MGMAE: Motion Guided Masking for Video Masked Autoencoding

Bingkun Huang[1,2]   Zhiyu Zhao[1,2]   Guozhen Zhang[1]   Yu Qiao[2]   Limin Wang[1,2,✉]

[1] State Key Laboratory for Novel Software Technology, Nanjing University   [2] Shanghai AI Lab

https://github.com/MCG-NJU/MGMAE

Abstract

Masked autoencoding has shown excellent performance on self-supervised video representation learning. Temporal redundancy has led to a high masking ratio and customized masking strategy in VideoMAE. In this paper, we aim to further improve the performance of video masked autoencoding by introducing a motion guided masking strategy. Our key insight is that motion is a general and unique prior in video, which should be taken into account during masked pre-training. Our motion guided masking explicitly incorporates motion information to build temporal consistent masking volume. Based on this masking volume, we can track the unmasked tokens in time and sample a set of temporal consistent cubes from videos. These temporal aligned unmasked tokens will further relieve the information leakage issue in time and encourage the MGMAE to learn more useful structure information. We implement our MGMAE with an online efficient optical flow estimator and backward masking map warping strategy. We perform experiments on the datasets of Something-Something V2 and Kinetics-400, demonstrating the superior performance of our MGMAE to the original VideoMAE. In addition, we provide the visualization analysis to illustrate that our MGMAE can sample temporal consistent cubes in a motion-adaptive manner for more effective video pre-training.

1. Introduction

Attention-based Transformer [38] has witnessed great success in computer vision since the introduction of Vision Transformer (ViT) [12]. It has been applied for a variety of vision tasks and obtains state-of-the-art performance, such as image classification [36, 59, 52], object detection [23, 46], semantic segmentation [49], and object tracking [8]. Thanks to this high performance, ViT models have been also applied to the video domain for action recognition [5, 1] and detection [33, 54]. However, the high capacity of Transformer often demands pre-training on a large-scale dataset to reduce the over-fitting risk of subsequent fine-
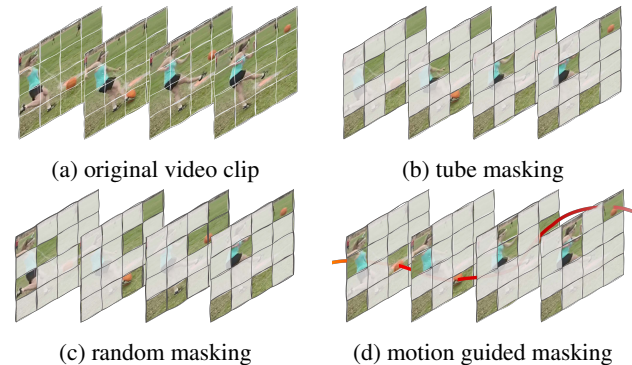
✉: Corresponding author (lmwang@nju.edu.cn).

(a) original video clip   (b) tube masking

(c) random masking   (d) motion guided masking

Figure 1: **Comparison of different masking strategies**. Masked autoencoding [11, 17] has been explored in video domain for self-supervised pre-training by employing different masking strategies: random masking [14] and tube masking [35]. We propose to track masking maps under the guidance of motion information (termed as motion guided masking). Our resulting MGMAE can build a more challenging and meaningful task for video pre-training.

tuning. Therefore, an effective pretraining strategy of ViT is particularly important for obtaining excellent performance in the video domain due to the smaller video dataset.

The early video transformers [1, 5] often rely on the pre-training of image-based transformer derived from the large-scale image dataset [10]. This pre-training scheme makes the learnt video model to be naturally biased by image-based ViTs. Recently, masked autoencoding (MAE) [35, 14, 41] has been explored for pre-training video transformer on the video dataset due to its simplicity and promising result in image domain [17]. However, unlike the image, video data is equipped with an extra time dimension and exhibits the unique property of temporal redundancy and correlation. This property requires some customized designs on video masked autoencoder compared with image-based MAE. For example, VideoMAE and MAE-ST both propose to use an extremely high masking ratio in video masked autoencoder pre-training to improve its performance. In addition, VideoMAE devises a tube masking strategy of dropping tokens at the same position across frames to further relieve the information leakage in time. This tube masking

approach, though straightforward, makes the assumption of no or small motion occurring between adjacent frames. Such an assumption might be not true for some scenarios with high-speed motion.

Based on the above analysis, in this paper, we aim to propose a new masking strategy for improving video masked autoencoder pre-training, by explicitly using motion information to reduce information leakage in time. Specifically, we devise the *Motion Guided Masking* in the video masked encoder processing and the resulted masked autoencoder is termed as **MGMAE**. Motion is general prior information contained by video. The optical flow representation explicitly encodes the movement of each pixel from the current frame to the next one. We propose to use this optical flow to align masking maps between adjacent frames to build consistent masking volumes across time. The consistent masking volumes enable to build a more challenging reconstruction task by enforcing only a small set of cube tracks visible to the encoder. Hopefully, this motion guided masking can further relieve the risk of information leakage in time and encourage learning more meaningful visual representations.

More specifically, we use an online and lightweight optical flow estimator (RAFT [34]) to capture motion information, which could be seamlessly integrated into the existing VideoMAE framework. To build the temporally consistent masking volume, we first randomly generate an initial masking map at the base frame. Then, we use the estimated optical flow to warp the initial masking map to adjacent frames. With multiple warping operations, we build the temporal consistent masking volume for all frames in the video. Finally, based on this masking volume, we sample a set of visible tokens to MAE encoders with top-k selection based on a frame-wise manner. The same autoencoding process with the original VideoMAE is applied to these sampled tokens for video pretraining. With this simple motion guided masking, we are able to further increase the difficulty of video pre-training task and thus lead to a better pre-trained model for subsequent fine-tuning.

We mainly verify the effectiveness of the proposed MGMAE on the datasets of Something-Something V2 [16] and Kinetics-400 [20] by comparing them with the original tube masking in VideoMAE. The results demonstrate that MGMAE pre-training can result in more powerful video foundation models with higher fine-tuning accuracy on the downstream tasks. In particular, on the motion-centric benchmark of Something-Something, the improvement of MGMAE is more evident, implying that our motion guided masking is adaptive to motion variations and can better capture temporal structure information for pre-training. We hope our findings can inspire some specific and unique designs in video masked autoencoding with respect to image counterparts.

