When the Future Becomes the Past: Taming Temporal Correspondence for Self-supervised Video Representation Learning

Supplementary Material

Contents

A. Supplementary Explanation of Method

A.1. Differences with Previous Methods

The success of Masked Image Modeling (MIM) has inspired its extension to the video domain as Masked Video Modeling (MVM), which has demonstrated impressive performance for multiple video tasks.

Given a video $\boldsymbol{V} = \{\boldsymbol{v}_j \in \mathbb{R}^{H\times W\times C}\}_{j=1}^{T}$, early MVM methods [8–10, 25, 27–29] aim to train the video encoder f through dense sampling from videos and forcing the model to recover the masked pixels. Specifically, as shown in Fig. 1a, for a sampled video clip $\boldsymbol{V}' \subset \boldsymbol{V}$, $\boldsymbol{V}'$ is divided into several tubes, which will be masked with learnable `[MASK]` tokens at a large probability. Next, the masked clip is passed through an encoder-decoder structure to predict the original clip $\widehat{\boldsymbol{V}'}$ by recovering the masked pixels. Finally, the mean square error (MSE) loss is utilized to minimize the difference between $\widehat{\boldsymbol{V}'}$ and $\boldsymbol{V}'$:

$$\mathcal{L}^{v}_{MSE} = \frac{1}{|\mathcal{M}|}\sum_{i\in\mathcal{M}} \|(\boldsymbol{V}')^i - (\widehat{\boldsymbol{V}'})^i\|_2^2, \tag{1}$$

where $\mathcal{M}$ is the set of `[MASK]` tokens.

Recent MVM methods [7, 11, 14, 30] have shifted to a random sampling strategy to reduce computational cost in the early efforts. Typically, as shown in Fig. 1b, these methods first sample a current frame $\boldsymbol{v}_c \in \boldsymbol{V}$ with several masked patches, then sample an unmasked past frame $\boldsymbol{v}_p \in \boldsymbol{V}$. The masked current frame $\boldsymbol{v}_c$ are restored to $\widehat{\boldsymbol{v}}_c$ using $\boldsymbol{v}_p$ as prior information based on a conditional decoder. Similarly, the mean square error (MSE) loss is used to minimize the difference between $\widehat{\boldsymbol{v}}_c$ and $\boldsymbol{v}_c$:

$$\mathcal{L}^{f}_{MSE} = \frac{1}{|\mathcal{M}|}\sum_{i\in\mathcal{M}} \|(\boldsymbol{v}_c)^i - (\widehat{\boldsymbol{v}}_c)^i\|_2^2. \tag{2}$$

As discussed in Sec.1, the random sampling strategy introduces uncertainty in reconstruction since one single conditional frame can lead to multiple potential predictions. Besides, previous MVM methods primarily restore the masked regions at the pixel level, making the model retain excessive low-level information. To address these issues, as shown in Fig. 1c, we propose a framework named T-CoRe with two key properties: **1)** a sandwich sampling strategy to establish temporal correspondence from auxiliary frames, thereby reducing the reconstruction uncertainty; **2)** an auxiliary branch on top of a self-distillation structure to reconstruct the masked patches in the latent space, facilitating the capture high-level semantic representations. Following the detailed illustration in Sec.3, we employ CE loss for representation reconstruction and MSE loss for temporal squeezing:

$$\begin{aligned}
\mathcal{L}_{pt} &= -\frac{1}{|\mathcal{M}(\boldsymbol{v}_c)|}\sum_{i\in\mathcal{M}(\boldsymbol{v}_c)} (\hat{\boldsymbol{p}}_c)^i \log(\boldsymbol{p}_c^p)^i, \\
\mathcal{L}_{ft} &= -\frac{1}{|\mathcal{M}(\boldsymbol{v}_c)|}\sum_{i\in\mathcal{M}(\boldsymbol{v}_c)} (\hat{\boldsymbol{p}}_c)^i \log\big(\boldsymbol{p}_c^f\big)^i,
\end{aligned} \tag{3}$$

$$\mathcal{L}_{pf} = \frac{1}{|\mathcal{M}(\boldsymbol{v}_c)|}\sum_{i\in\mathcal{M}(\boldsymbol{v}_c)} \|(\boldsymbol{p}_c^p)^i - (\boldsymbol{p}_c^f)^i\|_2^2, \tag{4}$$

