Representation learning from videos in-the-wild: An object-centric approach

Rob Romijnders* Aravindh Mahendran Michael Tschannen† Josip Djolonga
Marvin Ritter Neil Houlsby Mario Lucic

Google Research (Brain Team)

Abstract

We propose a method to learn image representations from uncurated videos. We combine a supervised loss from off-the-shelf object detectors and self-supervised losses which naturally arise from the video-shot-frame-object hierarchy present in each video. We report competitive results on 19 transfer learning tasks of the Visual Task Adaptation Benchmark (VTAB), and on 8 out-of-distribution-generalization tasks, and discuss the benefits and shortcomings of the proposed approach. In particular, it improves over the baseline on all 18/19 few-shot learning tasks and 8/8 out-of-distribution generalization tasks. Finally, we perform several ablation studies and analyze the impact of the pretrained object detector on the performance across this suite of tasks.

1. Introduction

Learning transferable visual representations is a key challenge in computer vision. The aim is to learn a representation function once, such that it may be transferred to a plethora of downstream tasks. In the context of image classification, models trained on large amounts of labeled data excel in transfer learning [42], but there is a growing concern that this approach may not be effective for more challenging downstream tasks [79]. Recent advances, such as contrastive self-supervised learning combined with strong data augmentation, present a promising avenue [7].

We consider the problem of learning image representations from uncurated videos. While these videos are noisy and unlabeled, they contain abundant natural variations of the objects present. Furthermore, videos decompose temporally into a hierarchy of videos, shots, and frames, which can be used to define pretext tasks for self-supervised learning [68]. We extend this hierarchy to its natural continuation, namely, the spatial decomposition of frames into objects. We then use the "video, shot, frame, object" hierarchy to define a more holistic pre-text task. In this setting we are hence given uncurated videos and an off-the-shelf pre-trained object detector, and we propose a method of supplementing the loss function with cues at the object level.

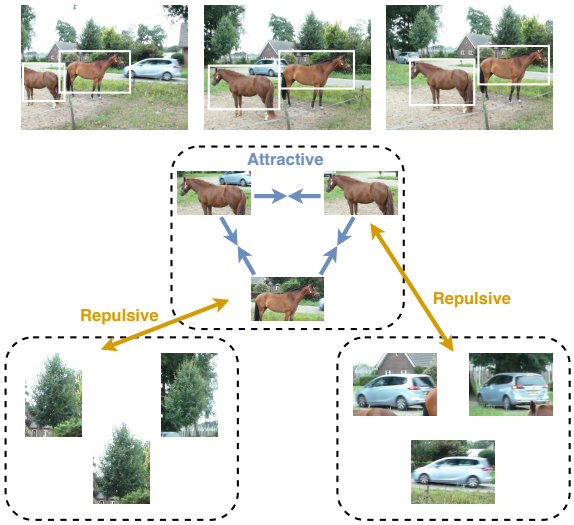

Figure 1: We propose to learn from objects in video. An off-the-shelf pretrained object detector tags each frame with bounding boxes and class labels. A contrastive loss encourages pairs of objects with the same class label (*positives*) to be embedded closer to each other than those from disparate ones (*negatives*). We augment existing work in video representation learning [68] with an object level loss.

Videos, at the frame and shot level, convey global scene structure, and different frames and shots provide a *natural data augmentation* of the scene. This makes videos a good fit for contrastive learning losses that rely on heavy data augmentation for learning scene representations [6]. At the object level, videos also provide rich information about the structure of individual objects. This can be valuable for tasks such as orientation estimation, counting, and object detection. Furthermore, object-centric representations can generalize to

*Work done as part of Google AI residency.
†Work done while at Google Research.

scenes constituted as a novel combination of known objects. Intuitively, each occurrence of the object forms a natural augmentation for objects of that class. Finally, one can make use of the fact that the same object appears in consecutive frames to enable representations which are more robust to perturbations and distributions shifts. Contrastive learning in this setting is illustrated in Figure 1.

Our contributions:

1. We extend the framework from VIVI [68] to include object level cues using an off-the-shelf object detector.

2. We demonstrate improvements using object level cues on recent few-shot transfer learning benchmarks and out-of-distribution generalization benchmarks. In particular, the method improves over the baseline on all 18/19 few-shot learning tasks and 8/8 out-of-distribution generalization tasks.

3. We ablate various aspects of the setup to reveal the source of the observed benefits, including (i) randomizing the object classes and locations, (ii) using only the object labels, (iii) using the detector as a classifier in a semi-supervised setup, (iv) cotraining with IMAGENET labels, and (v) using larger ResNet models.

2. Related work

Self-supervised image representation learning. The self-supervised signal is provided through a pretext task (e.g. converting the problem to a supervised problem), such as reconstructing the input [31], predicting the spatial context [12, 56], learning to colorize the image [80], or predict the rotation of the image [21]. Other popular approaches include clustering [5, 82, 17] and recently generative modeling [14, 41, 15]. A promising recent line of work casts the problem as mutual information maximization of representations of different views of the same image [69]. These views can come from augmentations or corruptions of the same input image [33, 2, 6, 27], or by considering different color channels as separate views [67].

Representation learning from videos. The order of frames in a video provides a useful learning signal [54, 45, 18, 74]. Information from temporal ordering can be combined with spatial ordering to infer object relations [72] or co-occurrence statistics [34]. Other pretext tasks include predicting the playback rate [77], or clustering [46].

Orthogonal to the pretext tasks, one could use the paradigm of *slow feature learning* in videos. This line of work dates back to [75], which developed a method to learn slow varying signals in time series, and inspired several recent works [26, 38, 85, 23]. Our loss at the frame level uses insights from *slow feature learning* in the form of temporal coherence [55, 58].

Tracking patches is an alternative form of supervision which has some similarity with our object level loss. For example, [70, 72, 19] learn temporally coherent representations for the patches. Our method learns representations for the objects within a fully convolutional neural network. Other approaches have investigated learning specific structures to represent the objects in video [52, 25].