2. Related Work

Masked Visual Modeling. Masked autoencoder is a long-standing unsupervised learning framework in computer vision. The early work presented general form of denoising autoencoder [40, 39] for learning representation by reconstructing the clean signal from the noisy inputs. The other work [27] also treated masking modeling as inpainting missing regions from the surrounding context by using convolutions. Inspired by the great success of masked language modeling [11], some works also attempted to apply this pre-training paradigm to the vision domain for self-supervised pre-training. For example, iGPT [7] followed the GPT work [30] in NLP and processed a sequence of pixels for casual prediction of the next pixels. The original ViT [12] used the masked token prediction as a self-supervised training step on large-scale image datasets but failed to obtain impressive results. Recently, several interesting works have obtained a great breakthrough in self-supervised image pre-training by using masked image modeling, such as BEiT [3], SimMIM [50], and MAE [17]. BEiT [3] directly followed the BERT framework and proposed to predict the discrete token label for masked patches, by requiring an explicit tokenizer to build the token dictionary. SimMIM [50] and MAE [17] shared the same design of directly predicting the pixels of masked patches without any tokenizer design. Furthermore, MAE [17] devised an asymmetric encoder-decoder architecture to speed up the masked image pre-training.

Since the great success in masked image modeling, some works have tried to extend this new pre-training paradigm to the video domain for self-supervised video pre-training. BEVT [45] and VIMPAC [32] proposed to learn video representation by predicting discrete visual tokens in a similar way to BEiT. However, their performance improvement in video action recognition is limited. MaskFeat [48] used the HOG features [9] as the reconstructed targets of masked patches and achieved excellent performance on the video recognition with a multi-scale vision transformer. VideoMAE [35] and MAE-ST [14] extended the image MAE to the video domain for representation learning with vanilla vision transformer. They both proposed to use an extremely high masking ratio to deal with video data redundancy. Meanwhile, Video-MAE [35] used the tube masking to further increase the difficulty of reconstruction. Several works building upon VideoMAE have emerged. For instance, MAR [29] reduced both training and inference costs by introducing running cell masking. Meanwhile, VideoMAE V2 [41] proposes a dual masking strategy to decrease pre-training overhead, and by expanding both the model size and dataset, it further explores the scalability of VideoMAE. Our proposed motion guided masking aims to improve the performance of VideoMAE by building a more challenging masking and reconstruction task. In contrast to the original VideoMAE, our MGMAE explicitly use the optical flow to align the masking maps

across frames and generate the temporal consistent masking volume to sample a set of visible tokens.

Motion Guided Modeling. Motion information, such as optical flow, is a general prior information in videos and represents the unique characteristics distinct from images. Optical flow has been widely introduced to provide a strong prior in both low-level and high-level vision tasks on video. For low-level video tasks, the motion is often used to align the information of auxiliary frames to the corresponding region of the target frame. In the case of video super-resolution, BasicVSR++ [6] uses optical flow to enhance the appearance of low-resolution frames by transferring features from neighbor frames. For video inpainting, Zhang *et al.* [56] exploits the motion difference extracted by optical flows to instruct the attention retrieval in transformer for high-fidelity video inpainting. As for video frame interpolation, mainstream methods leverage optical flow directly on the image to synthesize the intermediate frame, such as DAIN [4] and RIFE [18], while Zhang *et al.* [55] introduces a unified operation utilizing inter-frame attention to concurrently extract motion and appearance information, and blends a hybrid CNN and Transformer design for efficiency and fine-grained detail preservation. For high-level video tasks, the optical flow is directly used as a data modality as network input for action recognition [44, 31]. TDD [42] utilized motion trajectories to pool deep convolutional features for action recognition. Trajectory Convolution [57] incorporated the motion information into temporal convolutional kernel design. MSNet [21] proposes a pluggable MotionSqueeze module to generate motion information across frames. VideoMS [19] generates mask maps by calculating the feature difference after patch embedding, making an attempt at dynamically adjusting mask positions. AdaMAE [2] introduces an end-to-end trainable adaptive masking strategy for MAEs, leveraging an auxiliary sampling network to prioritize tokens from high spatiotemporal information regions. Yang *et al.* [51] leverage hierarchical motion information to improve the extracted video features. MotionFormer [28] employed the trajectory for attention computation in the video transformer. TEA [22] and TDN [43] used RGB difference to approximate the motion information and incorporate this information into the video CNN backbone design. MGSampler [58] explored the motion information to select a subset of representative frames for efficient video action recognition. Our MGMAE shares the same spirit with these motion guided modeling works. We focus on employing motion information as a cue to generate masking maps for masked video pre-training.

3. Method

In this section, we first revisit the pre-training paradigm of VideoMAE to well introduce our MGMAE in Sec. 3.1.

Then we present the details of motion guided masking map generation in Sec. 3.2. Finally, we describe the MGMAE pre-training under temporal consistent masking maps in Sec. 3.3.

3.1. VideoMAE revisited

VideoMAE is a simple masked video autoencoder with an asymmetric encoder-decoder architecture with an extra cube embedding to handle the input sampled frames. Next, we briefly revisit its implementation detail.

Cube Embedding. VideoMAE divides the input video clip $\mathbf{I}$ of size $T \times 3 \times H \times W$ into non-overlapping cubes $\mathbf{C} = \left\{ \mathbf{C}_i \mid \mathbf{C}_i \in \mathbb{R}^{2 \times 16 \times 16 \times 3} \right\}_{i=1}^N$, where $N = \frac{T}{2} \times \frac{H}{16} \times \frac{W}{16}$ is the number of cubes. Then apply cube embedding on the cubes to produce the video tokens $\mathbf{T} = \left\{ \mathbf{T}_i \mid \mathbf{T}_i \in \mathbb{R}^D \right\}_{i=1}^N$, where T_i represents the cube embedding with positional encoding, and D is the channel.

Masking Strategy. VideoMAE uses the tube masking strategy with an extremely high masking ratio ρ (*i.e.* 90%), which samples the same spatial positions across all frames of the input video clip. Specifically, VideoMAE first generate a $\frac{H}{16} \times \frac{W}{16}$ binary mask map $\mathbf{M}'$ where 0 represents unmasked and 1 represents masked. Then it replicates it in temporal dimension and then flattens it to produce the token-level mask map $\mathbf{M}$ whose size is N for the input video clip. We denote $\mathbb{M}$ as the masking maps.

Encoder. The encoder is a vanilla ViT with joint space-time attention [5]. For computation efficiency, only the unmasked visible tokens $\mathbf{T}^v = \{\mathbf{T}_i\}_{i \notin \mathbb{M}}$ added with the fixed positional embedding are fed into the encoder to obtain the latent features $\mathbf{Z}$ of size $N_v \times D$, where $N_v = \lfloor (1-\rho)N \rfloor$ is the total number of the unmasked visible tokens.

