Supplementary Material
SIGMA: Sinkhorn-Guided Masked Video Modeling

Mohammadreza Salehi*, Michael Dorkenwald*, Fida Mohammad Thoker*‡,
Efstratios Gavves, Cees G. M. Snoek, Yuki M. Asano†

University of Amsterdam

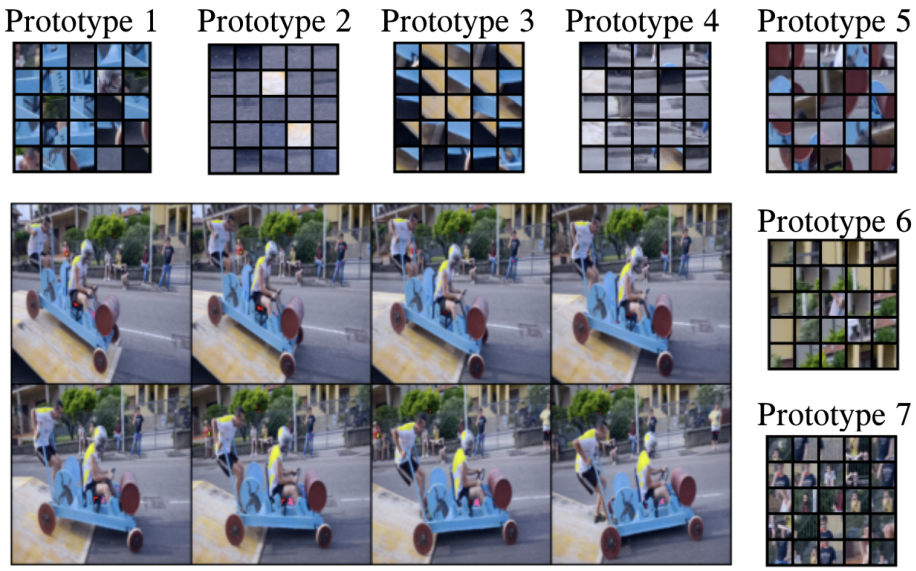

Fig. 1. Visualization of prototypes. We visualize the 25 space-time tubes with the highest similarity to a particular prototype inside a video. For simplicity, we visualize the first patch inside the space-time tube. We observe that different prototypes attend to particular semantic parts of the video, as prototype 1 corresponds to the blue parts of the car.

1 Visualization of Prototypes

In this section, we analyze the prototypes learned by our method. For that, we use SIGMA pretrained on Kinetics and using DINO as a projection network. We visualize the 25 space-time tubes inside videos from DAVIS that have the highest similarity with a given prototype in Fig. 1 and Fig. 2. For simplcity, we visualize the first patch in time of the space-time tube. We observe in Fig. 1 that the patches for a particular prototype are semantically similar, as prototype 6 captures the tree parts in the background, while prototype 7 captures faces/persons in the video. Similarly in Fig. 2 where a different set of prototypes are visualized which correspond to the white background, one to the person in white, and one to the clothing of the runner.

* equal contribution. † now at University of Technology Nuremberg. ‡ now at KAUST.

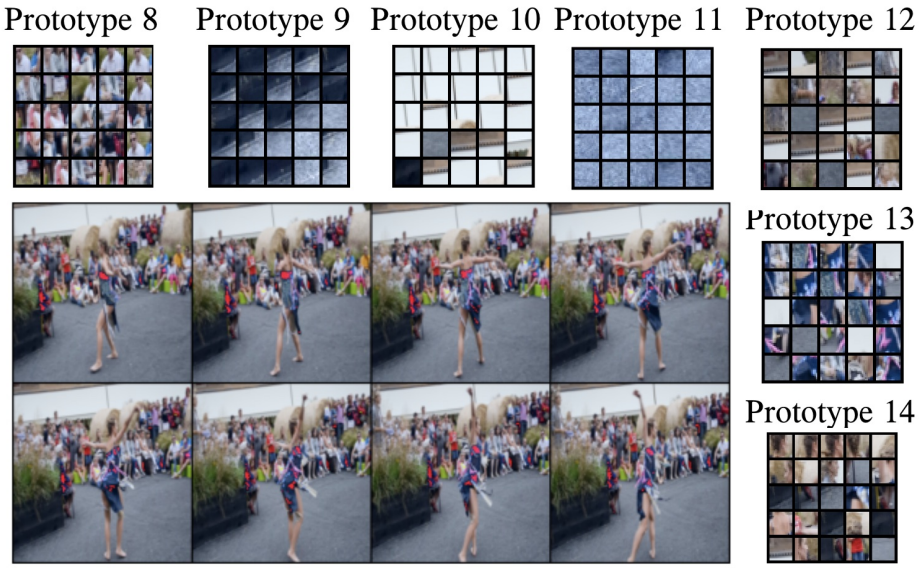

Fig. 2. Visualization of prototypes (2). We visualize the 25 space-time tubes with the highest similarity to a particular prototype inside a video. For simplicity, we visualize the first patch inside the space-time tube. We observe that different prototypes attend to particular semantic parts of the video, for example, prototype 1 corresponds to the person(s) in white.