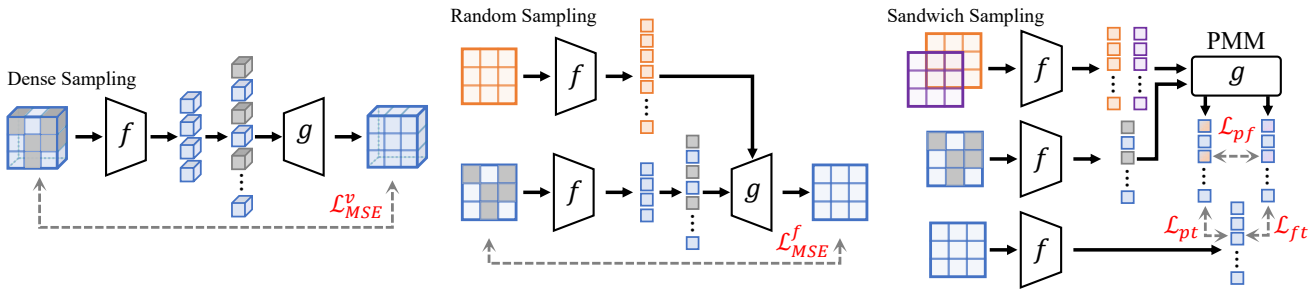
where $\boldsymbol{p}_c^p$, $\boldsymbol{p}_c^f$ are the restored representations of the current frame guided by the past and future frame, respectively, and $\hat{\boldsymbol{p}}_c$ is the representation from the teacher branch.

B. Detailed Description of Experiments

B.1. Training Datasets

Kinetics-400 [16] is a large-scale video dataset with 400 categories of daily actions, widely used for tasks like action recognition and video understanding. It includes 239,789 available training videos, each about 10 seconds long. We extract frames at $FPS = 2$ to create our training set.

ImageNet-1k [5] is a widely used large-scale static image dataset containing 1,000 categories, covering a broad range of real-world objects. The training subset consists of 1.28 million images in the training subset, which we use to pre-train our framework.



(a) Structure of early MVM methods. (b) Structure of recent MVM methods. (c) Structure of T-CoRe (Ours).

Figure 1. Comparison of structures between our framework with previous MVM methods.

B.2. Evaluation Datasets and Metrics

DAVIS-2017 [21] is a benchmark dataset for video object segmentation. The following three metrics are commonly used to evaluate the overall segmentation performance.

1) $\mathcal{J}_{\mathrm{m}}$ measures the average region similarity by calculating the overlap between the predicted segmentation mask P_i and the ground truth mask G_i for each video $\boldsymbol{V}_i$:

$$\mathcal{J}_{\mathrm{m}} = \frac{1}{n}\sum_{i=1}^{n}\frac{|P_i \cap G_i|}{|P_i \cup G_i|}. \tag{5}$$

2) $\mathcal{F}_{\mathrm{m}}$ accesses the average contour accuracy by calculating the harmonic mean of the precision Pre_i and recall Rec_i for the predicted boundary of each video $\boldsymbol{V}_i$:

$$\mathcal{F}_{\mathrm{m}} = \frac{1}{n}\sum_{i=1}^{n}\frac{2\cdot Pre_i \cdot Rec_i}{Pre_i + Rec_i}. \tag{6}$$

3) $\mathcal{J}\&\mathcal{F}_{\mathrm{m}}$ provides a comprehensive measure by averaging $\mathcal{J}_{\mathrm{m}}$ and $\mathcal{F}_{\mathrm{m}}$:

$$\mathcal{J}\&\mathcal{F}_{\mathrm{m}} = \frac{\mathcal{J}_{\mathrm{m}} + \mathcal{F}_{\mathrm{m}}}{2}. \tag{7}$$

JHMDB [15] is primarily used for action recognition and human pose estimation. In our work, we use it for the human pose propagation task, which is evaluated using the PCK@k metric to measure the precision of predictions:

$$\text{PCK@}k = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{|S_i|}\sum_{j=1}^{|S_i|}\mathbb{1}\left[D(\hat{p}_{i,j}, p_{i,j}) < k\cdot d_i\right], \tag{8}$$

where S_i is the set of key points and d_i is the scale of the human body in video $\boldsymbol{V}_i$, $D(\hat{p}_{i,j}, p_{i,j})$ represents Euclidean distance between predicted key point $\hat{p}_{i,j}$ and ground truth key point $p_{i,j}$. The parameter k is the threshold for the maximum allowable distance error. Following previous works [7, 11, 14], we use PCK@0.1 and PCK@0.2 as evaluation metrics.