Predicting the next frame or learning to synthesize a (future) frame were also considered as pretext tasks [24, 65, 51]. Given that frame prediction requires learning of fine-detailed features, one can predict only the moving pixels [77], or turn to time-agnostic prediction [37].

Object level supervision. In terms of self-supervision we follow [68] to learn from the natural hierarchy present in the videos and make use of the losses studied therein. In contrast to [68] we incorporate object-level information in the final loss and show that it leads to benefits both for few-shot transfer learning and out-of-distribution generalization. Incorporating the pixel, object, and patch information for learning and improving *video representations* was also considered in [73, 70, 85, 84, 19]. In contrast to these works, we do not rely on a strong tracker trained on a similar distribution, but on an off-the-shelf, parameter efficient object detector. Furthermore, we learn representations for images, not videos. Contemporary works also use object information for learning video representations [44] or for training graph neural networks on videos [71].

3. Method

Self-supervision via video-shot-frame hierarchy. A video can be decomposed into a hierarchy of shots, frames and objects which is illustrated in Figure 2. For the first two levels in the hierarchy, we follow the setup from [68], named *VIVI*, which we summarize here.

At the **shot level**, *VIVI* learns, in a contrastive manner, to embed shots such that they are predictive of other shots in the same video [69]. At the frame level, *VIVI* learns to embed frames such that frames from the same shot are closer to each other relative to frames from other shots. Following the findings in [68], the shot level loss is an instance of the InfoNCE loss [69] between shot representations. *VIVI* (1) maps frame k in shot ℓ to a representation $f^i_{k,\ell}$ in video i, (2) aggregates these frame representations into a shot representation s^i_ℓ, and (3) predicts the representation of the next shot, $\hat{s}^i_{\ell+1}$, given the sequence of preceding shots, $s^i_{1:\ell}$, resulting in the loss:

$$\mathcal{L}_{\text{shot}} = -\frac{1}{N}\sum_{i,\ell,m} \log \frac{e^{g(\hat{s}^i_{\ell+m}, s^i_{\ell+m})}}{\frac{1}{N}\sum_j e^{g(\hat{s}^i_{\ell+m}, s^j_{\ell+m})}}, \qquad (1)$$

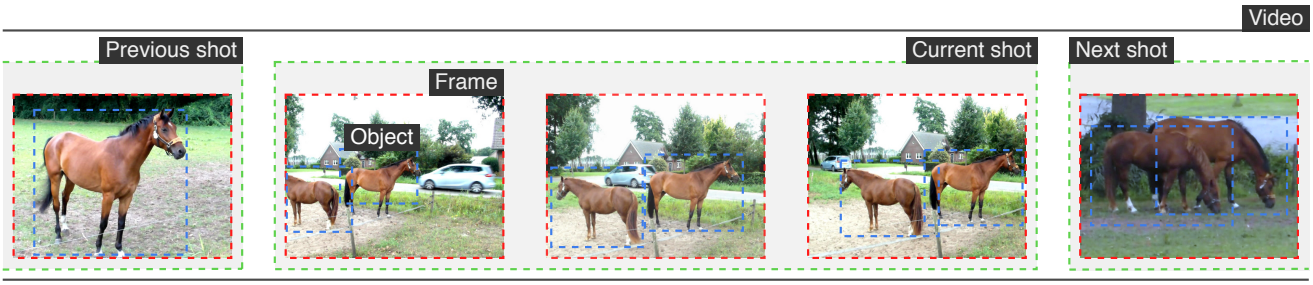

Figure 2: Learning from the natural hierarchy present in the videos. Each video in a dataset consists of multiple shots (indicated in the gray boxes), each shot consists of multiple frames. This hierarchy can be used to formulate a contrastive loss for learning image representations [68] (cf. Section 3). We extend this hierarchy to the object level by using an off-the-shelf detector.

where $g(\cdot, \cdot)$, called the critic, is used to compute similarity between shots, N indicates the total numbers of videos in a mini batch, and m indicates the number of prediction steps into the future. In practice, optimization is more stable when contrasting against shot representations from the entire batch of videos.

Contrastive learning is also applied at the **frame level** based on the intuition that frames within a shot typically contain similar scenes. *VIVI* learns to assign a higher similarity to frame representations coming from the same shot by applying a triplet loss defined in [62] (cf. Figure 2). In particular, for a fixed frame, frames coming from the same shot are considered as positives, while the frames from other shots are negatives. Denoting positive pairs as $f^p_{k,\ell}$ and negatives as $f^n_{k,\ell}$, the semi-hard loss can be written as [62]:

$$\mathcal{L}_{\text{frame}} = \sum_{k,\ell} \max\{\left\| e_{k,\ell} - e^p_{k,\ell} \right\|_2^2 - \|e_{k,\ell} - e^n_{k\ell}\|_2^2 + \alpha, 0\}$$

Extending the hierarchy with object-level losses. Data augmentation is a key novelty behind recent advances in representation learning [53, 6, 67, 8]. These augmentations are usually obtained by applying synthetic transformations like random cropping, left-right flipping, color distortion, blurring, or adding noise. However, independent non-rigid movement of an object against background, as seen in real video data, is hard to expect from synthetic augmentations.

In order to exploit these natural augmentations, which occur in video, we use a contrastive loss that encourages representations of objects of the same category to be closer together as opposed to representations of different categories (cf. Figure 1). To construct this loss, we apply an off-the-shelf object detector to all frames and extract the bounding boxes and class labels. Given the representations of each bounding box (will be discussed later), we use a triplet loss where objects from the same class form positive pairs, and objects from different classes form negative pairs. In particular, given the embedding of the b-th bounding box r_b, and the embeddings of the positive r_p and negative r_n (with respect to r_b), we apply the following loss for each frame:

$$\mathcal{L}_{\text{OBJECT}} = \sum_b \max\{\|r_b - r_p\|_2^2 - \|r_b - r_n\|_2^2 + \alpha, 0\} \ . \quad (2)$$

To scale the triplet loss to a large number of boxes, we follow insights from the literature [84, 43, 64] and use semi-hard negative mining [62].

