

A. Datasets

A.1. Image datasets

ImageNet (IN1K) [70]. We use the ILSVRC 2012 challenge subset of ImageNet that has 1.28M training and 50K val images with 1000 classes. The 1000 classes cover a wide range of concepts from fine-grained species of dogs, to every day indoor and outdoor objects. The dataset is released under a non-commercial license. This subset of ImageNet is widely used for benchmarking image recognition models.

iNaturalist-2018 (iNat18) [44]. The iNaturalist dataset is a fine-grained plant and animal species classification dataset. We use the 2018 version of the dataset that has 437K training and 24K val images with 8142 classes. The dataset was collected in collaboration with iNaturalist, a citizen science effort that uses pictures submitted by people around the world. The dataset is released under a non-commercial iNaturalist license. To the best of our knowledge, no PII or harmful content has been reported in the dataset.

Places-365 (P365) [91]. The Places dataset is a scene recognition dataset that evaluates image recognition models on indoor and outdoor scene classification. The dataset consists of 1.8M training and 36K val images with 365 scene classes. The dataset has public images and is released under a non-commercial license. To the best of our knowledge, no PII or harmful content has been reported in the dataset.

A.2. Video datasets

Something Something-v2 (SSv2) [37]. This is a video action classification dataset with a special emphasis on temporal modeling. It consists of $\sim 169K$ training and $\sim 25K$ validation clips, each a few seconds long, classified into one of 174 action classes. Due to the nature of the classes considered (*e.g.* “covering something” and “uncovering something”), the dataset requires temporal modeling to correctly classify each video. The dataset has been collected by consenting participants who recorded the videos given the action label, and released under a non-commercial license. To the best of our knowledge, no PII or harmful content has been reported in the dataset.

EPIC-Kitchens-100 (EK100) [22]. This dataset consists of 100 hours total of unscripted egocentric videos. Each video is densely labeled with human-object interactions (“clips”), which consists of a start time, end time, one of 300 nouns (the object interacted with) and one of 97 verbs (the type of interaction). There are $\sim 67K$ training and $\sim 10K$ validation clips. Following prior work [22, 33], we tackle the task of recognizing the 3,806 (verb, noun) pairs given a clip. Note that not all (verb, noun) combinations occur in both training and testing data. We use this dataset as a transfer task to evaluate the learned representation. The data is released under CC-BY-NC 4.0 license. The videos were collected by consenting participants who wore egocentric

cameras while recording their daily activities, typically cooking. Given the egocentric nature of the videos, PII such as faces are not visible in the videos. To the best of our knowledge, no offensive content has been reported on this dataset.

Kinetics-400 (K400) [48]. This dataset consists of $\sim 240K$ training and $\sim 20K$ validation third-person video clips that are 10 seconds in length. Each clip is labeled into one of 400 action categories. The task requires classifying each validation video into one of these categories. The dataset is based on publicly available web videos from YouTube. Due to the videos being taken down over time, the dataset changes over time making apples-to-apples comparison with prior work difficult. Hence, we use a static dataset like SSv2 for pre-training and the primary comparisons. We will release the set of train and test videos that we had access to from this dataset. To the best of our knowledge, no PII or harmful content has been reported on this dataset.

B. Implementation Details

B.1. Details about Pretraining

We pretrain the model jointly on IN1K and SSv2 using the hyperparameters in Table 3. The dataset-specific hyperparameters are in the individual columns, and others are in the middle. These apply to ViT-B, ViT-L, and ViT-H unless specified otherwise. We use the same hyperparameters for pretraining the models on IN1K and K400 (Table 2). The ViT-H model in Table 2 is pretrained for 2400 epochs.

Config	IN1K	SSv2
Optimizer	AdamW	
Peak learning rate	3e-4	
Weight decay	0.05	
Optimizer Momentum	$\beta_1 = 0.9, \beta_2 = 0.95$ [16]	
Batch size	2048	
Sample replication	1	4
Warmup epochs	40	
Total epochs	800 (default), 1600 (Table 2)	
Augmentations:		
ShortSideScale	N/A	256px
RandomResizedCrop		
size	224px	
scale	[0.2, 1.0]	[0.08, 1.0]
ratio	[0.75, 1.33]	
interpolation	Bicubic	Bilinear
RandomHorizontalFlip	$p = 0.5$	$p = 0.0$
Normalize	Yes	

Table 3. Pretraining hyperparameters

We train the model using 64 (or 128 for ViT-L, ViT-H)

32GB+ GPUs (A100 or Volta 32GB). The model is trained with an ℓ_2 loss on the pixel values. We normalize the target using the mean and variance of the pixels in the patch, where the norm is computed for each color channel separately, before applying the loss. Note that we use the original 0-255 pixel values before normalizing for this loss. We use a decoder with 4 layers and 384 dimension for ViT-B, 4 layers and 512 dimension for ViT-L, and 8 layers and 512 dimension for ViT-H.