2 Dataset details

In this section, we list the details of the datasets used in our experiments.

Something-Something V2 (SSv2) [13] contains 220K videos with 174 action classes and is considered motion-heavy because of its focus on motion and directional aspects inherent to the actions. Example classes are: Pushing something from left to right, Pulling something from right to left, Putting something down etc.

Kinetics-400 (K400) [16] is a dataset for recognizing actions in videos, which comprises realistic action videos gathered from YouTube. The dataset contains 306,245 short-trimmed videos, covering 400 action categories, making it one of the largest and most extensively used datasets for evaluating state-of-the-art video action recognition models. Some example classes are: Bungee jumping, Cutting pineapple, Doing aerobics.

DAVIS [23] includes 150 videos, with 60 allocated for training, 30 for validation, and 60 for testing. All the validation videos for this dataset include full-frame annotations as opposed to the test set. Therefore, we use the validation split to test our object segmentation performances.

YTVOS [38] is a larger video object segmentation dataset compared to DAVIS, comprises 4,453 videos, each annotated under one of 65 object categories. Like DAVIS, YTVOS provides ground truth masks only for the first frames in both the test and validation sets. Therefore, a random subset of 20% of the training set is used for evaluations.

Table 1. Benchmark Details for the downstream evaluation setup, experiments, and datasets we use. For that, we use the SEVERE benchmark [32].

Evaluation Setup	Experiment	Dataset	Task	#Classes	#Finetuning	#Testing	Eval Metric
Domain Shift	SSv2	Something-Something [13]	Action Recognition	174	168,913	24,777	Top-1 Accuracy
	Gym99	FineGym [25]	Action Recognition	99	20,484	8,521	Top-1 Accuracy
Sample Efficiency	UCF (10^3)	UCF 101 [27]	Action Recognition	101	1,000	3,783	Top-1 Accuracy
	Gym (10^3)	FineGym [25]	Action Recognition	99	1,000	8,521	Top-1 Accuracy
Action Granularity	FX-S1	FineGym [25]	Action Recognition	11	1,882	777	Mean Class Acc
	UB-S1	FineGym [25]	Action Recognition	15	3,511	1,471	Mean Class Acc
Task Shift	UCF-RC	UCFRep [41]	Repetition Counting	-	421	105	Mean Error
	Charades	Charades [26]	Multi-label Recognition	157	7,985	1,863	mAP

Also, the meta-information provided by datasets is used to ensure that objects within the same category have consistent class IDs.

UCF101 [27] The dataset comprises of 13,320 video clips that are divided into 101 categories. These 101 categories are further grouped into 5 types - Body motion, Human-human interactions, Human-object interactions, Playing musical instruments, and Sports. The combined duration of these video clips is over 27 hours. All the videos were sourced from YouTube and have a fixed frame rate of 25 FPS, with a resolution of 320×240. Some example classes are: Handstand Pushups, Billiards, Band Marching.

HMDB-51 [19] is a dataset designed for action recognition, which has been gathered from multiple sources including movies and public databases such as the Prelinger archive, YouTube, and Google videos. The dataset contains 6,766 clips that have been categorized into 51 different action categories, each of which contains at least 100 clips. Some example classes are: Ride Horse, Shoot Gun, Turn.

ImageNet1K [5] is often used to train deep learning models for computer vision tasks. The ImageNet1K dataset consists of 1000 object classes, and it includes 1,281,167 training images, 50,000 validation images.

CIFAR-100 [18] is composed of 32x32 color images and includes 100 classes that are divided into 20 superclasses. Each class has 600 images.

SEVERE Benchmark [32] encompasses eight different experimental settings from 4 different datasets. The setup for each subset in SEVERE-Benchmkark is listed in Table 1.

3 Evaluation details

Linear We follow the linear evaluation setup from MME [28] and the used setup in Table 2.