An alternative to the contrastive loss is the classic cross-entropy loss — learning a representation such that a linear classifier can recognize the target class. We formalize the former given the empirical evidence from recent research [40], but we present ablation studies in Section 4.

Representations of bounding boxes. A simple approach to obtain the representation would be to extract all the bounding boxes and feed them through the network. However, this is computationally prohibitive, and we instead propose a method which reuses the feature grid present in RESNET50 models [28] illustrated in Figure 3.

Consider an image x of dimensions $H \times W \times C$, indicating the height, width, and number of channels of the image, respectively. A fully convolutional RESNET50 maps this image to a feature grid x' of dimensions $H' \times W' \times C'$. We represent the bounding box with center index (i, j) by the vector

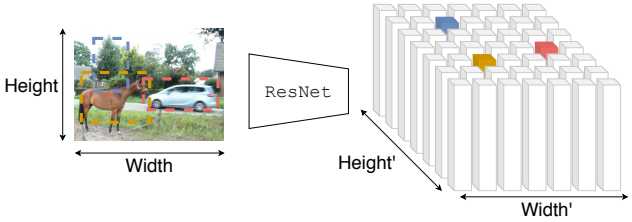

Figure 3: Illustration of the feature grid output by a RESNET50. Object bounding boxes are mapped to feature columns that correspond to the center pixel of that bounding box. Correspondence is illustrated using matching colors.

$r = x'\left[\lfloor i\frac{H'}{H}\rfloor, \lfloor j\frac{W'}{W}\rfloor\right]$ of size C'. This approach is conceptually similar to max pooling as used in Fast-RCNN [22], and reminisces of [81]. Given the computational efficiency and the fact that the effective receptive field is concentrated at the center [50], we chose this simple alternative.

Final loss function. We combine the losses using positive coefficients ω and β as

$$\mathcal{L} = \omega\mathcal{L}_{\text{OBJECT}} + \mathcal{L}_{\text{FRAME}} + \beta\mathcal{L}_{\text{SHOT}}. \quad (3)$$

This formulation enables a study of the benefits of each of the losses and leads to practical recommendations.

Headroom analysis using IMAGENET. In addition to the proposed method, we analyze the benefits of co-training the proposed network with a large labeled data source [68, 4]. This data source provides a vast quantity of labeled images and should help the model improve the performance on tasks which require fine-grained detail of specific object classes. In particular, we consider an affine map of the representation extracted by the network, followed by a softmax layer and a corresponding cross-entropy resulting in $\mathcal{L}_{\text{TOTAL}} = \mathcal{L} + \gamma\mathcal{L}_{\text{SUPERVISED}}$, where γ is a hyperparameter balancing the impact of this additional loss.

4. Experimental setup

4.1. Architectures and training details

Unless otherwise specified, all experiments are performed on a RESNET50 V2 [28] with batch normalization. For the shot prediction function, we use a LSTM with 256 hidden units. We parameterize the critic function g as a bilinear form. All frames are augmented using the same policy as [66], using random cropping, left-right flipping and color distortions. The coordinates of object bounding boxes are recalculated accordingly. All models are trained using a batch size of 512 for 120K iterations of stochastic gradient descent with a momentum constant of 0.9. The learning rate starts as 0.8 and decreases by a factor of 10 after 90k and 110k training steps. When cotraining, we train for 100k iterations and decrease the learning rate after 70k, 85k and 95k iterations. Shots and frames are sampled using the same method as [68]: for each video, we sample a sequence of four shots, and we sample eight frames from each shot.

The coefficients ω and β weigh the loss contributions in Equation (3). We set $\beta = 0.04$ following [68], and $\omega = 5$, although we found that a wider range of values leads to the same performance (cf. Figure 6 in the Appendix).

Cotraining details. The experiments on cotraining use group normalization [76] with weight standardization [57], instead of batch normalization, for a fair comparison to [68]. When cotraining, we sample at every step a batch from each dataset — we compute the three-level loss (3) on the sampled videos, and the classification log-loss on the sampled IMAGENET images. Cotrained models train with batch size of 512 for videos and 2048 for images for 100k iterations, using the learning rate schedule described above. Images are preprocessed using the *inception crop* function from [66].

Datasets. We train on videos from the YT8M dataset and cotrain with IMAGENET [9]. The videos are sampled at 1Hz and we run the detector, a MobileNet [61], with a single shot multi box detector [49], trained on OPENIMAGESV4 [1]. The detector runs at 19ms per frame on a V100 GPU. We detected the objects offline and stored the annotated videos to disk for use during training. Table 6 in the appendix shows how often common objects are detected in the video frames. As the detector has been trained on OPENIMAGESV4, we use its 600 category label space for constructing positive and negative pairs for $\mathcal{L}_{\text{OBJECT}}$. We use the feature grid from the ResNet block 4 to construct representations for objects in a frame and limit the number of objects in each frame to a maximum of 5. We discard objects with detection score below 0.05, which accounts for approximately 3% of the detected objects. Figure 5 shows a histogram of the detection scores. Finally, given that the YT8M dataset is a dynamic dataset, our training set contains the videos still available in May 2020: 3.3 million training and one million validation videos. The baselines were re-trained on this new dataset.

4.2. Evaluation

We evaluate two aspects of the learned representations: How well the representations transfer to novel classification tasks, and how robust are the resulting classifiers to distribution shifts.

Transferability. The main objective of this work is learning image representations that transfer well to novel, previously unseen tasks. To empirically validate our approach, we report the results for transfer learning on the Visual Task Adaptation Benchmark (VTAB), a suite of 19 image classification tasks [79]. The tasks are organized into three categories, *Natural* containing commonly used classification datasets (Caltech101, Cifar-100, DTD, Flowers102, Pets, SUN397 and SVHN), *Specialized* comprising of images recorded with specialized equipment (Resisc45, EuroSAT, Patch Camelyon, Diabetic Retinopathy), and *Structured* containing scene understanding tasks (CLEVR-dist, CLEVR-count, dSPRITES-orient, dSPRITES-pos, sNORB-azimuth, sNORB-elevation, DMLab, KITTI). For more details and references to the respective datasets, please refer to [79].