VIP [32] is designed for fine-grained instance parsing, which can be applied to the semantic part propagation task. The primary evaluation metric is mIoU, which measures the segmentation performance by calculating the overlap between the predicted segmentation mask $P_{i,j}$ and the ground truth mask $G_{i,j}$ for each video $\boldsymbol{V}_i$ and each class C_j:

$$\text{mIoU} = \frac{1}{|C|}\sum_{j=1}^{|C|}\frac{1}{n}\sum_{i=1}^{n}\frac{|P_{i,j}\cap G_{i,j}|}{|P_{i,j}\cup G_{i,j}|}. \tag{9}$$

B.3. Training Settings

Sampling strategy. For each video, we first randomly select a current frame within the range $[0.3, 0.7]$ of the total video duration. Then, we randomly sample the past and future frames relative to the current frame based on the offset range of $[0.15, 0.25]$. Moreover, we generate 2 global views (224×224) and 8 local views (96×96) from the current frame, applying standard data augmentations such as color jittering, gray scaling, and Gaussian blur. Following [19], 50% of the global crops in the student branch are masked with 10% to 50% of randomly selected patches.

Optimizing settings. We adopt ViT-Small and ViT-Base [6] with a patch size of 16 as the backbone models in our framework. The feature dimensions are set to 384 and 768, respectively. For the ViT-S/16 backbone, we pre-train our framework for 400 epochs with a batch size of 256, where the first 20 epochs are allocated for warm-up. For the ViT-B/16 backbone, we pre-train our framework for 200 epochs with a batch size of 128, with the first 10 epochs used for warm-up. The base learning rate blr for ViT-S/16 and ViT-B/16 are adaptively set to 2×10^{-3} and 1×10^{-3}, respectively, and decay to 1×10^{-6} using a cosine decay schedule. The real learning rate lr is scaled according to the batch size: $lr = blr \cdot \sqrt{bs/1024}$. The learning rates for PMM are set to $0.1 \times lr$ for ViT-S/16 and $0.13 \times lr$ for ViT-B/16. The student branch is optimized by AdamW [18] and the teacher branch is updated with the exponential moving average of the student weights.

Loss function. The hyper-parameters of the loss function are set as follows: $\lambda_1 = 0.8$, $\lambda_2 = 20$, $\lambda_3 = 1.0$, and $\lambda_4 = 0.1$. Note that λ_3 and λ_4 are taken from the default setting of [19] and we only tune the λ_1 and λ_2 in our experiments.

The general hyperparameters settings for our T-CoRe

Hyperparameter	Notation	Value	
		ViT-S/16	ViT-B/16
Sampling strategy			
Current frame range	/	[0.3, 0.7]	
Past frame offset range	$[\alpha, \beta]$	[0.15, 0.25]	
Future frame offset range	$[\alpha, \beta]$	[0.15, 0.25]	
Mask probability	/	0.5	
Mask ratio	/	[0.1, 0.5]	
Global crop size	/	(224×224)	
Local crop size	/	(96×96)	
Past and future frame size	/	(224×224)	
Optimizing settings			
Optimizer	/	AdamW	
Learning rate scheduler	/	Cosine	
Weight decay	/	$0.04 \rightarrow 0.4$	
Momentum	/	$0.992 \rightarrow 1$	
Number of ViT encoder blocks	/	12	
Patch size	p	16	
Base learning rate	blr	2×10^{-3}	1×10^{-3}
PMM learning rate	/	$0.1 \times lr$	$0.13 \times lr$
Epochs	/	400	200
Warm-up epochs	/	20	10
Batch size	bs	256	128
Number of ViT feature dim.	d	384	768
Loss function			
Weight of reconstruction loss	λ_1	0.8	
Weight of squeezing loss	λ_2	20	
Weight of DINO loss	λ_3	1	
Weight of koleo loss	λ_4	0.1	

Table 1. The hyperparameters settings for our T-CoRe framework during the training process.