Decoder. The decoder is a narrower and shallower ViT than the encoder. It takes the concatenated token sequences as input, which is formed by the concatenation between the latent features $\mathbf{Z}$ and the learnable $[\texttt{MASK}]$ tokens with the fixed position embedding added, to reconstruct the normalized video cubes $\hat{\mathbf{C}} = \left\{ \hat{\mathbf{C}}_i \mid \hat{\mathbf{C}}_i \in \mathbb{R}^{2 \times 16 \times 16 \times 3} \right\}_{i=1}^N$.

Loss. The pre-training object is to minimize the *Mean Square Error* Loss between the normalized $\mathbf{C}$ and $\hat{\mathbf{C}}$ on the masked positions, *i.e.* $\frac{1}{\rho N} \sum_{i \in \mathbb{M}} \left| \hat{\mathbf{C}}_i - \text{norm}(\mathbf{C}_i) \right|^2$.

After the pre-training, the encoder will be used as the backbone network to fine-tune on the downstream tasks to obtain a specialized model.
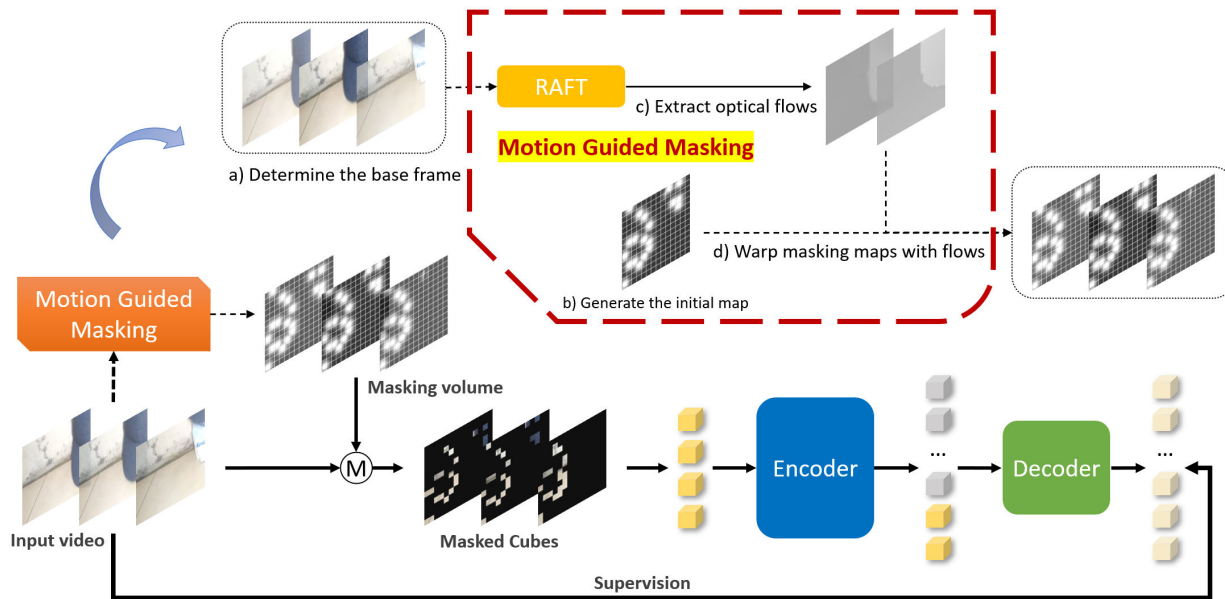
Figure 2: **Pipeline of MGMAE**. Our MGMAE follows the simple pipeline of masking and reconstruction for video self-supervised pre-training. Our core design is to propose a motion guided masking strategy to generate temporal consistent masking volume. With this masking volume, we track the visible cubes and attain temporal consistency for masking maps. As a result, we can build a more challenging reconstruction task and encourage extracting more effective representations during masked self-supervised pretraining.

3.2. Motion Guided Masking Map

Time is a unique characteristic of video and has different properties with the space dimension. When devising masked video autoencoder, we need to carefully take this extra time dimension into account and come up with a customized design. Information leakage in time is an important issue in masked video pre-training. When information leakage occurs, the model can easily reconstruct the masked cubes based on the visible tokens of adjacent frames. In this case, it will greatly reduce the difficulty of the reconstruction task and lead to a pre-trained model with poor fine-tuning performance. A trivial solution to information leakage is to increase the masking ratio. VideoMAE [35] and MAE-ST [14] increase the masking ratio to 90% to greatly increase the difficulty of reconstruction. In addition, Video-MAE makes the small motion assumption and adapts the tube masking strategy, which masks the same spatial position in all frames. However, this small motion prior is not always true for motion-dominated videos. A more reasonable approach is to keep each object in the video clip visible or invisible at all times. To achieve this goal, we propose the motion guided masking strategy to replace the tube masking strategy in VideoMAE. The strategy has two procedures: we first use the optical flow as guidance to generate temporally consistent masking volumes of the input video clip

and then sample the unmasked visible tokens based on the temporal consistent masking volume. We will detail these two procedures in Sec. 3.2 and Sec. 3.3.

In general, the procedure for generating the temporal consistent masking volumes has four steps as follows.

- Step 1: Determine the base frame $\mathbf{I}_b$, where b is the index of the base frame.

- Step 2: Randomly generate a pixel-level initial mask map $\mathbf{M}_b$ with size $H \times W$ as the mask map of $\mathbf{I}_b$.

- Step 3: Extract the dense flows $\mathbf{F}$ bidirectionally from the base frame $\mathbf{I}_b$ in the input video clip $\mathbf{I}$.

- Step 4: Warp the initial masking map $\mathbf{M}_b$ under the guidance of dense flows $\mathbf{F}$ and progressively build the temporal consistent masking volume $\mathbf{M}$ of size $T \times H \times W$.

Determine the base frame. By default, we choose the *middle frame* as the base frame. In motion guided masking, we need to ensure all objects in the base frame remain consistently visible or invisible in all frames of the input clip. Note that objects may (dis)appear over time due to object or camera movement, and warping the masking map under optical flow can result in some holes due to pixels mapped

out of bounds. So the choice of the base frame may have an impact on the suppression of information leakage. We ablate the choice of the base frame in Sec. 4.2 and the middle frame is the optimal choice.