Optimized video dataloading. AI training clusters usually load data from high-latency high-throughput filesystems, where dataloading for image and video datasets is bottlenecked not by the throughput, but by the latency for each read. For videos, this problem is exacerbated by the additional time spent decoding videos, ultimately resulting in situations where video training is bottlenecked by dataloading. In OmniMAE, where the training step is lightweight, owing to 95% masking for videos, in order to see a congruent improvement in wall-clock times, we needed to optimize our video dataloading. PyTorch dataloaders read data in synchronous fashion, which means that while a sample is loaded from disk and decoded, the corresponding dataloading process blocks without doing any work while waiting. Having multiple dataloader workers can mitigate this, but there is a cap to it based on the CPU memory and cores. To mitigate this issue, we implemented asynchronous dataloaders using `asyncio`, allowing the same process to send multiple requests without blocking. This allows us to reduce the amortized data reading time to effectively zero, resulting in much faster training speeds. As Figure 5b shows, our sample replication strategy provides a training speedup over this optimized dataloading setup.

B.2. Transfer Learning

Table 4 specifies the hyperparameters for finetuning ViT-B, ViT-L and ViT-H on the image datasets we utilize – IN1K, iNat18 and P365. We report the peak test accuracy observed during training. For all three datasets we use the same settings with just different peak learning rates and total epochs.

For fine tuning on video datasets, we sample 16 frames from clips. For SSv2 and EK100 we sample 2.7 second clips. For K400 we sample 2 second clips. At test time, we sample 5 clips with 3 spatial crops and report the final test accuracy at the end of training. Table 5 specifies the hyperparameters for finetuning on SSv2 and EK100, and Table 6 on K400.

For all the datasets, we use the same finetuning hyperparameters for ViT-L and ViT-H.

B.3. Details about Ablations

For ablations, we start from a base pretraining configuration on IN1K and SSv2 that includes 1) 75% and 90% masking on IN1K and SSv2 respectively; 2) Random and Tube masking on IN1K and SSv2 respectively; 3) A common

Config	ViT-B	ViT-{L, H}
Optimizer	AdamW	
Peak learning rate		
IN1K	4e-3	2e-3
iNat18		2e-3
P365		2e-3
Total epochs		
IN1K	100	50
iNat18	300	100
P365	60	50
Warmup epochs		5
Weight decay	1e-4	5e-2
Layerwise LR decay [6, 20]	0.65	0.75
Optimizer Momentum	$\beta_1 = 0.9, \beta_2 = 0.999$	
Batch size		1024
DropPath [45]	0.1	0.2
EMA [69]		1e-4
Augmentations:		
RandomResizedCrop		
size		224px
scale		[0.08, 1.0]
ratio		[0.75, 1.33]
interpolation		Bicubic
RandomHorizontalFlip		$p = 0.5$
RandomAugment [21]		
magnitude		9
num_layers		0.5
RandomErasing [90]		$p = 0.25$
Normalize		Yes
mixup [88]		0.8
CutMix [86]		1.0
LabelSmoothing [73]		0.1

Table 4. Finetuning hyperparameters for IN1K, iNat18 and P365

4-layer 384D decoder for both datasets; 4) Peak learning rate of 3e-4.

Masking ratio. For the masking ratio ablation, we vary the amount of patches that are masked for each of the modalities. We start from the default 75% masking for each modality [40], and increase it upto 90% for images and 95% for videos. We notice that while downstream performance on images is stable across different masking ratios on the image dataset during pretraining, increasing the video masking leads to improved performance on downstream video tasks. We observe the best performance at 95% masking on videos.