We consider transfer learning in the low data regime, where each task has only 1000 labeled samples available.

METHOD	DATASET	SIGNAL	VTAB	NATURAL	SPECIALIZED	STRUCTURED
Transitive Invariance [72]	YouTube 100k	Tracklets	44.2	35.0	61.8	43.4
MT [13]	IMAGENET & SoundNet	Tracklets	59.2	51.9	78.9	55.8
Supervised (RESNET50)	IMAGENET	None	**68.5**	**71.3**	**83.0**	**58.9**
Detector backbone	OPENIMAGESV4	None	61.6	60.0	80.4	53.5
VIVI [68]	YT8M	None	60.9 †	55.0 †	79.5 †	56.7 †
OURS	YT8M	Detector	**64.1** †	**59.0** †	**81.6** †	**59.8** †
Boxes and labels at random	YT8M	None	60.3	55.2	78.0	55.0
Boxes at random coordinates	YT8M	Detector	63.4	57.5	81.1	59.7
Distilling from IMAGENET	YT8M	Classifier	63.1	59.6	**81.6**	57.0
Also predict cross entropy	YT8M	Detector	**64.9**	**60.5**	81.3	**60.5**

Table 1: Evaluation on the Visual Task Adaptation Benchmark. Each number indicates the average classification accuracy over all data sets in the corresponding category. † indicates a statistical significant difference between the baseline [68] and the proposed method. The last 4 methods show results of 4 ablation studies which investigate the benefits of the corresponding training signals.

The evaluation protocol is the same as in [68, 42, 79, 60]: for each dataset we (i) train on 800 training samples using our learned model as initialization, (ii) sweep over two learning rates (0.1, 0.01) and two learning rate schedules (10K steps with decay every 3K, or 2.5K steps with decay every 750), (iii) pick the learning rate and learning rate schedule according to the highest validation accuracy on the 200 validation samples, and then (iv) retrain the model using all 1000 samples. We report statistical significance at the $p = 0.05$ level on a Welch's two sided t-test based on 12 independent runs of the transfer protocol. The error bars in the diagrams indicate bootstrapped 95% confidence intervals.

Robustness. As discussed in Section 3, we were guided by the intuition that the model should learn to be more invariant to natural augmentations. We thus expect our model to be more robust and generalize better to out-of-distribution (OOD) images.

We follow two recent studies on OOD generalization [29, 11] and evaluate robustness as accuracy on a suite of 8 datasets measuring various robustness aspects. These datasets are defined on the IMAGENET label space: (1) IMAGENET-A [30] measures the accuracy on samples from the web that were adversarial to a RESNET50 trained from scratch on IMAGENET. (2) IMAGENET-C [29] measures the accuracy on samples from IMAGENET under perturbations such as blur, pixelation, and compression artifacts. (3) IMAGENET-V2 [59] presents a new test set for the IMAGENET dataset. (4) OBJECTNET [3] consists of images collected by crowd sourcing, where participants were asked to photograph objects in unusual poses and unusual backgrounds. (5-8) IMAGENET-VID, IMAGENET-VID-pm-k, YT-BB-ROBUST, and YT-BB-ROBUST-pm-k present frames from video sequences [63]. We measure both accuracy of the anchor frame, denoted as anchor accuracy, and worst-case accuracy in the 20 neighboring frames (i.e. if any frame from the set of 10 preceding and 10 following frames is misclassified, the video is judged as misclassified), denoted as pm-k.

We also evaluate our models on the texture-shape data set from [20]. Our method uses a contrastive loss to learn specifically from objects. Learning with our loss encourages objects in different appearances to have similar representations. As such, we hypothesize that our models have higher shape bias, compared to texture bias. [20] provide a dataset to measure the texture-shape bias. The test set consists of 1280 images whose texture has been stylized. Each image has a label according to its shape, and a label according to the stylization of its texture. We report the fraction of correct predictions based on shape, as proposed by the authors. For further details we refer to the paper [20].

5. Results

5.1. Transferability

Table 1 shows our results on the Visual Task Adaptation Benchmark (VTAB). We observe statistically significant increases in accuracy over the baseline [68] which demonstrate the benefits of supplementing the self-supervised hierarchy with object level supervision. The results per dataset are presented in Figure 4.

Rows 1 and 2 in Table 1 compare against two prior works on representation learning from videos: Transitive Invariance (TI) [72] and Multi-task Self-Supervised Visual Learning (MT) [13]. TI uses context based self-supervision together with tracking in videos to formulate a pretext task for representation learning and row 1 shows the performance of

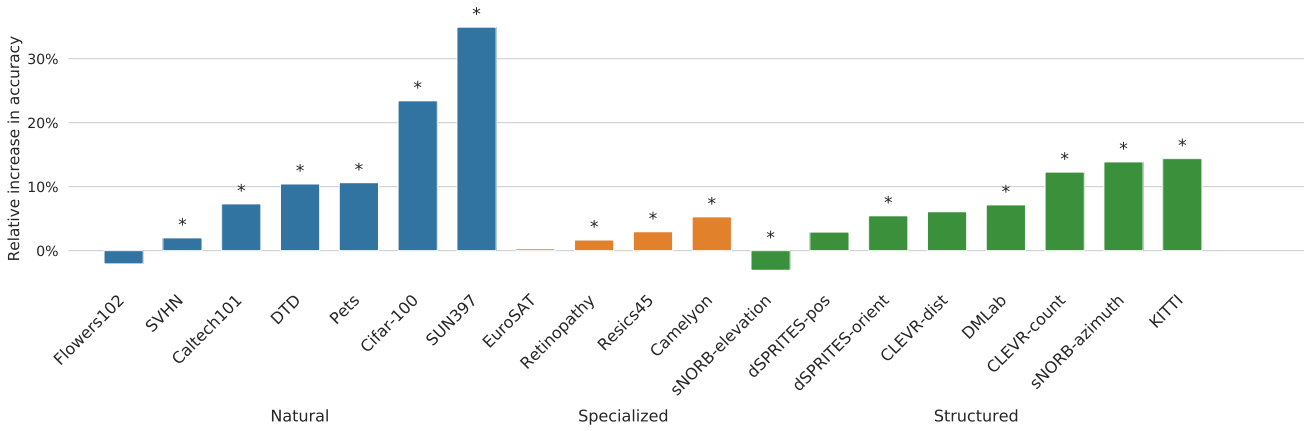

Figure 4: Relative increase in VTAB accuracy per dataset: blue for Natural, orange for Specialized and green for Structured datasets. Stars indicate statistical significance. Relative increase refers to the increase in accuracy by learning from objects, divided by accuracy of VIVI [68], which only learns at two levels of the hierarchy.