Generate the initial mask map. We initialize a pixel-level masking map for the base frame with the distribution of the Gaussian Mixture Model (GMM). We use the masking map to indicate the visible or invisible state of the cubes in the base frame. Previous masking strategies usually adapt a token-level binary initialization, *i.e.* either all 0 or all 1 within each token of size $2 \times 16 \times 16$, which actually breaks the continuity of the object surface texture.

Specifically, we first randomly pick $\hat{N}_v = \lfloor (1 - \rho) \times \frac{H}{16} \times \frac{W}{16} \rfloor$ tokens whose centers are denoted with $c = \{\vec{c_i} : (c_{i1}, c_{i2})\}_i^{\hat{N}_v}$. Then we generate 2D Gaussian distributions $\mathcal{N}_i(c_i, \sigma^2)$ centered on the midpoint of each token, where σ is the standard deviation and taken as the cube size $(16, 16)$. Thus we will obtain the mixed Gaussian distribution $\mathcal{P}(c, \sigma^2) = \sum_i^{\hat{N}_v} \mathcal{N}_i(\vec{c_i}, \sigma^2)$ corresponding to the base frame. And the probability density function of the mixed Gaussian distribution is used to indicate the probability that the cube (token) is visible in the base frame.

Extract optical flows. We use both online and offline alternative methods to extract optical flow. Online method adapts RAFT [34] (the small version) to estimate the flows of the input video clip. Offline method applies the traditional TVL1 [53] algorithm to extract the dense flows between all adjacent frames in advance. We perform the consistent crop-resize-rescale operations when reading flows. Online and offline methods achieve the similar results. More details see in Sec. 4.2.

In practice, we only extract the flows $\mathbf{F}$ forward and backward from the base frame $\mathbf{I}_b$, i.e.

$$\mathbf{F} = \{v_{i \to i+1}\}_{i=1}^{b-1} \cup \{v_{i \to i-1}\}_{i=b+1}^{T}, \qquad (1)$$

where the flow $v_{i \to j}$ denotes the flow from $\mathbf{I}_i$ to $\mathbf{I}_j$.

Warp masking maps with flows. We utilize the method of *backward warping* to generate the temporal consistent masking map of the video clip in a progressive manner. Forward warping ϕ_F and backward warping ϕ_B are two opposite patterns of warping. Both can be effectively employed to construct the masking volume from the initial masking map. Regrettably, forward warping suffers from hole or occlusion problems, that is, no flow vectors may pass to a certain pixel, or there may be multiple flow vectors passing to the same pixel. In contrast, backward warping maps the pixels of a given map one by one to individual locations. It's noteworthy that while backward warping doesn't escape from the issue of holes caused by mapping out of bounds, these holes are

usually fewer than in forward warping and tend to occur at the boundaries of the map, thus causing less damage to the information distribution. For the holes caused by backward warping, we fill them with the values of $\mathbf{M}_b$ at the same position to simulate the tube masking strategy.

Formally, given the flows $\mathbf{F}$ and the base frame masking volume $\mathbf{M}_b$, the mask map $\mathbf{M}_i$ of $\mathbf{I}_i$ can be constructed as

$$\mathbf{M}_i = \begin{cases} \phi_B(\mathbf{M}_{i+1}, v_{i \to i+1}), & 1 \le i < b \\ \mathbf{M}_b, & i = b \\ \phi_B(\mathbf{M}_{i-1}, v_{i \to i-1}), & b < i \le T \end{cases} . \qquad (2)$$

Subsequently, the entire masking volume $\mathbf{M} = \{\mathbf{M}_i\}_{i=1}^{T}$ of the video clip $\mathbf{I}$ can be constructed by backward warping the flows $\mathbf{F}$ bidirectionally, originating from the base frame mask map $\mathbf{M}_b$.

3.3. Motion Guided MAE

We build our MGMAE based on the above motion guided masking map. The temporal consistent masking volume indicates the probability that the corresponding position in the adjacent frame is visible under optical flow tracking. In order to suppress information leakage as much as possible, we sample the video tokens with the highest visible probability along the temporal dimension. Specifically, we first perform average pooling with kernel size $2 \times 16 \times 16$ on the masking volume $\mathbf{M}$ to obtain the token-level masking volume $\mathbf{M}'$ of size $\frac{T}{2} \times \frac{H}{16} \times \frac{W}{16}$. Then we pick the top-$\hat{N}_v$ locations of each mask map of size $\frac{H}{16} \times \frac{W}{16}$ along the temporal dimension and thus sample N_v corresponding video tokens as the unmasked visible tokens.

These sampled tokens according to our temporal consistent masking volume are fed into the asymmetric encoder-decoder for autoencoding based pre-training. The resulted pre-training framework is called *Motion Guided Masked Autoencoder* (MGMAE). The pre-trained model by our MG-MAE is applied in the same way with the original Video-MAE for fine-tuning the downstream tasks.

Discussion. Previous works [35, 14] extended MAE to the video domain. They opted for the random (agnostic) masking and tube (space-only) masking strategies, respectively. *Random masking* introduces no explicit inductive bias about the video's space and time dimension. It aims to present a unified feature representation learning framework with minimal domain knowledge. We argue that although this idea is simple, time is inherently a distinctive dimension from space. By recognizing this, we could better leverage this prior information for enhanced video masked autoencoding. *Tube masking* assumes that a large area of the frame contains no or small motion, and thus masking the same position across frames could greatly reduce the information leakage risk. However, for motion-dominated video datasets such as

Something-Something, this assumption will no longer hold true. Our proposed *motion guided masking* offers a more general and conceptually simple solution to take temporal correlation into account. It could be viewed as an adaptive video masking strategy and create more challenging yet meaningful tasks in video pre-training.

4. Experiments

4.1. Dataset

Following the original VideoMAE, we evaluate our MGMAE on Kinetics-400 (K400) [20] and Something-Something V2 (SSV2) [16]. K400 contains about 240k training videos and 20k validation videos from YouTube and the actions in K400 are usually coupled with specific objects or scenes, such as brushing teeth and playing piano. While SSV2 contains about 169k training videos and 25k validation videos, and the categories in SSV2 only care about specific motion patterns (e.g. push, pull). We first pre-train the video transformer with our MGMAE on the corresponding dataset for self-supervised representation learning. Then, we report the fine-tuning performance of pre-trained models on the target datasets for action recognition. In our MGMAE pre-training, we generally follow the setting and implementation of the original VideoMAE [35]. We use the RAFT [34] to extract optical flow due to its efficiency and accuracy in our MGMAE pre-training. The implementation details on our MGMAE could found in the appendix.