Masking type. For this ablation, we vary the type of masking used in each modality. Starting from the default masking type of (Random, Tube), we experiment with Causal masking on images, and Frame, Causal or Random masking on videos. In all cases we keep the masking ratio to be 75% for

Config	ViT-B	ViT-{L, H}
Optimizer	AdamW	
Peak learning rate	1e-3	
Total epochs	40	
Batch size	512	
Warmup epochs	5	
Weight decay	5e-2	
Layerwise LR decay [6, 20]	0.75	
Optimizer Momentum	$\beta_1 = 0.9$ $\beta_2 = 0.999$	
DropPath [45]	0.1	0.2
Augmentations:		
ShortSideScale	256px	
RandomResizedCrop		
size	224px	
scale	[0.08, 1.0]	
ratio	[0.75, 1.33]	
interpolation	Bicubic	
RandomAugment [21]		
magnitude	7	
num_layers	4	
RandomErasing [90]	$p = 0.25$	
Normalize	Yes	
mixup [88]	0.8	
CutMix [86]	1.0	
LabelSmoothing [73]	0.1	

Table 5. Finetuning hyperparameters for SSv2 & EK100.

Config	ViT-B	ViT-{L, H}
Optimizer	AdamW	
Peak learning rate	1e-3	
Total epochs	175	100
Sample replication	2	
Batch size	1024	512
Warmup epochs	9	
Weight decay	5e-2	
Layerwise LR decay [6, 20]	0.75	
Optimizer Momentum	$\beta_1 = 0.9$ $\beta_2 = 0.999$	
DropPath [45]	0.1	0.2
Augmentations:		
ShortSideScale	256px	
RandomResizedCrop		
size	224px	
scale	[0.08, 1.0]	
ratio	[0.75, 1.33]	
interpolation	Bicubic	
RandomHorizontalFlip	$p = 0.5$	
RandomAugment [21]		
magnitude	9	
num_layers	2	
Normalize	Yes	
mixup [88]	0.8	
CutMix [86]	1.0	
LabelSmoothing [73]	0.1	

Table 6. Finetuning hyperparameters for K400.

images and 90% for videos.

Decoder capacity. For this ablation, we explored different decoder capacities by varying the decoder depth and embedding dimension. In particular, we test decoder depths of 2, 4 and 8 layers as well as embedding dimensions of 384 and 512.

Sample replication. We briefly note the procedure for sample replication. Let B denote the total batch size for training without any replication. To maintain the total batch size for training, when replicating a sample t times, we sample $\frac{B}{t}$ training samples from the dataset. After replication, each of the B samples is augmented and processed individually. Thus, sample replication reduces the I/O associated with reading and decoding a sample by a factor proportional to the replication factor t .

Dataset ratio. We experiment with varying the relative dataset ratio of IN1K and SSv2 such that we replicate only one dataset at a time. In addition to the default dataset ratio of 1:1 (IN1K:SSv2), we test dataset ratios of 1:2, 1:3, 2:1 and 3:1. For such dataset ratios, the samples for one of the datasets are replicated for every epoch, leading to longer training wall clock time.

Specify random variance. Due to the large number of ex-

periments and compute associated with training the models, we note the random variance across a small subset of our experiments from § 4.2. We measure the random variance of both pretraining and transfer learning. We pretrain the model with different random seeds and finetune it on ImageNet and SSv2. Across a trial of 3 such pretraining and 2 finetuning (total 6 runs), we observed a variance of 0.3% and 0.7% on ImageNet and SSv2 respectively.

B.4. Visualization details

To visualize the pixel reconstructions, we train another model without the patch mean/var normalization in the loss. This ensures the model can directly generate the pixel values that we can visualize without needing to provide it the patch’s mean/variance. For visualization, we reshape the predicted pixel values to the original image dimensions, and replace the unmasked patches with the ground truth pixel values.