Config	DAVIS-2017	VIP	JHMDB
Top-K	7	10	7
Queue Length	20	20	20
Neighborhood Size	20	20	20

Table 2. The hyperparameters settings for our T-CoRe framework during the evaluation process.

framework during the training process are summarized in Tab. 1. All experiments in this work are conducted with Pytorch [20] on a Linux machine equipped with an AMD EPYC 9654 96-Core Processor and 4 NVIDIA 4090 GPUs.

B.4. Evaluation settings

Following [7], the hyperparameters for three downstream tasks are listed in Tab. 2. Note that these hyperparameters remain fixed in our framework without further tuning to ensure a fair comparison.

B.5. Competitors

We compare our T-CoRe with various state-of-the-art self-supervised representation learning methods, which can be categorized into the following two types:

1) Contrastive learning methods:

- SimCLR [3] aims to learn meaningful representations by contrasting positive and negative samples.
- MoCo v3 [4] designs a momentum encoder with a memory bank to store negative samples.
- DINO [2] uses a self-distillation structure to learn representations with Vision Transformer [6] as the encoder.
- ODIN^2 [13] combines object discovery with representation networks to capture meaningful semantics without annotation.
- CrOC [24] proposes a cross-view consistency objective with an online clustering mechanism for semantic segmentation.

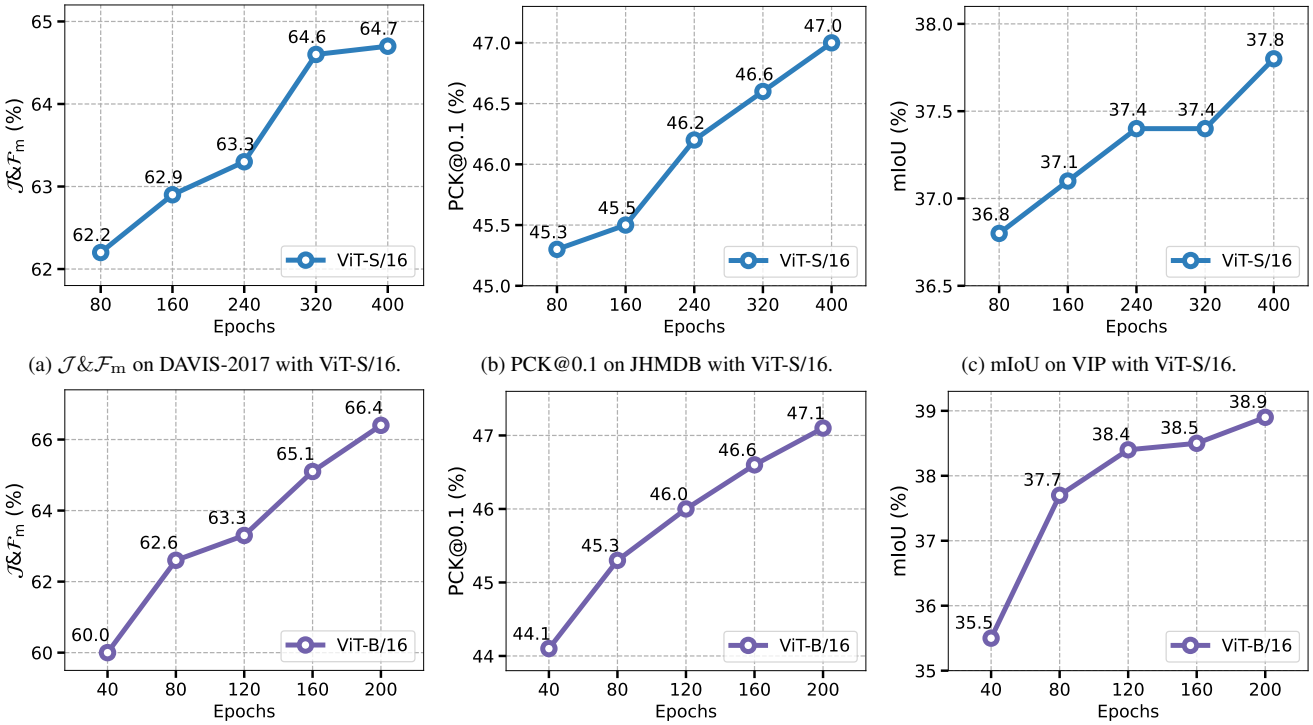
2) Masked modeling methods:

