuracy on anchor frames to this stricter loss function is indicative of a lack in model robustness. The lower the drop the more robust the model. In Table 3 we observe that our co-trained model is slightly more robust than its purely supervised counterpart, although the results are still within error bars. This is similar to the difference in performance drop observed for fine-tuning on ImageNet-Vid as reported in the benchmark paper itself [54, Table 1]. These initial results suggest that our co-training approach leads to a similar effect as fine-tuning, despite the domain shift between YT8M and ImageNet-Vid. It seems that robustness to natural perturbations in videos is extremely challenging and worth investigating in the future.

6. Conclusion

We propose and evaluate a versatile framework for learning transferable, data-efficient image representations by exploiting video-induced visual invariances at different levels of granularity. The framework can be instantiated with any image-based SSL loss at the frame/shot-level and arbitrary sequence prediction proxy tasks at the video-level. Our experiments reveal that purely self-supervised models benefit greatly from exploiting video-induced invariances, outperforming the SSL baselines trained on ImageNet by a large margin, in particular on problems that require predicting the structural properties of the data. Moreover, when augmenting the proposed framework with a supervised classification loss, the resulting models outperform a standard ImageNet-pretrained model using 10× fewer labeled examples, and set a new state of the art on the VTAB benchmark when co-trained with the full ImageNet data set.

Future research could target better understanding of how the choice of losses and data sets used for upstream training impacts the performance on different tasks in downstream evaluation. While we found our co-trained models to be somewhat more robust to natural perturbations induced by videos than models trained only on images, further research is needed on learning models that overcome robustness issues related to perturbations induced by videos.

Acknowledgments We would like to thank Xiaohua Zhai for inspiring discussions, in particular on how to learn from video shots, and for contributions to preliminary experiments that led to this paper. Further, we would like to thank Raphael Marinier for help with preparing the YT8M data set. Finally, we are grateful to Lucas Beyer for his implementation of GroupNorm with weight standardization.

References

[1] Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. Youtube-8m: A large-scale video classification benchmark. *arXiv:1609.08675*, 2016. 1, 5

[2] Pulkit Agrawal, Joao Carreira, and Jitendra Malik. Learning to see by moving. In *Proc. ICCV*, 2015. 1, 2

[3] Relja Arandjelovic and Andrew Zisserman. Look, listen and learn. In *Proc. ICCV*, 2017. 1, 2

[4] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. In *NeurIPS*, 2019. 2

[5] Uta Buchler, Biagio Brattoli, and Bjorn Ommer. Improving spatiotemporal self-supervision by deep reinforcement learning. In *Proc. ECCV*, 2018. 2

[6] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. *Proc. ECCV*, 2018. 2

[7] Mathilde Caron, Piotr Bojanowski, Julien Mairal, and Armand Joulin. Unsupervised pre-training of image features on non-curated data. In *Proc. ICCV*, pages 2959–2968, 2019. 4

[8] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation policies from data. In *Proc. CVPR*, 2019. 7, 11, 17

[9] Virginia R de Sa. Learning classification with unlabeled data. In *NeurIPS*, 1994. 2

[10] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proc. CVPR*, 2009. 5

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *NAACL-HLT*, 2018. 3

[12] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *Proc. ICCV*, 2015. 2, 4, 7

[13] Carl Doersch and Andrew Zisserman. Multi-task self-supervised visual learning. In *ICCV*, 2017. 2, 4, 6, 7, 12, 13

[14] Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. In *Proc. ICLR*, 2017. 2

[15] Alexey Dosovitskiy, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with convolutional neural networks. In *NeurIPS*, 2014. 2, 4, 11, 17

[16] Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould. Self-supervised video representation learning with odd-one-out networks. In *Proc. CVPR*, 2017. 2

[17] Ruohan Gao, Dinesh Jayaraman, and Kristen Grauman. Object-centric representation learning from unlabeled videos. In *Proc. ACCV*, 2016. 2

[18] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *Proc. ICLR*, 2018. 2, 4, 11, 12, 17

