Selective Structured State-Spaces for Long-Form Video Understanding

Jue Wang Wentao Zhu Pichao Wang Xiang Yu Linda Liu Mohamed Omar Raffay Hamid
Amazon Prime Video
{juewangn, zhuwent, wpichao, xiangnyu, lindliu, omarmk, raffay}@amazon.com

Abstract

Effective modeling of complex spatiotemporal dependencies in long-form videos remains an open problem. The recently proposed Structured State-Space Sequence (S4) model with its linear complexity offers a promising direction in this space. However, we demonstrate that treating all image-tokens equally as done by S4 model can adversely affect its efficiency and accuracy. To address this limitation, we present a novel Selective S4 (i.e., S5) model that employs a lightweight mask generator to adaptively select informative image tokens resulting in more efficient and accurate modeling of long-term spatiotemporal dependencies in videos. Unlike previous mask-based token reduction methods used in transformers, our S5 model avoids the dense self-attention calculation by making use of the guidance of the momentum-updated S4 model. This enables our model to efficiently discard less informative tokens and adapt to various long-form video understanding tasks more effectively. However, as is the case for most token reduction methods, the informative image tokens could be dropped incorrectly. To improve the robustness and the temporal horizon of our model, we propose a novel long-short masked contrastive learning (LSMCL) approach that enables our model to predict longer temporal context using shorter input videos. We present extensive comparative results using three challenging long-form video understanding datasets (LVU, COIN and Breakfast), demonstrating that our approach consistently outperforms the previous state-of-the-art S4 model by up to 9.6% accuracy while reducing its memory footprint by 23%.

1. Introduction

Video understanding is an active research area where a variety of different models have been explored including *e.g.*, two-stream networks [19, 20, 52], recurrent neural networks [3, 63, 72] and 3-D convolutional networks [59–61]. However, most of these methods have primarily focused on short-form videos that are typically with a few seconds in length, and are not designed to model the complex long-term spatiotemporal dependencies often found in long-form videos (see Figure 1 for an illustrative example). The recent vision transformer (ViT) [14] has shown promising capability in modeling long-range dependencies, and several variants [1,4,15,41,45,49,65] have successfully adopted the transformer architecture for video modeling. However, for a video with T frames and S spatial tokens, the complexity of standard video transformer architecture is $\mathcal{O}(S^2T^2)$, which poses prohibitively high computation and memory costs when modeling long-form videos. Various attempts [54,68] have been proposed to improve this efficiency, but the ViT pyramid architecture prevents them from developing long-term dependencies on low-level features.

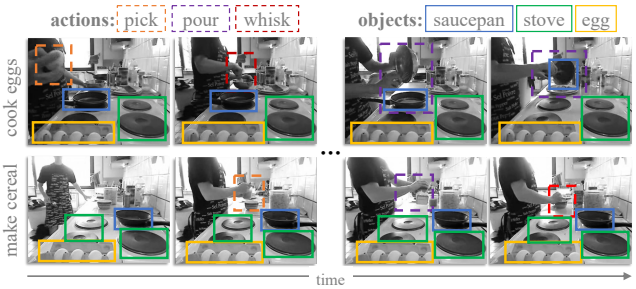

Figure 1. **Illustration of long-form videos –** Evenly sampled frames from two long-form videos, that have long duration (more than 1 minute) and distinct categories in the Breakfast [36] dataset (grayscale frames are shown for better visualization). The video on top shows the activity of making scrambled eggs, while the one on the bottom shows the activity of making cereal. These two videos heavily overlap in terms of objects (*e.g.*, eggs, saucepan and stove), and actions (*e.g.*, picking, whisking and pouring). To effectively distinguish these two videos, it is important to model long-term spatiotemporal dependencies, which is also the key in long-form video understanding.

In addition to ViT, a recent ViS4mer [29] method has tried to apply the Structured State-Spaces Sequence (S4) model [23] as an effective way to model the long-term video dependencies. However, by introducing simple masking techniques we empirically reveal that the S4 model can have different temporal reasoning preferences for different downstream tasks. This makes applying the same image token selection method as done by ViS4mer [29] for all long-

form video understanding tasks suboptimal.