C. Additional Visualizations

We present additional visualizations in Figs. 6 and 7. To match the model’s training settings, we visualize by masking

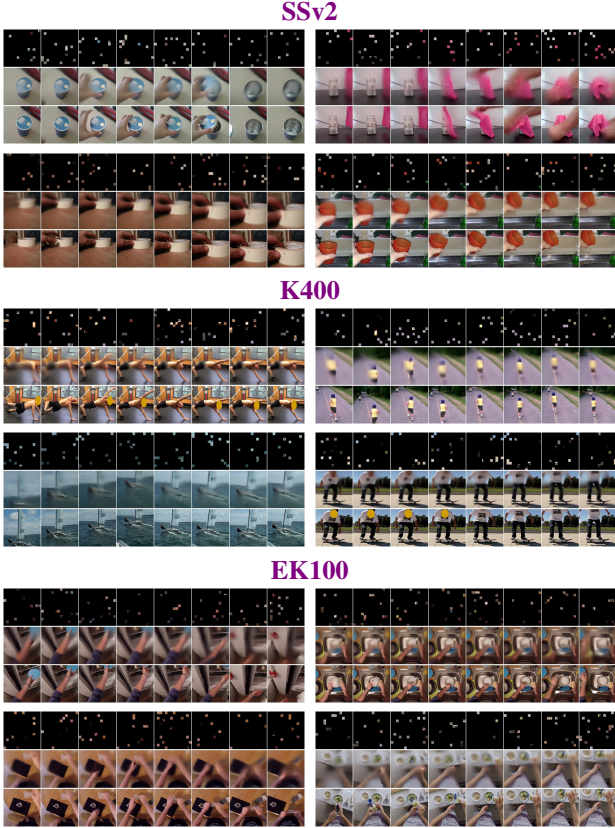


Figure 6. **Additional Reconstruction visualizations** using OmniMAE on different **video** datasets. We show the model predictions for a masking ratio of 95%.



Figure 7. **Additional Reconstruction visualizations** using OmniMAE on the **IN1K** image dataset. We show the model predictions for a masking ratio of 90%.

90% of the image patches and 95% of the video patches.

D. Additional Ablations

Effect of extra data. Since OmniMAE is trained with extra data compared to MAE, one concern is whether the gains can be attributed to joint training or simply the extra frames. To that end, we experiment with training MAE with individual frames from SSv2 instead of videos. To ensure an exact apples-to-apples comparison, we use the exact setup for OmniMAE, and simply convert each video input into individual frames, hence ensuring the exact same epochs, number of parameter updates, data, learning rates schedule etc. As we see in Table 7, the SSv2 video classification per-

Setting	Data	IN1K	SSv2
OmniMAE	IN1K + SSv2 frames	82.7	64.2
OmniMAE	IN1K + SSv2	82.8	69.0
MAE (cf. Figure 2)	IN1K	83.4	59.5

Table 7. **Effect of extra data.** We train OmniMAE with the exact set of IN1K images and SSv2 frames used during original OmniMAE pretraining with IN1K images and SSv2 videos. This follows our setup in § 4.2, where we train ViT-B for 800 epochs. While the IN1K image classification performance in both settings is comparable, the SSv2 video classification performance drops significantly by almost 5% when trained only using frames and not video clips, although it is better than just training with IN1K images. This shows that the performance gains with OmniMAE are not merely due to the additional data being used for training.

formance drops by almost 5% when trained with frames and not video clips, although it is better than training with only IN1K images. This ensures that the gains are indeed from jointly training on the two modalities, rather than simply using more data during training.