[19] Priya Goyal, Dhruv Mahajan, Abhinav Gupta, and Ishan Misra. Scaling and benchmarking self-supervised visual representation learning. In *Proc. ICCV*, 2019. 3

[20] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proc. AISTATS*, 2010. 4

[21] Tengda Han, Weidi Xie, and Andrew Zisserman. Video representation learning by dense predictive coding. In *Proc. ICCV Workshops*, 2019. 4

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. CVPR*, 2016. 6

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *Proc. ECCV*, 2016. 5, 12

[24] Olivier J Hénaff, Ali Razavi, Carl Doersch, SM Eslami, and Aaron van den Oord. Data-efficient image recognition with contrastive predictive coding. *arXiv:1905.09272*, 2019. 2

[25] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. *Proc. ICLR*, 2019. 2

[26] Minyoung Huh, Pulkit Agrawal, and Alexei A Efros. What makes imagenet good for transfer learning? *arXiv:1608.08614*, 2016. 3

[27] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *Proc. ICML*, 2015. 5, 12

[28] Phillip Isola, Daniel Zoran, Dilip Krishnan, and Edward H Adelson. Learning visual groups from co-occurrences in space and time. *arXiv:1511.06811*, 2015. 1, 2

[29] Dinesh Jayaraman, Frederik Ebert, Alexei A Efros, and Sergey Levine. Time-agnostic prediction: Predicting predictable video frames. *Proc. ICLR*, 2019. 2

[30] Dinesh Jayaraman and Kristen Grauman. Slow and steady feature analysis: higher order temporal coherence in video. In *Proc. CVPR*, 2016. 2

[31] Dinesh Jayaraman and Kristen Grauman. Learning image representations tied to egomotion from unlabeled video. *IJCV*, 125(1):136–161, Dec 2017. 2

[32] Diederik P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. In *NeurIPS*, 2014. 2

[33] Alexander Kolesnikov, Xiaohua Zhai, and Lucas Beyer. Revisiting self-supervised visual representation learning. In *Proc. CVPR*, 2019. 4, 6, 12

[34] Simon Kornblith, Jonathon Shlens, and Quoc V Le. Do better imagenet models transfer better? *CVPR*, 2019. 3

[35] Hsin-Ying Lee, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Unsupervised representation learning by sorting sequences. In *Proc. ICCV*, 2017. 2

[36] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proc. CVPR*, 2017. 14

[37] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *Proc. ECCV*, 2014. 14

[38] Aravindh Mahendran, James Thewlis, and Andrea Vedaldi. Cross pixel optical-flow similarity for self-supervised learning. In *Proc. ACCV*. Springer, 2018. 2

[39] Jordi Mas and Gabriel Fernandez. Video shot boundary detection based on color histogram. *Notebook Papers TRECVID2003, NIST*, 15, 2003. 5

[40] Ishan Misra, C Lawrence Zitnick, and Martial Hebert. Shuf-

fle and learn: unsupervised learning using temporal order verification. In *Proc. ECCV*, 2016. 1, 2, 4

[41] Hossein Mobahi, Ronan Collobert, and Jason Weston. Deep learning from temporal coherence in video. In *Proc. ICML*, 2009. 1, 2

[42] Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y Ng. Multimodal deep learning. In *Proc. ICML*, 2011. 1, 2

[43] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *Proc. ECCV*, 2016. 4, 7

[44] Mehdi Noroozi, Hamed Pirsiavash, and Paolo Favaro. Representation learning by learning to count. In *Proc. ICCV*, 2017. 2

[45] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv:1807.03748*, 2018. 2, 4

[46] Andrew Owens, Jiajun Wu, Josh H McDermott, William T Freeman, and Antonio Torralba. Ambient sound provides supervision for visual learning. In *Proc. ECCV*, 2016. 2

[47] Deepak Pathak, Ross Girshick, Piotr Dollár, Trevor Darrell, and Bharath Hariharan. Learning features by watching objects move. In *Proc. CVPR*, 2017. 2, 6

[48] Siyuan Qiao, Huiyu Wang, Chenxi Liu, Wei Shen, and Alan Yuille. Weight standardization. *arXiv:1903.10520*, 2019. 5, 12