- MAE [12] proposes to mask and recover image patches at the pixel level based on an encoder-decoder structure.
- MAE-ST [8] extends MAE to learn spatiotemporal representations from videos.
- RC-MAE [17] introduces a mean teacher network into MAE for consistent reconstruction.
- VideoMAE [25] simply extends MAE into the video domain by masking video tubes with an extremely high masking ratio and recovering the masked pixels.
- DropMAE [29] applies adaptive spatial-attention dropout to enhance temporal relations in videos
- SiamMAE [11] uses a past frame and a masked current frame as input to a Siamese network, reconstructing the masked patches with a conditional decoder.
- CropMAE [7] samples different crops or augmented versions of a frame as input to a similar structure with SiamMAE.
- RSP [14] learns to recover a future frame through stochastic frame prediction, using the current frame for prior and posterior distributions.
- iBOT [31] aligns both cross-view `[CLS]` tokens and in-view patch tokens within a self-distillation framework.
- DINO v2 [19] employs a discriminative self-supervised pre-training approach and incorporates additional techniques [1, 22, 23, 26] to improve iBOT. For a fair comparison, we do not apply the costly techniques in DINO v2 like distillation from a larger model or pre-training with a larger resolution.

C. Additional Experimental Results

C.1. Training Dynamics

Fig. 2 illustrates the performance dynamics of the training schedule for ViT-S/16 and ViT-B/16 backbones across three downstream tasks. We report the performance of the framework at different checkpoints during the training process. The figures indicate that training with a larger model and a longer duration leads to further performance improvements on these downstream tasks.



(a) $\mathcal{J}\&\mathcal{F}_m$ on DAVIS-2017 with ViT-S/16. (b) PCK@0.1 on JHMDB with ViT-S/16. (c) mIoU on VIP with ViT-S/16.

(d) $\mathcal{J}\&\mathcal{F}_m$ on DAVIS-2017 with ViT-B/16. (e) PCK@0.1 on JHMDB with ViT-B/16. (f) mIoU on VIP with ViT-B/16.

Figure 2. The performance on three downstream tasks during the training phase with ViT-S/16 and ViT-B/16 backbones.

C.2. Computational Efficiency

Our framework (VIT-B/16 backbone) is pre-trained on 4 GPUs for 200 epochs with bs=128 in 30 hours. As shown in the Tab. 3, we provide a comparison of model efficiency. Although the self-distillation architecture introduces a slight computational overhead, our method achieves superior performance on downstream tasks with fewer training epochs, while maintaining relatively acceptable time and space costs, thus seeking a balance between computational efficiency and model effectiveness.

Method	Backbone	GPU Mem.	Epoch	PT-Time	$\mathcal{J}\&\mathcal{F}_m$	$\mathcal{J}_m$	$\mathcal{F}_m$
VideoMAE	ViT-B/16	8×24.39 GB	800	160 h	34.7	33.9	35.4
SiamMAE	ViT-B/16	4×5.04 GB	400	21 h	45.5	43.6	47.5
CropMAE	ViT-B/16	4×5.02 GB	400	22 h	57.8	56.9	58.7
RSP	ViT-B/16	4×12.19 GB	400	113 h	60.5	57.8	63.2
T-CoRe (Ours)	ViT-B/16	4×17.57 GB	200	30 h	**66.4**	**64.6**	**68.2**

Table 3. Efficiency comparison on ViT-B/16 with same settings.