References

- [1] Relja Arandjelovic and Andrew Zisserman. Look, listen and learn. In *ICCV*, 2017.
- [2] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lucic, and Cordelia Schmid. ViViT: A video vision transformer. In *ICCV*, 2021.
- [3] Yuki Markus Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *ICLR*, 2020.
- [4] Sara Atito, Muhammad Awais, and Josef Kittler. Sit: Self-supervised vision transformer. *arXiv preprint arXiv:2104.03602*, 2021.
- [5] Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. data2vec: A general framework for self-supervised learning in speech, vision and language. In *ICML*, 2022.
- [6] Hangbo Bao, Li Dong, and Furu Wei. BEiT: Bert pre-training of image transformers. In *ICLR*, 2022.
- [7] Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. In *ICLR*, 2022.
- [8] Maxim Berman, Hervé Jégou, Andrea Vedaldi, Iasonas Kokkinos, and Matthijs Douze. Multigrain: a unified image embedding for classes and instances. *arXiv preprint arXiv:1902.05509*, 2019.
- [9] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *ICML*, 2021.
- [10] Hervé Bourlard and Yves Kamp. Auto-association by multilayer perceptrons and singular value decomposition. *Biological Cybernetics*, 1988.
- [11] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018.
- [12] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Un-supervised learning of visual features by contrasting cluster assignments. In *NeurIPS*, 2020.
- [13] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021.
- [14] João Carreira and Andrew Zisserman. Quo vadis, action recognition? A new model and the kinetics dataset. In *CVPR*, 2017.
- [15] Lluís Castrejon, Yusuf Aytar, Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Learning aligned cross-modal representations from weakly aligned data. In *CVPR*, 2016.
- [16] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *ICML*, 2020.
- [17] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.
- [18] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *CVPR*, 2021.
- [19] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *ICCV*, 2021.
- [20] Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. Electra: Pre-training text encoders as discriminators rather than generators. *ICLR*, 2020.
- [21] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *CVPR*, 2020.
- [22] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Evangelos Kazakos, Jian Ma, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Rescaling egocentric vision. *IJCV*, 2021.
- [23] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2018.
- [24] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.
- [25] Alaaeldin El-Nouby, Gautier Izacard, Hugo Touvron, Ivan Laptev, Hervé Jégou, and Edouard Grave. Are large-scale datasets necessary for self-supervised pre-training? *arXiv preprint arXiv:2112.10740*, 2021.
- [26] Aleksandr Ermolov, Aliaksandr Siarohin, Enver Sangineto, and Nicu Sebe. Whitening for self-supervised representation learning. In *ICML*, 2021.
- [27] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. In *ICCV*, 2021.
- [28] Christoph Feichtenhofer, Haoqi Fan, Yanghao Li, and Kaiming He. Masked autoencoders as spatiotemporal learners. *arXiv preprint arXiv:2205.09113*, 2022.
- [29] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *CVPR*, 2016.

- [30] Patrick Gallinari, Yann LeCun, Sylvie Thiria, and F Fogelman Soulie. Mémoires associatives distribuées: une comparaison (distributed associative memories: a comparison). In *COGNITIVA*. 1987.
- [31] Rohit Girdhar, João Carreira, Carl Doersch, and Andrew Zisserman. Video action transformer network. In *CVPR*, 2019.
- [32] Rohit Girdhar and Kristen Grauman. Anticipative Video Transformer. In *ICCV*, 2021.
- [33] Rohit Girdhar, Mannat Singh, Nikhila Ravi, Laurens van der Maaten, Armand Joulin, and Ishan Misra. Omnivore: A Single Model for Many Visual Modalities. In *CVPR*, 2022.
- [34] Yunchao Gong, Liwei Wang, Micah Hodosh, Julia Hockenmaier, and Svetlana Lazebnik. Improving image-sentence embeddings using large weakly annotated photo collections. In *ECCV*, 2014.
- [35] Priya Goyal, Quentin Duval, Isaac Seessel, Mathilde Caron, Mannat Singh, Ishan Misra, Levent Sagun, Armand Joulin, and Piotr Bojanowski. Vision models are more robust and fair when pretrained on uncurated images without supervision. *arXiv preprint arXiv:2202.08360*, 2022.
- [36] Priya Goyal, Dhruv Mahajan, Abhinav Gupta, and Ishan Misra. Scaling and benchmarking self-supervised visual representation learning. In *ICCV*, 2019.
- [37] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, Florian Hoppe, Christian Thureau, Ingo Bax, and Roland Memisevic. The “something something” video database for learning and evaluating visual common sense. In *ICCV*, 2017.
- [38] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *NeurIPS*, 2020.
- [39] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, 2006.
- [40] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022.
- [41] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.
- [42] Geoffrey E Hinton and Richard Zemel. Autoencoders, minimum description length and helmholtz free energy. In *NeurIPS*, 1993.
- [43] Elad Hoffer, Tal Ben-Nun, Itay Hubara, Niv Giladi, Torsten Hoefer, and Daniel Soudry. Augment your batch: Improving generalization through instance repetition. In *CVPR*, 2020.
- [44] Grant Van Horn, Oisín Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *CVPR*, 2018.
- [45] Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q Weinberger. Deep networks with stochastic depth. In *ECCV*, 2016.
- [46] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.
- [47] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015.
- [48] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, AMustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.
- [49] Mark A Kramer. Nonlinear principal component analysis using autoassociative neural networks. *AICHE*, 1991.
- [50] Yann LeCun and Françoise Fogelman-Soulié. Modèles connexionnistes de l’apprentissage. *Intellectica*, 1987.
- [51] Junnan Li, Pan Zhou, Caiming Xiong, and Steven C.H. Hoi. Prototypical contrastive learning of unsupervised representations. In *ICLR*, 2021.
- [52] Yanghao Li, Chao-Yuan Wu, Haoqi Fan, Karttikeya Mangalam, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. Improved multiscale vision transformers for classification and detection. In *CVPR*, 2022.
- [53] Valerii Likhoshesterov, Anurag Arnab, Krzysztof Choromanski, Mario Lucic, Yi Tay, Adrian Weller, and Mostafa Dehghani. Polyvit: Co-training vision transformers on images, videos and audio. *arXiv preprint arXiv:2111.12993*, 2021.
- [54] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, Furu Wei, and Baining Guo. Swin transformer v2: Scaling up capacity and resolution. In *CVPR*, 2022.
- [55] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021.
- [56] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. *arXiv preprint arXiv:2106.13230*, 2021.