their pre-trained VGG-16 checkpoint. MT uses a variety of pretext tasks, including motion segmentation, coloring and exemplar learning [16] and row 2 shows the performance of their RESNET101 (up to block 3) checkpoint.

Ablation 1: Randomizing the location and the class. The object level loss $\mathcal{L}_{\text{OBJECT}}$ is made possible through additional supervision provided via an object detector pre-trained on OPENIMAGESV4. The detector contributes to representation learning by annotating object positions and object category labels in video frame and here we ablate these two sources: (i) We evaluate the contribution (1) from knowing the class of an object, but not its coordinates, and (2) when neither the class nor the location are known.

The results are detailed in Table 1. Randomizing both the label and the coordinates of the objects destroys all signal from the detector. Row *Boxes and labels at random* shows the results of this ablation and we observe that the performance is below the VIVI baseline, as expected. In contrast, when we randomize the object locations, but maintain the correct labels, we obtain an improvement over the baseline (row *boxes at random*). Interestingly, the VTAB score on structured datasets, 59.7%, equals the accuracy where both the class and location are known.

Ablation 2: Frame-level labels from a IMAGENET-pretrained model. We further investigate the effectiveness of knowing frame-level labels by obtaining soft-labels using an IMAGENET-pretrained model, effectively distilling the IMAGENET model on YT8M frames [32]. Its performance is noted in Table 1, row *distilling from* IMAGENET. Interestingly, this distilled model scores higher in natural datasets, but lower in structured datasets than the proposed method. These differences show how various upstream signals affect different downstream tasks differently.

Method	0	1k	20k	120k
VIVI	39.1	49.0 †	58.9 †	59.6 †
OURS	39.8	54.5 †	63.4 †	66.0 †

Table 2: Fraction of objects whose nearest neighbor in representation space is an object with the same class label for increasing number of training steps. † indicates a statistically significant difference at $p = 0.001$ using Fisher's exact test for all objects in 50 batches of 8 videos. As training progresses, our method has significantly more objects with matching neighbors than the vanilla VIVI model.

Ablation 3: Distilling the object detector. We distill a RESNET50 on YT8M where the training instances are cropped objects and the labels assigned by the object detector. The distilled RESNET50 achieves a score of 57.1% VTAB score compared to 64.1% of the proposed method. At the same time, using a non-pretrained ResNet of the same capacity achieves 42.1% when trained on 1000 downstream labels. Hence, the detector clearly provides a strong training signal, but it can be exploited to a higher degree by coupling it with a self-supervised loss as in the proposed method.

Ablation 4: Semi-supervised learning. One can also utilize the tagged frames as labelled training data [40]. To this end, we use a linear classifier to classify the bounding box representations r_b as one of 600 OPENIMAGESV4 classes using a binary cross-entropy loss added to the loss in Equation (3). This approach increases the VTAB score from 64.1% to 64.9%. We also investigated using this loss as a *replacement* for $\mathcal{L}_{\text{OBJECT}}$ in Equation (3). However, this performed worse, scoring 63.7%, which highlights the advantage of the contrastive formulation.

Method	mAP(%)
Our method ($\mathcal{L}_{\text{OBJECT}} + \mathcal{L}_{\text{FRAME}} + \beta\mathcal{L}_{\text{SHOT}}$)	40.4
VIVI ($\mathcal{L}_{\text{FRAME}} + \beta\mathcal{L}_{\text{SHOT}}$)	35.1
Only object level ($\mathcal{L}_{\text{OBJECT}}$)	39.3

Table 3: RETINANET performance on the MS-COCO dataset using various pre-trained backbones.

Effect of the contrastive loss. Lastly, we present a diagnostic for our training procedure at the object level. $\mathcal{L}_{\text{OBJECT}}$ is designed to embed representations for objects of the same class closer together. We verify whether this is indeed the case by measuring the fraction of nearest neighbors for each representation that belongs to the same category. Table 2 shows the progression of this metric during training, in comparison to a VIVI model trained in tandem. Our method results in a significantly higher fraction, indicating that more nearest neighbors belong to the same class as the query object. This verifies that our loss function and training procedure achieve the desired outcome.

Evaluation on detection. Our model learns from videos at the object level. It is natural to expect that a RESNET50 backbone pre-trained using our method will perform well when fine-tuned for downstream object detection. To this end, we fine-tune a RETINANET architecture [47] on the MS COCO object detection dataset [48]. Images are rescaled and randomly cropped to 640×640 during training. We train the model for 60 epochs with an initial learning rate of 0.08 and batch size 256.

Results are shown in Table 3. Pre-training using our method improves upon the VIVI baseline by 5.3% mAP points. Training on only the object level loss scores 1.1% mAP point lower compared to using all three levels of the hierarchy. These results suggest that the learned representations are indeed more object centric and that learning from all three levels combined yields representations more effective for downstream object detection.