Full finetuning We follow the default setup from [33] for full finetuning and the specifics are listed in Table 3.

Unsupervised segmentation We obtain video clips of size $[T, 3, 224, 224]$ from two datasets: DAVIS [23] and YTVOS [38]. For DAVIS [23], we use clips of length $T = 16$, and for YTVOS [38], the length is $T = 4$. Each clip is paired with its corresponding ground truth and fed into the model to extract final dense features of size $[\frac{T}{2}, d, 14, 14]$, where d represents the dimension of the encoder.

Table 2. Linear-Evaluation setting.

config	SSv2	K400	IN-1K	Others
optimizer	AdamW [21]			
base learning rate	1.e-3			
weight decay	0.05			
optimizer momentum	$\beta_1, \beta_2 = 0.9, 0.999$			
layer-wise lr decay [1]	0.75			
batch size	128			
learning rate schedule	cosine decay			
training epochs	30	40	30	100
flip augmentation	*no*	*yes*	*yes*	*yes*

Table 3. Full finetuning evaluation setup.

config	SSv2	K400	SEVERE
optimizer	AdamW		
base learning rate	1.0e-3		
weight decay	0.05		
optimizer momentum	$\beta_1, \beta_2 = 0.9, 0.999$		
layer-wise lr decay [1]	0.75		
batch size	32	16	16
learning rate schedule	cosine decay		
warmup epochs	5		
training epochs	50	75	100
flip augmentation	*no*	*yes*	*yes*
RandAug [4]	(9,0.5)		
label smoothing [30]	0.1		
mixup [40]	0.8		
cutmix [39]	1.0		
drop path	0.1		

Next, we resize the ground truth and feature maps to size 28 using nearest neighbor and linear interpolation, respectively. We then cluster the feature maps with different granularity values of K. We set K to the ground truth object counts for clustering, and three times higher than the average object counts per clip, which is 6, for overclustering evaluations. This results in clustering and overclustering maps. Finally, we repeat every cluster map two times and group the clusters into ground-truth classes by matching them either by pixel-wise precision or Hungarian matching on merged cluster maps, similar to [24].

Severe benchmark

For action recognition tasks in SEVERE benchmark (**GYM99**, **UCF**, **FX-S1** and **UBS1**) we follow the finetuning setup from Table 3.

For the Repetition Counting Task (denoted as **UCF-RC**), we adhere to the implementation details specified in the original work [41] on repetition counting. From the annotated video dataset, we construct 2 million sequences, each consisting of 32 frames with a spatial resolution of 224×224 pixels. These sequences serve as the input to our

model. The training process spans over 100 epochs with a batch size of 32, utilizing the Adam [17] optimizer. The learning rate is set to 5×10^{-5}. For the evaluation phase, we follow [41] to report the mean counting error.

For the Multi-label classification on **Charades**, we employ [10] to incorporate a per-class sigmoid output layer for multi-class prediction. In the training phase, we sample 16 frames with a stride of 8 from each video. The frames are resized to a spatial resolution of 224×224 pixels. We apply several data augmentation techniques, including random short-side scaling, random spatial cropping, and horizontal flipping. The model is trained over 57 epochs, utilizing a batch size of 16 and a learning rate of 1×10^{-4}. For the evaluation phase, spatiotemporal max-pooling is executed over 10 distinct clips from each video to aggregate the predictions. The performance is quantified using the mean Average Precision (mAP) across all classes.

4 Extended comparison for full finetuning on SSv2 and K400

We provide an extended version of comparison with state-of-the-art for full finetuning in Tab. 4 and Tab. 5. As is shown, SIGMA improves VideoMAE [34] baseline by 0.8% and 1.3% on SSv2(Tab. 5) when pretrained on K400 or SSv2 using an MLP projection network. Using a pretrained DINO [3] model as the projection network results in even larger improvement, reaching 1% and 1.8% across different pretrainings, getting state-of-the-art results across the models trained for the same number of epochs. For K400 we observe similar results. As shown in Tab. 4, SIGMA considerably improves VideoMAE [34] baseline and sets a new state-of-the-art for 800 training epochs.

MVD [36] achieves good performance while using a computationally expensive approach requiring longer pretraining. First, a video model, following VideoMAE [34], is trained on K400 for 1600 epochs. Then, an image model, following MAE [14], is trained on ImageNet [5] for 1600 epochs. Finally, the VideoMAE and MAE models are kept frozen and serve as the teachers for the main video model which is trained via distillation for 400 epochs. This complex and multi-step training process makes it hard to provide a one-to-one comparison between this and other methods. MGM [6] and MME [29] are two other models that have been trained with a higher number of epochs, yet they still perform comparably to our model which was trained with half the number of epochs, based on the K400 benchmark.