4.2. Ablation Studies

In this subsection, we conduct in-depth ablation experiments on the choice in each step of our motion guided masking strategy. We pre-train the ViT-base model 800 epochs on the SSV2 dataset with 16 80G-A100 GPUs, and then fine-tune the encoder on the SSV2 dataset for action recognition. All models share the same training schedule and report the 2 clips × 3 crops accuracy.

Choice of the base frame. In this study, we investigate the influence of base frame selection for initial masking generation process. We compare the middle frame as the based frame with either the first or a random frame, and the result is shown in Tab. 1a. It implies the middle frame is the best.

Warping method with optical flow. We compare two kinds of warping methods to align masking maps across frames as explained in Sec. 3.2. As previously mentioned above, the forward warping often leads to more severe occlusion and hole problems in the masking warping process. On the contrary, backward warping can effectively relieve this issue and ensure a more smooth masking warping. The

result in Tab. 1b demonstrates that back warping contributes to better performance.

Sampling strategy of top-k visible tokens. We examine and compare the two sampling strategies to select the visible tokens based on our temporal consistent masking volume. *Frame-level* strategy samples the top-k locations for each frame independently, while *clip-level* strategy samples the top-k locations for the entire video jointly. As in Tab. 1c, the frame-level top-k sampling strategy achieves slightly better performance.

Masking initialization at the base frame. We ablate the choice of generating the initial masking at the base frame as shown in Tab. 1d. The token-level initialization method divides the mask map into $\frac{H}{16} \times \frac{W}{16}$ tokens of size 16×16, and randomly sets 90% tokens to 0 (representing masked) and 10% tokens to 1 (representing unmasked). The pixel-level initialization method randomly sets 90% pixels to 0 and 10% pixels to 1. The initialization process of the mixed Gaussian method has been detailed in Sec. 3.2. The result demonstrates that the mixed Gaussian initialization method works the best.

Hole filling method. We investigate the various methods to handle the holes problem brought by the mapping out of bound in backward warping in Tab. 1e. To determine the real holes caused by warping, we set the 0 in the initial mask map to value $1e-8$, and then the locations equal to 0 in new mask maps are treated as the holes. We experiment with 5 methods to fill the holes: *Invisible* method fills all holes with 0, while *Visible* method fills all holes with 1. *Random* method randomly fills the holes to 0 with probability of masking ratio ρ and to 1 with probability of $1 - \rho$. *Previous map* method fills holes using the values from the same spatial positions as the last generated mask map. Conversely, *Tube* method fills the holes with the value from the corresponding positions of the initial mask map, aligning with tube masking principles. We see that the tube method performs the best among all the methods.

Method of optical flow estimation. We evaluate the effect of different methods of optical flow estimation as shown in Tab. 1f. For the offline method, we use the TVL1 algorithm to extract optical flows in advance and it achieve a comparable accuracy to the online RAFT optical flow. Although VideoMAE is 1.3 times faster to train than MGMAE with RAFT-small (set to 6 testing iterations) to estimate flows, MGMAE has a clear advantage in terms of performance and reducing the risk of overfitting. We find that the offline method is not much faster than the online method because IO (reading optical flow from disk) will be a bottleneck to

case	Acc@1	Acc@5
first frame	70.5	92.7
random frame	70.6	92.9
middle frame	**71.0**	**93.1**

(a) **Base frame selection.** We perform ablation study to select the base frame as the first, random or the middle frame.

case	Acc@1	Acc@5
forward	70.5	92.9
backward	**71.0**	**93.1**

(b) **Warping method.** We choose forward or backward warping method for aligning masking maps.

case	Acc@1	Acc@5
clip-level	70.7	92.9
frame-level	**71.0**	**93.1**

(c) **Sampling strategy.** We perform a study to choose the visible token generation strategy based on motion guided masking volume.

case	Acc@1	Acc@5
token rand.	70.9	93.0
pixel rand.	70.8	93.0
mixed Gauss	**71.0**	**93.1**

(d) **Masking initialization.** We compare three methods to generate the masking map in the base frame. Two binary random methods and one mixed Gaussian method.

case	Acc@1	Acc@5
random	70.8	92.9
invisible	70.7	92.8
visible	70.8	**93.1**
previous map	70.6	92.8
tube	**71.0**	**93.1**

(e) **Hole filling.** We choose some baseline choices to fill the value in the hole place caused by warping. We also use the tube filling consistent with the tube masking.

method	time	Acc@1
None (VideoMAE)	32 h	69.6
TVL1 (offline)	41 h	71.2
RAFT with 6 iters	43 h	**71.3**
RAFT with 12 iters	56 h	71.0

(f) **Method of optical flow estimation.** We perform the ablation study to investigate the influence of different methods of optical flow estimation.

Table 1: Parts of the ablation experiments on the Something-Something V2 dataset. Our MGMAE pre-training is implemented with the 16-frame vanilla ViT-B backbone. All models are pre-trained for 800 epochs and the masking ratio is $\rho = 90\%$. The inference protocol is to report the fine-tuning action recognition accuracy with 2 clips $\times$ 3 crops. The default choice for our model is colored in gray . Although the default setting is not optimal in terms of the method of optical flow estimation, we believe that this does not affect the conclusions of the ablation experiments.

increase the training speed. Note that our default setting is not optimal, but the conclusions drawn in other ablation experiments should not be impacted.

Masking ratio. The performance of MGMAE highlights the importance of improving masking strategies even at high masking rates (*e.g.* 90%). However, after applying MGMAE, it remains questionable whether such high masking rates are still necessary. Indeed, as pointed out by [35, 14], blindly increasing the masking rate could potentially degrade the model performance. Our ablation study presented in Fig. 3 shows that sustaining a extremely high masking rate of over 80% is also crucial even with MGMAE. We think the video background and large objects mainly drive the need for a high masking ratio. Video backgrounds are often wide and simple. If the mask ratio is not high enough, the model can still rebuild pixels from other background parts, even if nearby frames mask similar sections. For large objects, a lower ratio might let the model use the texture from a different part of the object when another section is masked. It can also be observed that MGMAE performs optimally with 85% masking ratio, but 90% still seems to be a decent choice when considering the trade-off between training efficiency and performance.