Co-training with IMAGENET. Table 4 shows the resulting accuracies on VTAB when cotraining with IMAGENET. Compared to the cotrained VIVI baseline, our method with its object-level loss increases the VTAB score from 69.0% to 69.4%. This increase in accuracy is modest in comparison to those in Table 1. We argue that IMAGENET is a clean curated dataset whereas YT8M is noisy. Adding cotraining with clean IMAGENET improves the accuracy on natural datasets from 60.9% to 70.3%. It is not surprising that adding more noisy supervision, at the object level, does not give massive gains in this setting. We repeat the experiment with a higher capacity RESNET50. Again we observe modest, but statistically significant, improvements over VIVI. The largest improvement is on the structured datasets, which increase from 62.2% to 63.0%. These experiments highlight an interesting dichotomy between natural and structured subsets of VTAB: learning with IMAGENET yields improvements on natural datasets, while using the detector yields improvements for structured datasets.

METHOD	VTAB	NATURAL	SPEC.	STRUC.
ResNet-50				
VIVI [68]	69.0 †	70.3 †	82.7 †	60.9
OURS	69.4 †	70.8 †	82.9 †	61.4
3x wider ResNet-50				
VIVI [68]	70.2 †	71.4 †	83.7	62.2 †
OURS	70.5 †	71.6 †	83.6	63.0 †

Table 4: VTAB scores of the models cotrained on YT8M and IMAGENET. The presented numbers are the average image classification accuracy of the fine-tuned models over the respective VTAB category. † notes a statistically significant difference between VIVI and our method.

5.2. Robustness

Table 5 presents the classification accuracies on the eight robustness datasets. To get predictions in the IMAGENET label space, we fine-tune our learned representation and report results in row *fine tuning*.

Our method compares favorably to the baseline on all datasets, which confirms the intuition that extending the video-shot-frame hierarchy to objects results in more robust image representations. The robustness results for the cotrained models are presented in Table 5, row *cotraining*. As expected, the results improve across all datasets. The final two columns of Table 5 note the delta between anchor accuracy and `pm-k` accuracy. A lower delta indicates a more robust model, irrespective of the anchor accuracy. In three out of four cases our method scores a lower (better) delta.

We evaluate our models on the texture-shape data set from [20]. As the evaluation is done using the IMAGENET label space, we use the same models that we evaluated on the robustness datasets. First, we evaluate the fine-tuned models. The VIVI model, using only the video-shot-frame hierarchy, scores 20.6 shape fraction on the provided dataset. Using our method to learn from the video-shot-frame-object hierarchy, the shape fraction increases to 24.0. A higher shape fraction indicates a better model, as the network has higher relative accuracy according to the shape of the object depicted. Similarly, cotrained models improve from 25.8 to 28.5 when using our method to learn from objects in video. These results indicate a promising direction for future research.

MODEL	METHOD	IMAGENET [10]	IMAGENET-A [30]	IMAGENET-C [29]	IMAGENET-V2 [59]	OBJECTNET [3]	IMAGENET-VID [63]	IMAGENET-VID-pm-k [63]	YT-BB [63]	YT-BB-pm-k [63]	Δ IMAGENET-VID	Δ YT-BB
		↑	↑	↑	↑	↑	↑	↑	↑	↑	↓	↓
VIVI	Fine tuning	62.6	0.5	6.8	51.1	16.2	57.9	36.5	58.0	39.9	21.4	**18.1**
OURS	Fine tuning	**65.2**	**0.6**	**9.5**	**53.4**	**18.4**	**61.7**	**43.4**	**60.8**	**42.3**	**18.3**	18.6
VIVI	Cotraining	73.1	1.1	24.3	59.8	20.9	58.7	41.7	49.4	35.4	17.0	14.0
OURS	Cotraining	**73.3**	**1.2**	**24.4**	**60.8**	**21.0**	**59.7**	**44.0**	**50.0**	**37.6**	**15.7**	**12.4**

Table 5: Accuracy on robustness datasets from literature. These datasets are typically a perturbed version of ImageNet-like images and videos. Each dataset indicates a specific aspect of robustness. Higher accuracy corresponds to better robustness. Lower Δ corresponds to better robustness. *Cotraining* refers to a model trained on YT8M and IMAGENET.

6. Discussion

We presented a hierarchy, videos-shots-frames-objects, to learn representations from video at multiple levels of granularity. The learned representations transfer better to downstream image classification tasks and exhibit higher accuracy on out-of-distribution datasets. We identify three aspects for future research.

A taxonomy for learning transferable representations. Our results show that using different signals from videos benefits transfer learning to Natural, Specialized or Structured image classification tasks in a specific manner. We consider our work in a larger line of research that creates a taxonomy for learning methods and their effect on transfer learning to specific datasets, similar to [78], which outlined a taxonomy of multi modal learning. To give examples: Using the noisy videos from YT8M mainly improves transferability to Specialized and Structured tasks on the VTAB benchmark. Using the clean images from IMAGENET improves transferability to Natural tasks. Our method, which receives implicit supervision from OPENIMAGESV4, shows highest improvement on Natural tasks. Thus using different sources of supervision improves the transferability to different tasks. We believe that understanding how different data and learning methods impact the performance on different data domains is a central research question in transfer learning, and that this work contributes towards this grand challenge by providing insight into the benefits of learning from uncurated video data.

Learning about objects without labels. Our method uses an off-the-shelf detector to identify the objects. As the detector was trained on labeled data, learning at the object level of the hierarchy uses implicit supervision. Contemporary literature focused on other self-supervised methods to improve learning from video. For example, one could derive signals about objects using optical flow or keypoint detection [35, 36]. Combining these ideas in our paradigm of learning in the hierarchy might provide a useful research direction.

Learning about entire videos instead of image representations. Our method shows improved results concerning transfer learning and robustness of image models for single images. This improvement raises the question how these results will translate to video understanding. Recently, there has been interest in video recognition [39] and video action localization [83]. We look forward to testing our learning methods on these tasks.

Improved robustness from learning about objects. We have shown how our method results in more robust image classifiers. This observation suggests that learning about objects, invariant to other parts of the images, improves robustness. Several computer vision tasks concern objects. Therefore, we suggest that having object centered representation will contribute to developments in robustness.

Acknowledgements

We thank Justin Gilmer and Chen Sun for helpful discussions. RR thanks Mario Lucic and Neil Houlsby for mentoring this research as part of the AI residency, and thanks Basil Mustafa for encouraging conversations.