To address this challenge, we propose a cost-efficient adaptive token selection module, termed S5 (*i.e.*, selective S4) model, which adaptively selects informative image tokens for the S4 model, thereby learning discriminative long-form video representations. Previous token reduction methods for efficient image transformers [37, 42, 50, 66, 70, 71] heavily rely on a dense self-attention calculation, which makes them less effective in practice despite their theoretical guarantees about efficiency gains. In contrast, our S5 model avoids the dense self-attention calculation by leveraging S4 features in a gumble-softmax sampling [30] based mask generator to adaptively select more informative image tokens. Our mask generator leverages S4 feature for its global sequence-context information and is further guided by the momentum distillation from the S4 model.

To further improve the robustness and the temporal predictability of our S5 model, we introduce a novel long-short mask contrastive learning (LSMCL) to pre-train our model. In LSMCL, randomly selected image tokens from long and short clips include the scenario that the less informative image tokens are chosen, and the representation of them are learned to match each other. As a result, the LSMCL not only significantly boosts the efficiency compared to the previous video contrastive learning methods [17, 51, 64], but also increases the robustness of our S5 model when dealing with the mis-predicted image tokens. We empirically demonstrate that the S5 model with LSMCL pre-training can employ shorter-length clips to achieve on-par performance with using longer-range clips without incorporating LSMCL pre-training.

We summarize our **key contributions** as the following:

• We propose a Selective S4 (S5) model that leverages the global sequence-context information from S4 features to adaptively choose informative image tokens in a task-specific way.

• We introduce a novel long-short masked contrastive learning approach (LSMCL) that enables our model to be tolerant to the mis-predicted tokens and exploit longer duration spatiotemporal context by using shorter duration input videos, leading to improved robustness in the S5 model.

• We demonstrate that two proposed novel techniques (S5 model and LSMCL) are seamlessly suitable and effective for long-form video understanding, achieving the state-of-the-art performance on three challenging benchmarks. Notably, our method achieves up to **9.6%** improvement on LVU dataset compared to the previous state-of-the-art S4 method, while reducing the memory footprint by **23%**.

2. Related Work

We discuss the literature with respect to the three most relevant fields: video understanding with long-form format, efficient token selection for vision transformer training, and self-supervised learning with videos.

a. Long-Form Video Modeling: Transformers have shown excellent performance in modeling long-term dependencies, *e.g.*, in natural language processing (NLP) [5, 12, 13]. But the high computational cost caused by dense self-attention calculation becomes a bottleneck to apply in not only NLP but also computer vision. Much subsequent work [11, 31, 33, 40, 41, 48, 65] focuses on improving the transformer efficiency. However, they are not designed for dealing with plethora of spatial and temporal image tokens that are common in long-form video scenarios. LF-VILA [54] develops a hierarchical feeding architecture to include more frames in the model, thus capturing longer temporal information. Similarly, MeMViT [68] better utilizes temporal information by emerging the previously cached "memory" from the past. The pyramid structure leveraged by LF-VILA and MeMViT shows efficiency improvements, but may lose low-level spatial-temporal contextual information. Gu et al. [23] proposed a Structured State-Space Sequence (S4) model, a novel alternative to CNNs or transformers, to model the long-range dependencies by simulating a linear time invariant (LTI) system. Subsequently, S4ND [46] and ViS4mer [29] extend S4 model to the video classification task. ViS4mer [29] stacks multiple S4 layers with different scales in modeling long-form videos, and S4ND [46] substitutes the traditional convolutional layer with the proposed S4ND layer in image and short-form video classification tasks. The equal importance assumption to all the image tokens by ViS4mer and S4ND can be further improved by introducing suitable token selection mechanisms, especially when dealing with the long-form input sequences. Consequently, we propose a token Selection S4 (S5) model to further enhance the efficiency while maintaining the long-form representation power.