C.3. Additional Visualization Results

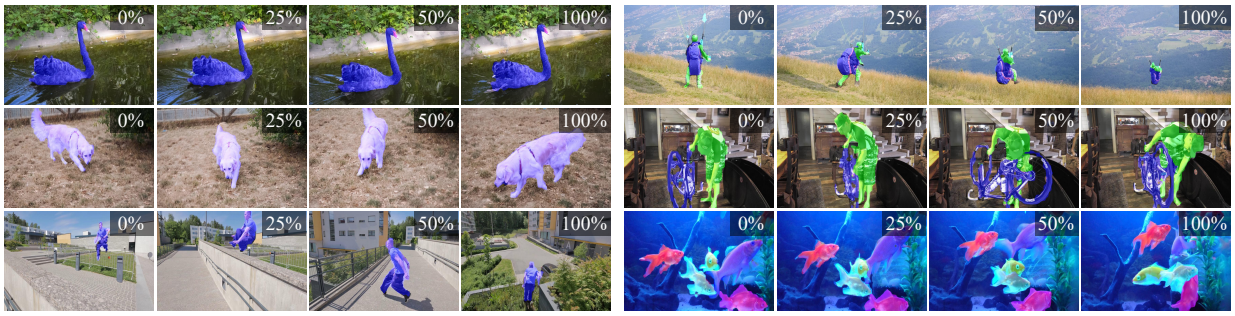
Cross-attention Maps. We provide additional cross-attention heatmaps of the masked patches between the current frame and both the past and future frames in PMM. As shown in Fig. 3, the masked patches could successfully match similar regions in the auxiliary frames through the cross-attention mechanism, demonstrating the favorable ability to establish temporal correspondence. This capability remains effective in perceiving and matching the corresponding targets even when the target is almost completely masked in the current frame. Moreover, it is worth noting that PMM can also capture edge details, such as the outline of the rope in the top-right example.

Downstream Tasks. In Fig. 4, we provide more visualization results on three downstream tasks. The prediction masks show that T-CoRe performs well in instance segmentation and posture tracking across most scenarios, making it an effective pre-training framework for video representation learning that facilitates the video understanding process.

C.4. k-NN Images for Pre-training on ImageNet-1k

To ensure a fair comparison with previous methods in the image domain, we extend our framework for pre-training on the ImageNet-1k [5]. Specifically, we employ k-NN images to simulate the adjacent frames in videos for establishing temporal correspondence. In this setting, the auxiliary branch receives one k-NN image. The k-NN images are determined based on our framework pre-trained without the auxiliary branch, where k is set to 5 at default. Fig. 5 presents several examples of original images and their corresponding k-NN images, which exhibit similar appearance and share the same semantics as the original images, effectively simulating the selection of adjacent frames from a video to establish correspondence.

Figure 3. Additional cross-attention heatmaps of the masked current frame $\bar{v}_c$ to the past and future frames v_p, v_f in the PMM.



(a) Video Object Segmentation on DAVIS-2017

(b) Human Pose Propagation on JHMDB

(c) Semantic Part Propagation on VIP

Figure 4. Additional visualization results of T-CoRe for three downstream tasks including (a) video object segmentation on DAVIS-2017 [21], (b) human pose propagation on JHMDB [15], and (c) body part propagation on VIP [32].