Table 4. Benchmark I: Comparison for full finetuning on Kinetics 400 (K400). We compare against all previous methods for pretraining the ViT-Base backbone on K400 and subsequently, fully finetuning the backbone with the K400 labels. M. Guid. denotes motion guidance such as optical flow used e.g. reconstructing targets or masking.

	Method	M. Guid.	Backbone	Epochs	Extra data	Frames	Params	Top-1
	supervised							
	SlowFast [9]	ResNet101	–	–	–	16+64	60	79.8
	MViTv1 [8]	–	MViTv1-B	–	–	32	37	80.2
	TimeSformer [2]	–	ViT-B	–	IN-21K	96	430	80.7
	VideoSwin [20]	–	Swin-L	–	IN-21K	32	197	83.1
	self-supervised							
K400 pretraining	VideoMAE [33]	×	ViT-S	1600	–	16	87	79.0
K400 pretraining	MVD [37]	×	ViT-S	1600 + 400	IN-1K	16	22	80.6
K400 pretraining	SIGMA-DINO (ours)	×	ViT-S	800	IN-1K	16	87	79.4
K400 pretraining	VIMPAC [31]	×	ViT-L	100	HowTo100M+DALL-E	10	307	77.4
K400 pretraining	VideoMAE [33]	×	ViT-B	800	–	16	87	80.0
K400 pretraining	VideoMAE [33]	×	ViT-B	1600	–	16	87	80.9
K400 pretraining	OmniMAE [12]	×	ViT-B	800	IN-1K	16	87	80.8
K400 pretraining	ST-MAE [11]	×	ViT-B	1600	–	16	87	81.3
K400 pretraining	MME [28]	✓	ViT-B	1600	–	16	87	81.8
K400 pretraining	MVD [37]	×	ViT-B	1600 + 400	IN-1K	16	87	82.7
K400 pretraining	CMAE-V [22]	×	ViT-B	800	–	16	87	80.2
K400 pretraining	CMAE-V [22]	×	ViT-B	1600	–	16	87	80.9
K400 pretraining	MGM [7]	✓	ViT-B	800	–	16	87	80.8
K400 pretraining	MGM [7]	✓	ViT-B	1600	–	16	87	81.7
K400 pretraining	MGMAE [15]	✓	ViT-B	800	–	16	87	81.2
K400 pretraining	SIGMA-MLP (ours)	×	ViT-B	800	–	16	90	80.2
K400 pretraining	SIGMA-DINO (ours)	×	ViT-B	800	IN-1K	16	87	**81.6**

Table 5. Benchmark I: Comparison for full finetuning on Something-Something V2 (SSv2). The top part compromises supervised methods while the remaining methods are pretrained in a self-supervised manner. The middle section evaluates models trained on Kinetics 400 (K400) data for pretraining whereas the bottom part mainly uses SSv2 data. We compare against all previous methods pretrained on the ViT-Base backbone. M. Guid. denotes motion guidance such as optical flow used e.g. reconstructing targets or masking.