- [57] Ze Liu, Zheng Zhang, Yue Cao, Han Hu, and Xin Tong. Group-free 3d object detection via transformers. In *ICCV*, 2021.
- [58] Jiasen Lu, Vedanuj Goswami, Marcus Rohrbach, Devi Parikh, and Stefan Lee. 12-in-1: Multi-task vision and language representation learning. In *CVPR*, 2020.
- [59] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *CVPR*, 2020.
- [60] Ishan Misra, Rohit Girdhar, and Armand Joulin. An End-to-End Transformer Model for 3D Object Detection. In *ICCV*, 2021.
- [61] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *CVPR*, 2020.
- [62] Pedro Morgado, Ishan Misra, and Nuno Vasconcelos. Robust audio-visual instance discrimination. In *CVPR*, 2021.
- [63] Pedro Morgado, Nuno Vasconcelos, and Ishan Misra. Audio-visual instance discrimination with cross-modal agreement. In *CVPR*, 2021.
- [64] B. A. Olshausen and D. J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 1996.
- [65] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. In *NeurIPS*, 2018.
- [66] Andrew Owens and Alexei A Efros. Audio-visual scene analysis with self-supervised multisensory features. In *ECCV*, 2018.
- [67] Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer. In *ICML*, 2018.
- [68] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016.
- [69] Boris T Polyak and Anatoli B Juditsky. Acceleration of stochastic approximation by averaging. *SIAM journal on control and optimization*, 1992.
- [70] 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.
- [71] Ruslan Salakhutdinov and Geoffrey Hinton. Deep Boltzmann machines. In *AISTATS*, 2009.
- [72] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *NeurIPS*, 2014.
- [73] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, 2016.
- [74] Hao Tan, Jie Lei, Thomas Wolf, and Mohit Bansal. VIMPAC: Video pre-training via masked token prediction and contrastive learning. *arXiv preprint arXiv:2106.11250*, 2021.
- [75] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *arXiv preprint arXiv:1906.05849*, 2019.
- [76] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *arXiv preprint arXiv:2203.12602*, 2022.
- [77] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, 2021.
- [78] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.
- [79] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol. Extracting and composing robust features with denoising autoencoders. In *ICML*, 2008.
- [80] Rui Wang, Dongdong Chen, Zuxuan Wu, Yinpeng Chen, Xiyang Dai, Mengchen Liu, Yu-Gang Jiang, Luowei Zhou, and Lu Yuan. BEVT: Bert pretraining of video transformers. In *CVPR*, 2022.
- [81] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *ICCV*, 2021.
- [82] Chen Wei, Haoqi Fan, Saining Xie, Chao-Yuan Wu, Alan Yuille, and Christoph Feichtenhofer. Masked feature prediction for self-supervised visual pre-training. In *CVPR*, 2022.
- [83] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, 2018.
- [84] Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. Simmim: A simple framework for masked image modeling. In *CVPR*, 2022.
- [85] Xueting Yan, Ishan Misra, Abhinav Gupta, Deepti Ghadiyaram, and Dhruv Mahajan. ClusterFit: Improving Generalization of Visual Representations. In *CVPR*, 2020.

- [86] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *ICCV*, 2019.
- [87] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *ICML*, 2021.
- [88] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *ICLR*, 2018.
- [89] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip Torr, and Vladlen Koltun. Point transformer. In *ICCV*, 2021.
- [90] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *AAAI*, 2020.
- [91] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. In *NeurIPS*, 2014.
- [92] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. In *ICLR*, 2022.