Exposure of masked objects. Another proposition worth considering is whether occasional exposure of masked objects help with masked video modeling pre-train. We com-
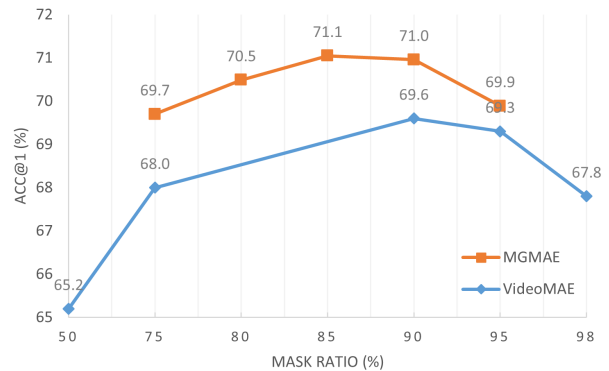

Figure 3: The effect of masking ratio on SSV2.

pleted a complementary experiment. Specifically, after building the masking volume, we add Gaussian noise on the mask map of *one randomly selected frame*. This modification may provide a chance for masked objects to be exposed. The results showed an accuracy of 71.2%, slightly higher than the default setting of 71.0%.

4.3. Main Results and Visualization Analysis

After the detailed ablation study on the design of MG-MAE, we further perform a deeper analysis by comparing it with the original VideoMAE. We also provide some intermediate visualization results to illustrate the motion guided

Model	Epochs	K400	SSV2
VideoMAE	800 / 800	0.5875	0.5278
MGMAE		0.6462	0.5820
Δ loss	-	$\Delta + 0.0605$	$\Delta + 0.0542$
VideoMAE	1600 / 2400	0.5809	0.5122
MGMAE		0.6378	0.5659
Δ loss	-	$\Delta + 0.0569$	$\Delta + 0.0537$

Table 2: Pre-train loss comparison of MGMAE and Video-MAE.

Model	Epochs	K400	SSV2
VideoMAE	800 / 800	80.0	69.6
MGMAE		81.2	71.0
Δ Acc@1	-	$\Delta + 1.2\%$	$\Delta + 1.4\%$
VideoMAE	1600 / 2400	81.5	70.8
MGMAE		81.8	72.3
Δ Acc@1	-	$\Delta + 0.3\%$	$\Delta + 1.5\%$

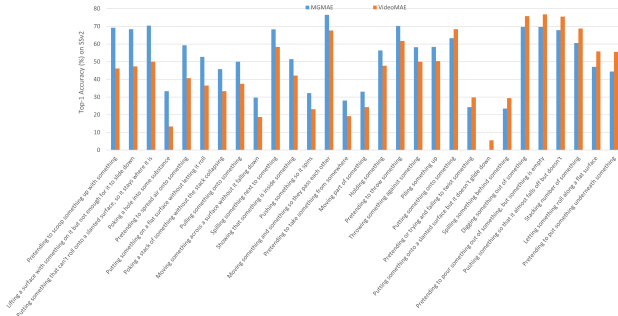Table 3: Accuracy comparison of MGMAE and VideoMAE.

Figure 4: Comparative Accuracy of MGMAE and Video-MAE by Class.

sampling process.

Pre-train loss implies a more challenging task. The core design of MGMAE is to dynamically sample the positions of masked token under the guidance of optical flows and aims at increasing the difficulty of the reconstruction task. As can be seen in Tab. 2, the pre-train loss of MGMAE is always larger than that of VideoMAE by more than 0.05. This loss gap implies motion guided masking further suppresses information leakage and indeed constructs a more challenging mask-reconstruct pretext task for masked video modeling. This more difficult task would like to encourage learning more effective representations.

Detailed breakdown of comparison between MGMAE and VideoMAE. To understand the distinct impacts of the MGMAE and VideoMAE masking strategies on video model pre-training, we delved deeper into the per-class accuracy
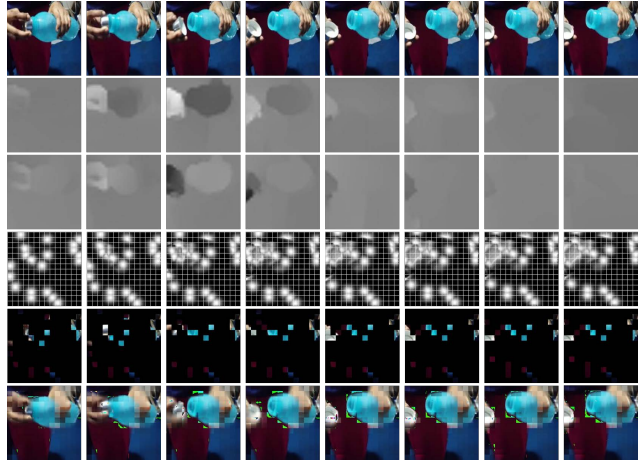

Figure 5: Visualization of the video in SSV2 validation set. We present our motion guided mask maps and the reconstructed images. From top to bottom, the original images, the x-direction flows, the y-direction flows, the motion guided mask maps, the masked images and the reconstructed images.

variations between the two. Fig. 4 showcases the 29 categories with the most pronounced differences in classification accuracy between MGMAE and VideoMAE.

MGMAE: a more effective video representation learner. MGMAE benefits greatly from the harder task constructed by our motion guided masking strategy. On the one hand, the model has to encode the relationship between visible and invisible tokens harder, which can better guide the model training. On the other hand, the suppression of information leakage may well reduce the overfitting risk of the pre-training, and thus the model can be pre-trained much longer. As shown in Tab. 3, MGMAE consistently maintains an obvious fine-tuning performance gap with VideoMAE on the motion-centric SSV2 dataset (1.4% at 800 epochs and 1.5% at 2400 epochs) and also has some improvement on the scene-centric Kinetics-400 dataset (1.2% at 800 epochs and 0.3% at 1600 epochs).

Visualization. We randomly seletec a video clip in SSV2 validation set and show its reconstruction example in Fig. 5. We can see that the mask map changes with object movements, which makes it more difficult for the model to reconstruct the original video.

4.4. Comparison with the state-of-the-art methods

We compare our approach with the previous state-of-the-art methods on the Kinetics-400 and Something-Something V2 datasets. The results are shown in Table 5 and Table 4.