References

[1] TensorFlow Object Detection API. Tensorflow hub: Open images v4 with ssd. `https://tfhub.dev/google/openimages_v4/ssd/mobilenet_v2/1`. Accessed: 2020-07-22.

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

[3] Andrei Barbu, David Mayo, Julian Alverio, William Luo, Christopher Wang, Dan Gutfreund, Josh Tenenbaum, and Boris Katz. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. In *Advances in Neural Information Processing Systems*, 2019.

[4] Lucas Beyer, Xiaohua Zhai, Avital Oliver, and Alexander Kolesnikov. S4L: self-supervised semi-supervised learning. In *International Conference on Computer Vision*, 2019.

[5] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *European Conference on Computer Vision*, 2018.

[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. *arXiv*, abs/2002.05709, 2020.

[7] Xinlei Chen, Haoqi Fan, Ross B. Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv*, abs/2003.04297, 2020.

[8] Ekin Dogus Cubuk, Barret Zoph, Dandelion Mané, Vijay Vasudevan, and Quoc V. Le. Autoaugment: Learning augmentation policies from data. *arXiv*, abs/1805.09501, 2018.

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. Imagenet: A large-scale hierarchical image database. In *Conference on Computer Vision and Pattern Recognition*, 2009.

[10] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. Imagenet: A large-scale hierarchical image database. In *Conference on Computer Vision and Pattern Recognition*, 2009.

[11] Josip Djolonga, Jessica Yung, Michael Tschannen, Rob Romijnders, Lucas Beyer, Alexander Kolesnikov, Joan Puigcerver, Matthias Minderer, Alexander D'Amour, Dan Moldovan, Sylvan Gelly, Neil Houlsby, Xiaohua Zhai, and Mario Lucic. On robustness and transferability of convolutional neural networks. *arXiv*, abs/2007.08558, 2020.

[12] Carl Doersch, Abhinav Gupta, and Alexei A. Efros. Unsupervised visual representation learning by context prediction. In *International Conference on Computer Vision*, 2015.

[13] Carl Doersch and Andrew Zisserman. Multi-task self-supervised visual learning. In *International Conference on Computer Vision*, 2017.

[14] Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. In *International Conference on Learning Representations*, 2017.

[15] Jeff Donahue and Karen Simonyan. Large scale adversarial representation learning. In *Advances in Neural Information Processing Systems*, 2019.

[16] Alexey Dosovitskiy, Philipp Fischer, Jost Tobias Springenberg, Martin A. Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with exemplar convolutional neural networks. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 2016.

[17] Alexey Dosovitskiy, Jost Tobias Springenberg, Martin A. Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with convolutional neural networks. In *Advances in Neural Information Processing Systems*, 2014.

[18] Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould. Self-supervised video representation learning with odd-one-out networks. In *Conference on Computer Vision and Pattern Recognition*, 2017.

[19] Ruohan Gao, Dinesh Jayaraman, and Kristen Grauman. Object-centric representation learning from unlabeled videos. In *Asian Conference on Computer Vision*, 2016.

[20] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*, 2019.

[21] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *International Conference on Learning Representations*, 2018.

[22] Ross B. Girshick. Fast R-CNN. In *International Conference on Computer Vision*, 2015.

[23] Daniel Gordon, Kiana Ehsani, Dieter Fox, and Ali Farhadi. Watching the world go by: Representation learning from unlabeled videos. *arXiv*, abs/2003.07990, 2020.

[24] Ross Goroshin, Joan Bruna, Jonathan Tompson, David Eigen, and Yann LeCun. Unsupervised feature learning from temporal data. In *International Conference on Learning Representations*, 2015.

[25] Vincent Le Guen and Nicolas Thome. Disentangling physical dynamics from unknown factors for unsupervised video prediction. *arXiv*, abs/2003.01460, 2020.

[26] Tengda Han, Weidi Xie, and Andrew Zisserman. Video representation learning by dense predictive coding. In *International Conference on Computer Vision*, 2019.

[27] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. *arXiv*, abs/1911.05722, 2019.

[28] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European Conference on Computer Vision*, 2016.

[29] Dan Hendrycks and Thomas G. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv*, abs/1807.01697, 2018.

[30] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. *arXiv*, abs/1907.07174, 2019.

[31] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 2006.

[32] Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. *arXiv*, abs/1503.02531, 2015.

[33] R. Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Philip Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. In *International Conference on Learning Representations*, 2019.

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

[35] Allan Jabri, Andrew Owens, and Alexei A. Efros. Space-time correspondence as a contrastive random walk. *arXiv*, abs/2006.14613, 2020.

[36] Tomas Jakab, Ankush Gupta, Hakan Bilen, and Andrea Vedaldi. Self-supervised learning of interpretable keypoints from unlabelled videos. In *Conference on Computer Vision and Pattern Recognition*, 2020.

[37] Dinesh Jayaraman, Frederik Ebert, Alexei A. Efros, and Sergey Levine. Time-agnostic prediction: Predicting predictable video frames. In *International Conference on Learning Representations*, 2019.

[38] Dinesh Jayaraman and Kristen Grauman. Slow and steady feature analysis: Higher order temporal coherence in video. In *Conference on Computer Vision and Pattern Recognition*, 2016.

[39] Will Kay, João Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset. *arXiv*, abs/1705.06950, 2017.

[40] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *arXiv*, abs/2004.11362, 2020.

[41] Diederik P. Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. In *Advances in Neural Information Processing Systems*, 2014.

[42] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Large scale learning of general visual representations for transfer. *arXiv*, abs/1912.11370, 2019.

[43] Alexander Kolesnikov, Xiaohua Zhai, and Lucas Beyer. Revisiting self-supervised visual representation learning. In *Conference on Computer Vision and Pattern Recognition*, 2019.

[44] Zihang Lai and Weidi Xie. Self-supervised video representation learning for correspondence flow. In *British Machine Vision Conference*, 2019.

[45] Hsin-Ying Lee, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Unsupervised representation learning by sorting sequences. In *International Conference on Computer Vision*, 2017.