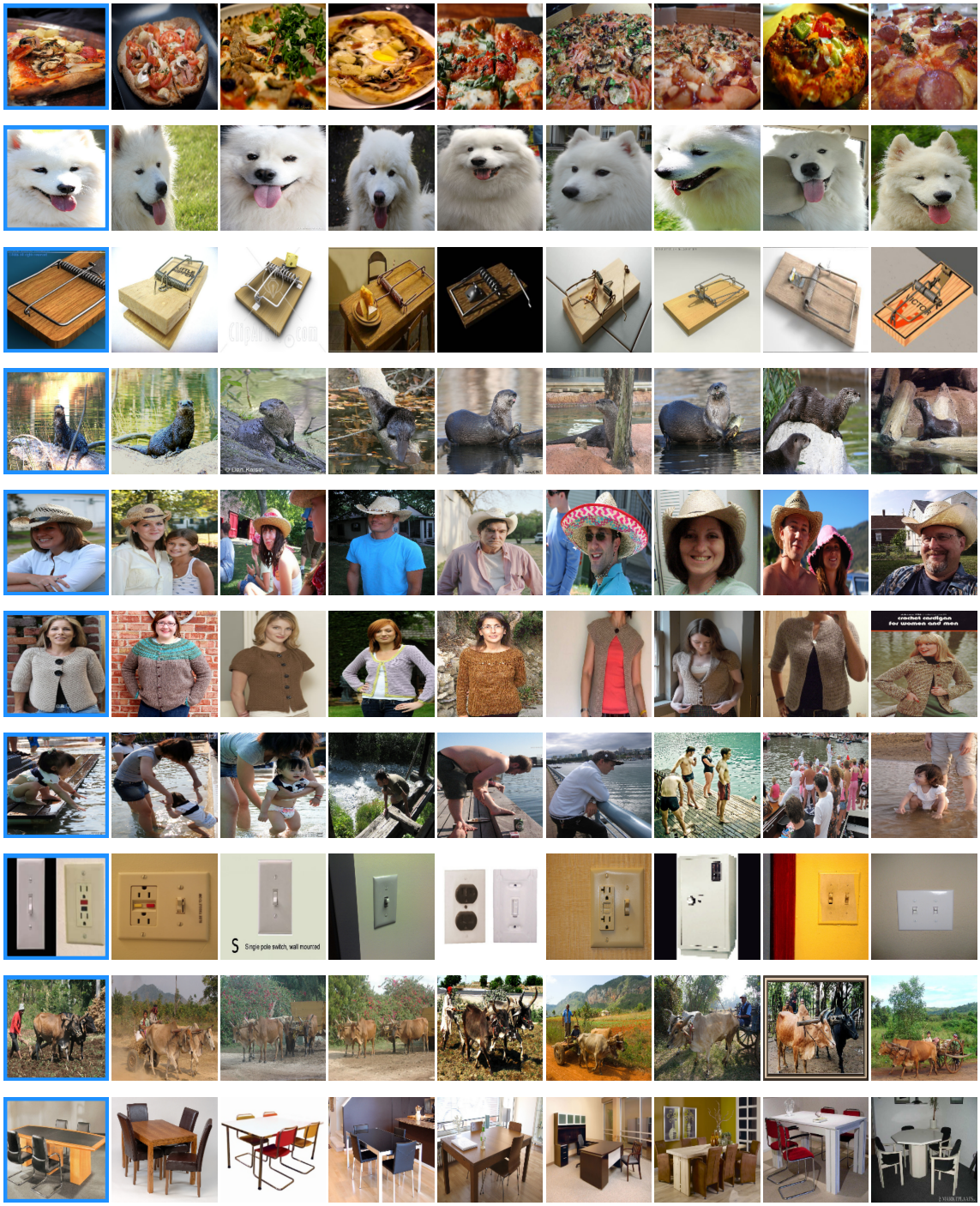


Figure 5. Examples of k-NN images for pre-training on ImageNet-1k. The images within the blue boxes in the first column are the origin images, while the following 8 columns show their top-k nearest neighbor images in the training set.

Figure 6. Failure cases on downstream tasks.

C.5. Failure Case Analysis

As shown in Fig. 6, obvious prediction errors occur in some challenging test samples, such as tightly fitting instances, considerable motion amplitudes, and significant camera movements. These failures likely result from the limitations in the training data processing, which prevents the model from learning such difficult scenarios. This issue could be mitigated by using higher-quality datasets, more precise sampling methods, and smaller patch sizes for training. In future work, we will consider further enhancing the framework to better handle more complex scenarios.

References

[1] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in Neural Information Processing Systems*, 33:9912–9924, 2020. 3

[2] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *International Conference on Computer Vision*, pages 9650–9660, 2021. 3

[3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, pages 1597–1607. PMLR, 2020. 3

[4] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *International Conference on Computer Vision*, pages 9640–9649, 2021. 3

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 248–255. Ieee, 2009. 1, 4

[6] 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 *International Conference on Learning Representations*, 2021. 2, 3

[7] Alexandre Eymaël, Renaud Vandeghen, Anthony Cioppa, Silvio Giancola, Bernard Ghanem, and Marc Van Droogenbroeck. Efficient image pre-training with siamese cropped masked autoencoders. In *European Conference on Computer Vision*, 2024. 1, 2, 3

[8] Christoph Feichtenhofer, Yanghao Li, Kaiming He, et al. Masked autoencoders as spatiotemporal learners. *Advances in Neural Information Processing Systems*, 35:35946–35958, 2022. 1, 3

[9] Rohit Girdhar, Alaaeldin El-Nouby, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Omnimae: Single model masked pretraining on images and videos. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10406–10417, 2023.