	Method	M. Guid.	Backbone	Epochs	Extra data	Frames	Params	Top-1
	supervised baselines							
	SlowFast [9]	×	ResNet101	–	K400	8+32	53	63.1
	MViTv1 [8]	×	MViTv1-B	–	K400	64	37	67.7
	TimeSformer [2]	×	ViT-B	–	IN-21K	8	121	59.5
	VideoSwin [20]	×	Swin-B	–	IN-21K	32	88	69.6
	self-supervised							
	MVD [37]	×	ViT-S	1600 + 400	IN-1K+K400	16	22	70.7
	SIGMA-DINO (ours)	×	ViT-S	800	IN-1K+K400	16	25	68.7
K400 pretraining	BEVT [35]	×	Swin-B	150	IN-1K+K400	32	88	67.6
	BEVT [35]	×	Swin-B	150	IN-1K+K400+DALL-E	32	88	70.6
	VIMPAC [31]	×	ViT-L	100	HowTo100M+DALL-E	10	307	68.1
	OmniMAE [12]	×	ViT-B	800	IN-1K+K400	16	86	69.0
	VideoMAE [33]	×	ViT-B	800	K400	16	87	68.5
	VideoMAE [33]	×	ViT-B	2400	K400	16	87	69.7
	MME [28]	✓	ViT-B	800	K400	16	87	70.5
	MVD [37]	×	ViT-B	1600 + 400	IN-1K+K400	16	87	72.5
	SIGMA-MLP (ours)	×	ViT-B	800	K400	16	90	69.8
	SIGMA-DINO (ours)	×	ViT-B	800	IN-1K+K400	16	87	**71.1**
	VideoMAE [33]	×	ViT-S	2400	–	16	22	66.8
	SIGMA-DINO (ours)	×	ViT-S	2400	IN-1K	16	22	**68.6**
SSv2 pretraining	OmniMAE [12]	×	ViT-B	800	IN-1K	16	86	69.5
	VideoMAE [33]	×	ViT-B	800	–	16	87	69.6
	VideoMAE [33]	×	ViT-B	2400	–	16	87	70.8
	CMAE-V [22]	×	ViT-B	800	–	16	87	69.7
	CMAE-V [22]	×	ViT-B	1600	–	16	87	70.5
	SIGMA-MLP (ours)	×	ViT-B	800	–	16	87	70.4
	SIGMA-DINO (ours)	×	ViT-B	800	IN-1K	16	87	**70.9**
	MME [28]	✓	ViT-B	800	–	16	87	70.0
	MGM [7]	✓	ViT-B	800	–	16	87	70.6
	MGM [7]	✓	ViT-B	1200	–	16	87	71.6
	MGM [7]	✓	ViT-B	1600	–	16	87	71.8
	MGMAE [15]	✓	ViT-B	800	–	16	87	71.0
	SIGMA-MLP (ours)	✓	ViT-B	800	–	16	87	**71.2**
	SIGMA-DINO (ours)	✓	ViT-B	800	IN-1K	16	87	**71.2**

References

1. Bao, H., Dong, L., Piao, S., Wei, F.: Beit: Bert pre-training of image transformers (2022) 4
2. Bertasius, G., Wang, H., Torresani, L.: Is space-time attention all you need for video understanding? ArXiv **abs/2102.05095** (2021) 6, 7
3. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9650–9660 (2021) 5
4. Cubuk, E.D., Zoph, B., Shlens, J., Le, Q.V.: Randaugment: Practical automated data augmentation with a reduced search space (2019) 4
5. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. pp. 248–255 (2009) 3, 5
6. Fan, D., Wang, J., Liao, S., Zhu, Y., Bhat, V., Santos-Villalobos, H., MV, R., Li, X.: Motion-guided masking for spatiotemporal representation learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5619–5629 (2023) 5
7. Fan, D., Wang, J., Liao, S., Zhu, Y., Bhat, V., Santos-Villalobos, H.J., MV, R., Li, X.: Motion-guided masking for spatiotemporal representation learning. In: ICCV. pp. 5596–5606. IEEE (2023) 6, 7
8. Fan, H., Xiong, B., Mangalam, K., Li, Y., Yan, Z., Malik, J., Feichtenhofer, C.: Multiscale vision transformers. In: ICCV. pp. 6804–6815. IEEE (2021) 6, 7
9. Feichtenhofer, C., Fan, H., Malik, J., He, K.: Slowfast networks for video recognition. In: ICCV (2018) 6, 7
10. Feichtenhofer, C., Fan, H., Xiong, B., Girshick, R., He, K.: A large-scale study on unsupervised spatiotemporal representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2021) 5
11. Feichtenhofer, C., Li, Y., He, K., et al.: Masked autoencoders as spatiotemporal learners. Advances in neural information processing systems **35**, 35946–35958 (2022) 6
12. Girdhar, R., El-Nouby, A., Singh, M., Alwala, K.V., Joulin, A., Misra, I.: Omnimae: Single model masked pretraining on images and videos (2023) 6, 7
13. Goyal, R., Ebrahimi Kahou, S., Michalski, V., Materzynska, J., Westphal, S., Kim, H., Haenel, V., Fruend, I., Yianilos, P., Mueller-Freitag, M., et al.: The" something something" video database for learning and evaluating visual common sense. In: Proceedings of the IEEE international conference on computer vision. pp. 5842–5850 (2017) 2, 3
14. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 16000–16009 (2022) 5
15. Huang, B., Zhao, Z., Zhang, G., Qiao, Y., Wang, L.: MGMAE: motion guided masking for video masked autoencoding. In: ICCV. pp. 13447–13458. IEEE (2023) 6, 7
16. Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., et al.: The kinetics human action video dataset. arXiv preprint arXiv:1705.06950 (2017) 2
17. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization (2017) 5
18. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009) 3
19. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: Hmdb: a large video database for human motion recognition. In: 2011 International conference on computer vision. pp. 2556–2563. IEEE (2011) 3
20. Liu, Z., Ning, J., Cao, Y., Wei, Y., Zhang, Z., Lin, S., Hu, H.: Video swin transformer. In: CVPR. pp. 3192–3201. IEEE (2022) 6, 7

21. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization (2019) 4
22. Lu, C., Jin, X., Huang, Z., Hou, Q., Cheng, M., Feng, J.: CMAE-V: contrastive masked autoencoders for video action recognition. CoRR **abs/2301.06018** (2023) 6, 7
23. Pont-Tuset, J., Perazzi, F., Caelles, S., Arbeláez, P., Sorkine-Hornung, A., Van Gool, L.: The 2017 davis challenge on video object segmentation. arXiv preprint arXiv:1704.00675 (2017) 2, 3
24. Salehi, M., Gavves, E., Snoek, C.G., Asano, Y.M.: Time does tell: Self-supervised time-tuning of dense image representations. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 16536–16547 (2023) 4
25. Shao, D., Zhao, Y., Dai, B., Lin, D.: Finegym: A hierarchical video dataset for fine-grained action understanding. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2616–2625 (2020) 3
26. Sigurdsson, G.A., Varol, G., Wang, X., Farhadi, A., Laptev, I., Gupta, A.: Hollywood in homes: Crowdsourcing data collection for activity understanding. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. pp. 510–526. Springer (2016) 3
27. Soomro, K., Zamir, A.R., Shah, M.: Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402 (2012) 3
28. Sun, X., Chen, P., Chen, L., Li, C., Li, T.H., Tan, M., Gan, C.: Masked motion encoding for self-supervised video representation learning. In: CVPR. pp. 2235–2245. IEEE (2023) 3, 6, 7
29. Sun, X., Chen, P., Chen, L., Li, T.H., Tan, M., Gan, C.: M3 video: Masked motion modeling for self-supervised video representation learning. arXiv preprint arXiv:2210.06096 (2022) 5
30. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision (2015) 4
31. Tan, H., Lei, J., Wolf, T., Bansal, M.: VIMPAC: video pre-training via masked token prediction and contrastive learning. CoRR **abs/2106.11250** (2021) 6, 7
32. Thoker, F.M., Doughty, H., Bagad, P., Snoek, C.G.M.: How severe is benchmark-sensitivity in video self-supervised learning? In: European Conference on Computer Vision (ECCV) (2022) 3
33. Tong, Z., Song, Y., Wang, J., Wang, L.: Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In: NeurIPS (2022) 3, 6, 7
34. Tong, Z., Song, Y., Wang, J., Wang, L.: Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. Advances in neural information processing systems **35**, 10078–10093 (2022) 5
35. Wang, R., Chen, D., Wu, Z., Chen, Y., Dai, X., Liu, M., Jiang, Y.G., Zhou, L., Yuan, L.: Bevt: Bert pretraining of video transformers. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 14733–14743 (2022) 7
36. Wang, R., Chen, D., Wu, Z., Chen, Y., Dai, X., Liu, M., Yuan, L., Jiang, Y.G.: Masked video distillation: Rethinking masked feature modeling for self-supervised video representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6312–6322 (2023) 5
37. Wang, R., Chen, D., Wu, Z., Chen, Y., Dai, X., Liu, M., Yuan, L., Jiang, Y.: Masked video distillation: Rethinking masked feature modeling for self-supervised video representation learning. In: CVPR. pp. 6312–6322. IEEE (2023) 6, 7
38. Xu, N., Yang, L., Fan, Y., Yue, D., Liang, Y., Yang, J., Huang, T.: Youtube-vos: A large-scale video object segmentation benchmark. arXiv preprint arXiv:1809.03327 (2018) 2, 3
39. Yun, S., Han, D., Oh, S.J., Chun, S., Choe, J., Yoo, Y.: Cutmix: Regularization strategy to train strong classifiers with localizable features (2019) 4
40. Zhang, H., Cisse, M., Dauphin, Y.N., Lopez-Paz, D.: mixup: Beyond empirical risk minimization (2018) 4

41. Zhang, H., Xu, X., Han, G., He, S.: Context-aware and scale-insensitive temporal repetition counting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020) 3, 4, 5