Method	Backbone	Pre-train data	Frames	GFLOPs	Param	Acc@1	Acc@5
TEINet$_{En}$ [24]	ResNet50$_{\times2}$		8+16	99×10×3	50	66.5	N/A
TANet$_{En}$ [26]	ResNet50$_{\times2}$	ImageNet-1K	8+16	99×2×3	51	66.0	90.1
TDN$_{En}$ [43]	ResNet101$_{\times2}$		8+16	198×1×3	88	69.6	92.2
SlowFast [15]	ResNet101	Kinetics-400	8+32	106×1×3	53	63.1	87.6
MViTv1 [13]	MViTv1-B		64	455×1×3	37	67.7	90.9
TimeSformer [5]	ViT-B	ImageNet-21K	8	196×1×3	121	59.5	N/A
TimeSformer [5]	ViT-L		64	5549×1×3	430	62.4	N/A
ViViT FE [1]	ViT-L		32	995×4×3	N/A	65.9	89.9
Motionformer [28]	ViT-B	IN-21K+K400	16	370×1×3	109	66.5	90.1
Motionformer [28]	ViT-L		32	1185×1×3	382	68.1	91.2
Video Swin [25]	Swin-B		32	321×1×3	88	69.6	92.7
VIMPAC [32]	ViT-L	HowTo100M+DALLE w/o label	10	N/A×10×3	307	68.1	N/A
BEVT [45]	Swin-B	IN-1K+K400+DALLE w/o label	32	321×1×3	88	70.6	N/A
MAE-ST$_{1600e}$ [14]	ViT-L	Kinetics-400 w/o label	16	597×1×3	305	72.1	93.9
MaskFeat↑312 [48]	MViT-L	Kinetics-600	40	2828×1×3	218	75.0	95.0
VideoMAE$_{800e}$ [35]	ViT-B	SSV2 w/o label	16	180×2×3	87	69.6	92.0
VideoMAE$_{2400e}$ [35]	ViT-B		16	180×2×3	87	70.8	92.4
MGMAE$_{800e}$	ViT-B	SSV2 w/o label	16	180×2×3	87	71.0	93.1
MGMAE$_{2400e}$	ViT-B		16	180×2×3	87	**72.3**	**93.5**

Table 4: **Comparison on the Something-Something V2 dataset**. We only list the results obtained with the similar backbones.

Method	Backbone	Pre-train data	Frames	GFLOPs	Param	Acc@1	Acc@5
NL I3D [47]	ResNet101		128	359×10×3	62	77.3	93.3
TANet [26]	ResNet152	ImageNet-1K	16	242×4×3	59	79.3	94.1
TDN$_{En}$ [43]	ResNet101		8+16	198×10×3	88	79.4	94.4
TimeSformer [5]	ViT-L		96	8353×1×3	430	80.7	94.7
ViViT FE [1]	ViT-L	ImageNet-21K	128	3980×1×3	N/A	81.7	93.8
Motionformer [28]	ViT-L		32	1185×10×3	382	80.2	94.8
Video Swin [25]	Swin-B		32	282×4×3	88	**82.7**	**95.5**
VIMPAC [32]	ViT-L	HowTo100M+DALLE w/o label	10	N/A×10×3	307	77.4	N/A
BEVT [45]	Swin-B	IN-1K+DALLE w/o label	32	282×4×3	88	80.6	N/A
ip-CSN [37]	ResNet152		32	109×10×3	33	77.8	92.8
SlowFast [15]	R101+NL	None	16+64	234×10×3	60	79.8	93.9
MViTv1 [13]	MViTv1-B		32	170×5×1	37	80.2	94.4
MAE-ST$_{1600e}$ [14]	ViT-B		16	180×7×3	87	81.3	94.9
VideoMAE$_{800e}$ [35]	ViT-B	Kinetics-400 w/o label	16	180×5×3	87	80.0	94.4
VideoMAE$_{1600e}$ [35]	ViT-B		16	180×5×3	87	81.5	**95.1**
MGMAE$_{800e}$	ViT-B	Kinetics-400 w/o label	16	180×5×3	87	81.2	94.9
MGMAE$_{1600e}$	ViT-B		16	180×5×3	87	**81.8**	95.0

Table 5: **Comparison on the Kinetics-400 dataset**. We only list the results obtained with the similar backbones.

For a fair comparison, we mainly list the results with similar computational cost. On the Something-Something V2 dataset, our MGMAE with ViT-B backbone achieves a performance of 72.3% when trained for 2400 epochs, which outperforms the original VideoMAE by 1.5%. On the Kinetics-400 dataset, our MGMAE obtains slightly better performance than the original VideoMAE. The small performance improvement might be ascribed to the fact that Kinetics-400 is a scene-centric action recognition benchmark and motion information is less important compared with the Something-Something dataset.

5. Conclusion

In this paper, we have proposed the Motion Guided Masked Autoencoders (MGMAE), which adapts the motion guided masking strategy to dynamically sample the unmasked visible tokens under the guidance of flows, thus suppressing information leakage to build a more challenging task for masked video pre-training. Experiments have shown that MGMAE has good performance and maintains a high-performance advantage over the previous methods under fair comparison. In addition, our strategy also reduces the risk of pre-training overfitting, which allows the model to benefit from longer pre-training.

References

[1] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. In *IEEE/CVF International Conference on Computer Vision*, 2021. 1, 9

[2] Wele Gedara Chaminda Bandara, Naman Patel, Ali Gholami, Mehdi Nikkhah, Motilal Agrawal, and Vishal M Patel. Adamae: Adaptive masking for efficient spatiotemporal learning with masked autoencoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14507–14517, 2023. 3

[3] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. BEit: BERT pre-training of image transformers. In *International Conference on Learning Representations*, 2022. 2

[4] Wenbo Bao, Wei-Sheng Lai, Chao Ma, Xiaoyun Zhang, Zhiyong Gao, and Ming-Hsuan Yang. Depth-aware video frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3703–3712, 2019. 3

[5] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *International Conference on Machine Learning*, 2021. 1, 3, 9

[6] Kelvin CK Chan, Shangchen Zhou, Xiangyu Xu, and Chen Change Loy. Basicvsr++: Improving video super-resolution with enhanced propagation and alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5972–5981, 2022. 3

[7] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *International Conference on Machine Learning*, 2020. 2

[8] Yutao Cui, Cheng Jiang, Limin Wang, and Gangshan Wu. Mixformer: End-to-end tracking with iterative mixed attention. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 1

[9] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), 20-26 June 2005, San Diego, CA, USA*, pages 886–893. IEEE Computer Society, 2005. 2

[10] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255. IEEE Computer Society, 2009. 1

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics*, 2019. 1, 2

[12] 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. 1, 2

[13] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. In *IEEE/CVF International Conference on Computer Vision*, 2021. 9

[14] Christoph Feichtenhofer, Haoqi Fan, Yanghao Li, and Kaiming He. Masked autoencoders as spatiotemporal learners. In *Advances in Neural Information Processing Systems*, 2022. 1, 2, 4, 5, 7, 9