[10] Agrim Gupta, Stephen Tian, Yunzhi Zhang, Jiajun Wu, Roberto Martín-Martín, and Li Fei-Fei. Maskvit: Masked visual pre-training for video prediction. In *International Conference on Learning Representations*, 2023. 1

[11] Agrim Gupta, Jiajun Wu, Jia Deng, and Fei-Fei Li. Siamese masked autoencoders. *Advances in Neural Information Processing Systems*, 36:40676–40693, 2023. 1, 2, 3

[12] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. 3

[13] Olivier J Hénaff, Skanda Koppula, Evan Shelhamer, Daniel Zoran, Andrew Jaegle, Andrew Zisserman, João Carreira, and Relja Arandjelović. Object discovery and representation networks. In *European Conference on Computer Vision*, pages 123–143. Springer, 2022. 3

[14] Huiwon Jang, Dongyoung Kim, Junsu Kim, Jinwoo Shin, Pieter Abbeel, and Younggyo Seo. Visual representation learning with stochastic frame prediction. In *International Conference on Machine Learning*, pages 21289–21305, 2024. 1, 2, 3

[15] Hueihan Jhuang, Juergen Gall, Silvia Zuffi, Cordelia Schmid, and Michael J Black. Towards understanding action recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 3192–3199, 2013. 2, 5

[16] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 1

[17] Youngwan Lee, Jeffrey Ryan Willette, Jonghee Kim, Juho Lee, and Sung Ju Hwang. Exploring the role of mean teachers in self-supervised masked auto-encoders. In *International Conference on Learning Representations*, 2023. 3

[18] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. 2

[19] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. 2, 3

[20] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, 32, 2019. 3

[21] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017. 2, 5

[22] Yangjun Ruan, Saurabh Singh, Warren Richard Morningstar, Alexander A Alemi, Sergey Ioffe, Ian Fischer, and Joshua V Dillon. Weighted ensemble self-supervised learning. In *In-*

ternational Conference on Learning Representations, 2023. 3

[23] Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, and Hervé Jégou. Spreading vectors for similarity search. In *International Conference on Learning Representations*, 2019. 3

[24] Thomas Stegmüller, Tim Lebailly, Behzad Bozorgtabar, Tinne Tuytelaars, and Jean-Philippe Thiran. Croc: Cross-view online clustering for dense visual representation learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7000–7009, 2023. 3

[25] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *Advances in Neural Information Processing Systems*, 35:10078–10093, 2022. 1, 3

[26] Hugo Touvron, Andrea Vedaldi, Matthijs Douze, and Hervé Jégou. Fixing the train-test resolution discrepancy. *Advances in Neural Information Processing Systems*, 32, 2019. 3

[27] Limin Wang, Bingkun Huang, Zhiyu Zhao, Zhan Tong, Yinan He, Yi Wang, Yali Wang, and Yu Qiao. Videomae v2: Scaling video masked autoencoders with dual masking. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14549–14560, 2023. 1

[28] 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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14733–14743, 2022.

[29] Qiangqiang Wu, Tianyu Yang, Ziquan Liu, Baoyuan Wu, Ying Shan, and Antoni B Chan. Dropmae: Masked autoencoders with spatial-attention dropout for tracking tasks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14561–14571, 2023. 1, 3

[30] Jiange Yang, Bei Liu, Jianlong Fu, Bocheng Pan, Gangshan Wu, and Limin Wang. Spatiotemporal predictive pre-training for robotic motor control. *arXiv preprint arXiv:2403.05304*, 2024. 1

[31] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. *International Conference on Learning Representations*, 2022. 3

[32] Qixian Zhou, Xiaodan Liang, Ke Gong, and Liang Lin. Adaptive temporal encoding network for video instance-level human parsing. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 1527–1535, 2018. 2, 5