[49] Sylvestre-Alvise Rebuffi, Hakan Bilen, and Andrea Vedaldi. Learning multiple visual domains with residual adapters. In *NeurIPS*, 2017. 3

[50] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *IJCV*, 2015. 2, 8

[51] Nawid Sayed, Biagio Brattoli, and Björn Ommer. Cross and learn: Cross-modal self-supervision. In *German Conference on Pattern Recognition*, 2018. 1, 2

[52] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proc. CVPR*, 2015. 4, 12

[53] Pierre Sermanet, Corey Lynch, Yevgen Chebotar, Jasmine Hsu, Eric Jang, Stefan Schaal, and Sergey Levine. Time-contrastive networks: Self-supervised learning from video. In *ICRA*, 2018. 2

[54] Vaishaal Shankar, Achal Dave, Rebecca Roelofs, Deva Ramanan, Benjamin Recht, and Ludwig Schmidt. A systematic framework for natural perturbations from videos. *arXiv:1906.02168*, 2019. 2, 8

[55] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*, 2014. 6

[56] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhudinov. Unsupervised learning of video representations using LSTMs. In *Proc. ICML*, 2015. 2

[57] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *Proc. ICCV*, 2019. 2

[58] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proc. CVPR*, 2015. 5, 7, 12

[59] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. YFCC100M: The new data in multimedia research. *Comm. ACM*, 59(2):64–73, 2016. 1

[60] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *arXiv:1906.05849*, 2019. 2

[61] Eleni Triantafillou, Tyler Zhu, Vincent Dumoulin, Pascal Lamblin, Kelvin Xu, Ross Goroshin, Carles Gelada, Kevin Swersky, Pierre-Antoine Manzagol, and Hugo Larochelle. Meta-dataset: A dataset of datasets for learning to learn from few examples. *arXiv:1903.03096*, 2019. 3

[62] Xiaolong Wang and Abhinav Gupta. Unsupervised learning of visual representations using videos. In *Proc. ICCV*, 2015. 2

[63] Xiaolong Wang, Kaiming He, and Abhinav Gupta. Transitive invariance for self-supervised visual representation learning. In *Proc. ICCV*, 2017. 1, 2, 6, 12

[64] Donglai Wei, Joseph J Lim, Andrew Zisserman, and William T Freeman. Learning and using the arrow of time. In *Proc. CVPR*, 2018. 1, 2, 4

[65] Laurenz Wiskott and Terrence J Sejnowski. Slow feature analysis: Unsupervised learning of invariances. *Neural Computation*, 14(4):715–770, 2002. 1, 2

[66] Yuxin Wu and Kaiming He. Group normalization. In *Proc. ECCV*, 2018. 5, 12

[67] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proc. CVPR*, 2018. 2

[68] Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer. S4l: Self-supervised semi-supervised learning. In *Proc. ICCV*, 2019. 4, 5, 6, 12

[69] Xiaohua Zhai, Joan Puigcerver, Alexander Kolesnikov, Pierre Ruyssen, Carlos Riquelme, Mario Lucic, Josip Djolonga, Andre Susano Pinto, Maxim Neumann, Alexey Dosovitskiy, et al. The Visual Task Adaptation Benchmark. *arXiv:1910.04867*, 2019. 1, 2, 3, 5, 6, 7, 17, 18

[70] Xiaohang Zhan, Xingang Pan, Ziwei Liu, Dahua Lin, and Chen Change Loy. Self-supervised learning via conditional motion propagation. In *Proc. CVPR*, 2019. 2, 3

[71] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *Proc. ECCV*, 2016. 2

[72] Chengxu Zhuang, Alex Lin Zhai, and Daniel Yamins. Local aggregation for unsupervised learning of visual embeddings. In *Proc. ICCV*, 2019. 2

[73] Will Y Zou, Andrew Y Ng, and Kai Yu. Unsupervised learning of visual invariance with temporal coherence. In *NIPS 2011 Workshop on Deep Learning and Unsupervised Feature Learning*, 2011. 1, 2