[15] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *IEEE/CVF International Conference on Computer Vision*, 2019. 9

[16] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fründ, Peter Yianilos, Moritz Mueller-Freitag, Florian Hoppe, Christian Thurau, Ingo Bax, and Roland Memisevic. The "something something" video database for learning and evaluating visual common sense. In *IEEE/CVF International Conference on Computer Vision*, 2017. 2, 6

[17] 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*, 2022. 1, 2

[18] Zhewei Huang, Tianyuan Zhang, Wen Heng, Boxin Shi, and Shuchang Zhou. Rife: Real-time intermediate flow estimation for video frame interpolation. *arXiv preprint arXiv:2011.06294*, 2020. 3

[19] Sunil Hwang, Jaehong Yoon, Youngwan Lee, and Sung Ju Hwang. Efficient video representation learning via masked video modeling with motion-centric token selection. *arXiv preprint arXiv:2211.10636*, 2022. 3

[20] Will Kay, Joao 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 preprint arXiv:1705.06950*, 2017. 2, 6

[21] Heeseung Kwon, Manjin Kim, Suha Kwak, and Minsu Cho. Motionsqueeze: Neural motion feature learning for video understanding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVI 16*, pages 345–362. Springer, 2020. 3

[22] Yan Li, Bin Ji, Xintian Shi, Jianguo Zhang, Bin Kang, and Limin Wang. TEA: temporal excitation and aggregation for action recognition. In *CVPR*, pages 906–915, 2020. 3

[23] 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 *IEEE/CVF International Conference on Computer Vision*, 2021. 1

[24] Zhaoyang Liu, Donghao Luo, Yabiao Wang, Limin Wang, Ying Tai, Chengjie Wang, Jilin Li, Feiyue Huang, and Tong Lu. TEINet: Towards an efficient architecture for video recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020. 9

[25] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 9

[26] Zhaoyang Liu, Limin Wang, Wayne Wu, Chen Qian, and Tong Lu. Tam: Temporal adaptive module for video recognition.

In *IEEE/CVF International Conference on Computer Vision*, 2021. 9

[27] Deepak Pathak, Philipp Krähenbühl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. Context encoders: Feature learning by inpainting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016. 2

[28] Mandela Patrick, Dylan Campbell, Yuki M. Asano, Ishan Misra, Florian Metze, Christoph Feichtenhofer, Andrea Vedaldi, and João F. Henriques. Keeping your eye on the ball: Trajectory attention in video transformers. In *NeurIPS*, pages 12493–12506, 2021. 3, 9

[29] Zhiwu Qing, Shiwei Zhang, Ziyuan Huang, Xiang Wang, Yuehuan Wang, Yiliang Lv, Changxin Gao, and Nong Sang. Mar: Masked autoencoders for efficient action recognition. *IEEE Transactions on Multimedia*, 2023. 2

[30] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 2019. 2

[31] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. *Advances in Neural Information Processing Systems*, 2014. 3

[32] 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. 2, 9

[33] Jing Tan, Jiaqi Tang, Limin Wang, and Gangshan Wu. Relaxed transformer decoders for direct action proposal generation. In *IEEE/CVF International Conference on Computer Vision*, 2021. 1

[34] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision*, pages 402–419. Springer, 2020. 2, 5, 6

[35] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Video-MAE: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In *Advances in Neural Information Processing Systems*, 2022. 1, 2, 4, 5, 6, 7, 9

[36] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, 2021. 1

[37] Du Tran, Heng Wang, Matt Feiszli, and Lorenzo Torresani. Video classification with channel-separated convolutional networks. In *IEEE/CVF International Conference on Computer Vision*, 2019. 9

[38] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017. 1

[39] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *International Conference on Machine Learning*, 2008. 2

[40] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, 2010. 2

[41] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14549–14560, 2023. 1, 2

[42] Limin Wang, Yu Qiao, and Xiaoou Tang. Action recognition with trajectory-pooled deep-convolutional descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 4305–4314. IEEE Computer Society, 2015. 3

[43] Limin Wang, Zhan Tong, Bin Ji, and Gangshan Wu. TDN: Temporal difference networks for efficient action recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 3, 9

[44] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks for action recognition in videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019. 3

[45] 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*, 2022. 2, 9

[46] 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 *IEEE/CVF International Conference on Computer Vision*, 2021. 1

[47] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018. 9

[48] Chen Wei, Haoqi Fan, Saining Xie, Chao-Yuan Wu, Alan Yuille, and Christoph Feichtenhofer. Masked feature prediction for self-supervised visual pre-training. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 2, 9

[49] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. In *Advances in Neural Information Processing Systems*, 2021. 1

[50] 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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 2

[51] Xitong Yang, Xiaodong Yang, Sifei Liu, Deqing Sun, Larry Davis, and Jan Kautz. Hierarchical contrastive motion learning for video action recognition. *arXiv preprint arXiv:2007.10321*, 2020. 3

[52] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Francis EH Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In *IEEE/CVF International Conference on Computer Vision*, 2021. 1

[53] Christopher Zach, Thomas Pock, and Horst Bischof. A duality based approach for realtime tv-l 1 optical flow. In

Pattern Recognition: 29th DAGM Symposium, Heidelberg, Germany, September 12-14, 2007. Proceedings 29, pages 214–223. Springer, 2007. 5

[54] Chen-Lin Zhang, Jianxin Wu, and Yin Li. Actionformer: Localizing moments of actions with transformers. *CoRR*, abs/2202.07925, 2022. 1

[55] Guozhen Zhang, Yuhan Zhu, Haonan Wang, Youxin Chen, Gangshan Wu, and Limin Wang. Extracting motion and appearance via inter-frame attention for efficient video frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5682–5692, 2023. 3

[56] Kaidong Zhang, Jingjing Fu, and Dong Liu. Flow-guided transformer for video inpainting. In *European Conference on Computer Vision*, pages 74–90. Springer, 2022. 3

[57] Yue Zhao, Yuanjun Xiong, and Dahua Lin. Trajectory convolution for action recognition. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 2208–2219, 2018. 3

[58] Yuan Zhi, Zhan Tong, Limin Wang, and Gangshan Wu. Mgsampler: An explainable sampling strategy for video action recognition. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 1493–1502. IEEE, 2021. 3

[59] Daquan Zhou, Bingyi Kang, Xiaojie Jin, Linjie Yang, Xiaochen Lian, Qibin Hou, and Jiashi Feng. Deepvit: Towards deeper vision transformer. *arXiv preprint arXiv:2103.11886*, 2021. 1