[46] Hyodong Lee, Joonseok Lee, Joe Yue-Hei Ng, and Paul Natsev. Large scale video representation learning via relational graph clustering. In *Conference on Computer Vision and Pattern Recognition*, 2020.

[47] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *International Conference on Computer Vision*, 2017.

[48] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In *European Conference on Computer Vision*, 2014.

[49] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: single shot multibox detector. In *European Conference on Computer Vision*, 2016.

[50] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard S. Zemel. Understanding the effective receptive field in deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, 2016.

[51] Michaël Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. In *International Conference on Learning Representations*, 2016.

[52] Matthias Minderer, Chen Sun, Ruben Villegas, Forrester Cole, Kevin P. Murphy, and Honglak Lee. Unsupervised learning of object structure and dynamics from videos. In *Advances in Neural Information Processing Systems*, 2019.

[53] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. *arXiv*, abs/1912.01991, 2019.

[54] Ishan Misra, C. Lawrence Zitnick, and Martial Hebert. Shuffle and learn: Unsupervised learning using temporal order verification. In *European Conference on Computer Vision*, 2016.

[55] Hossein Mobahi, Ronan Collobert, and Jason Weston. Deep learning from temporal coherence in video. In *International Conference on Machine Learning*, 2009.

[56] Mehdi Noroozi, Hamed Pirsiavash, and Paolo Favaro. Representation learning by learning to count. In *International Conference on Computer Vision*, 2017.

[57] Siyuan Qiao, Huiyu Wang, Chenxi Liu, Wei Shen, and Alan L. Yuille. Weight standardization. *arXiv*, abs/1903.10520, 2019.

[58] Vignesh Ramanathan, Kevin D. Tang, Greg Mori, and Fei-Fei Li. Learning temporal embeddings for complex video analysis. In *International Conference on Computer Vision*, 2015.

[59] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International Conference on Machine Learning*, 2019.

[60] Google Research. Github: Task adaptation. `https://github.com/google-research/task_adaptation`. Accessed: 2020-07-22.

[61] Mark Sandler, Andrew G. Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Conference on Computer Vision and Pattern Recognition*, 2018.

[62] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Conference on Computer Vision and Pattern Recognition*, 2015.

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

[64] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *Conference on Computer Vision and Pattern Recognition*, 2016.

[65] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhutdinov. Unsupervised learning of video representations using lstms. In *International Conference on Machine Learning*, 2015.

[66] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Conference on Computer Vision and Pattern Recognition*, 2015.

[67] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *arXiv*, abs/1906.05849, 2019.

[68] Michael Tschannen, Josip Djolonga, Marvin Ritter, Aravindh Mahendran, Neil Houlsby, Sylvain Gelly, and Mario Lucic. Self-supervised learning of video-induced visual invariances. In *Conference on Computer Vision and Pattern Recognition*, 2020.

[69] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv*, abs/1807.03748, 2018.

[70] Xiaolong Wang and Abhinav Gupta. Unsupervised learning of visual representations using videos. In *International Conference on Computer Vision*, 2015.

[71] Xiaolong Wang and Abhinav Gupta. Videos as space-time region graphs. In *European Conference on Computer Vision*, 2018.

[72] Xiaolong Wang, Kaiming He, and Abhinav Gupta. Transitive invariance for self-supervised visual representation learning. In *International Conference on Computer Vision*, 2017.

[73] Xiaolong Wang, Allan Jabri, and Alexei A. Efros. Learning correspondence from the cycle-consistency of time. In *Conference on Computer Vision and Pattern Recognition*, 2019.

[74] Donglai Wei, Joseph J. Lim, Andrew Zisserman, and William T. Freeman. Learning and using the arrow of time. In *Conference on Computer Vision and Pattern Recognition*, 2018.

[75] Laurenz Wiskott and Terrence J. Sejnowski. Slow feature analysis: Unsupervised learning of invariances. *Neural Computation*, 2002.

[76] Yuxin Wu and Kaiming He. Group normalization. *International Journal Computer Vision*, 2020.

[77] Yuan Yao, Chang Liu, Dezhao Luo, Yu Zhou, and Qixiang Ye. Video playback rate perception for self-supervised spatio-temporal representation learning. In *Conference on Computer Vision and Pattern Recognition*, 2020.

[78] Amir Roshan Zamir, Alexander Sax, William B. Shen, Leonidas J. Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In *Conference on Computer Vision and Pattern Recognition*, 2018.

[79] Xiaohua Zhai, Joan Puigcerver, Alexander Kolesnikov, Pierre Ruyssen, Carlos Riquelme, Mario Lucic, Josip Djolonga, Andre Susano Pinto, Maxim Neumann, Alexey Dosovitskiy, Lucas Beyer, Olivier Bachem, Michael Tschannen, Marcin Michalski, Olivier Bousquet, Sylvain Gelly, and Neil Houlsby. The visual task adaptation benchmark. *arXiv*, abs/1910.04867, 2019.

[80] Richard Zhang, Phillip Isola, and Alexei A. Efros. Colorful image colorization. In *European Conference on Computer Vision*, 2016.

[81] Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. Tracking objects as points. In *European Conference on Computer Vision*, 2020.

[82] Chengxu Zhuang, Alex Lin Zhai, and Daniel Yamins. Local aggregation for unsupervised learning of visual embeddings. In *International Conference on Computer Vision*, 2019.

[83] Dimitri Zhukov, Jean-Baptiste Alayrac, Ramazan Gokberk Cinbis, David F. Fouhey, Ivan Laptev, and Josef Sivic. Cross-task weakly supervised learning from instructional videos. In *Conference on Computer Vision and Pattern Recognition*, 2019.

[84] Will Y Zou, Andrew Y Ng, and Kai Yu. Unsupervised learning of visual invariance with temporal coherence. In *Advances in Neural Information Processing Systems*, 2011.

[85] Will Y. Zou, Andrew Y. Ng, Shenghuo Zhu, and Kai Yu. Deep learning of invariant features via simulated fixations in video. In *Advances in Neural Information Processing Systems*, 2012.