b. Adaptive Token Selection: Adaptive token selection is widely used to improve model efficiency. Traditional CNN methods such as SCsampler [34] filter informative clips by using motion and audio embeddings. Adaframe [69] utilizes memory-augmented LSTMs as agents, which predict where to look in the next time step. AR-NET [43] uses LSTM as decision maker to select useful frames and their resolutions. [37, 42, 50, 66, 70] apply this selection idea to transformers to adaptively select tokens for increased efficiency. For instance, STTS [66] leverages a token selection module, the named scorer network, to provide the importance score for each token and select the top-K frames with the highest scores. AdaViT [42] extends this idea to develop instance-specific policies, guiding the activation of patches, self-attention heads and transformer blocks. All of the above methods demonstrate how a light-weight token selection module can improve inference efficiency. However,

these methods are essentially designed for images, and may require non-trivial adaptation to the long-form video scenarios, *i.e.*, the video-level long-range reasoning and computationally expensive self-attention calculation. To avoid this dense self-attention calculation, our proposed S5 model leverages S4 features to model the long-term dependencies and adaptively pick informative tokens.

c. Video Self-Supervised Learning (SSL): Previous work on token reduction rarely considers the negative impact of mis-dropped tokens. EViT [37] simply fuses the unattended tokens and concatenates with the remaining ones. From the recent successful image SSL works [8, 9, 21, 25, 26], many follow-up works [16, 18, 51, 58, 64] learn discriminative video features with great generalization ability in downstream tasks. Specifically, LSTCL [64] and BraVe [51] utilize long and short clips in the concept of SSL, which enables the model to learn an effective representation by predicting temporal context captured from a longer temporal extent. This essentially broadens the temporal horizon of the model for predicting longer temporal context with fewer from shorter input frames. In this paper, we adopt this idea with an additional random masking strategy to increase the efficiency of contrastive learning in long-form videos, and to further improve the robustness and the temporal predictability of our S5 model in downstream tasks.

3. Approach

We start by summarizing Structured State-Space Sequence (S4) [23] model and ViS4mer [29] (§ 3.1), followed by empirical analysis of S4 model in various long-form video understanding tasks (§ 3.2), and then providing the details of our proposed approach to address these limitations (§ 3.3 and § 3.4).

3.1. Preliminaries

3.1.1 S4 Model

Recall that a simple State-Space Model *i.e.*, a linear time invariant (LTI) system can be written as:

$$\begin{aligned} \mathbf{x}'(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t). \end{aligned} \tag{1}$$

Under deep learning setting, $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ are learned via gradient descent while $+\mathbf{D}\mathbf{u}(t)$ is replaced by a residual connection. This formulation projects an input signal $\mathbf{u}(t)$ from one-dimensional space to an N-dimensional latent space $\mathbf{x}(t)$, which is then mapped back to a one-dimensional output signal $\mathbf{y}(t)$. Similar to RNNs, it has been found in previous work that Equation 1 also suffers from gradient vanish or exploding issues when modeling longer sequences. To tackle this issue, the work in [23] leveraged HiPPO theory [22] to initialize the $\mathbf{A}$ matrix. HiPPO specifies a certain expression of $\mathbf{A} \in \mathbb{R}^{\mathrm{N}\times\mathrm{N}}$ (see Equation 2), which allows the hidden state to memorize the input $\mathbf{u}(t)$ [1].

$$\text{HiPPO: } \mathbf{A}_{n,k} = -\begin{cases} (2n+1)^{0.5}(2k+1)^{0.5} & \text{if } n > k \\ n+1 & \text{if } n = k \\ 0 & \text{if } n < k, \end{cases} \tag{2}$$

where n and k indicate the row and column indices of $\mathbf{A}$. To implement Equation 1 using discrete inputs such as word or image tokens, the work in [23] leverages the bi-linear discretization method [62] and a discretized version of Equation 1 using a step size Δ is rewritten as:

$$\begin{aligned} \mathbf{x}_k &= \bar{\mathbf{A}}\mathbf{x}_{k-1} + \bar{\mathbf{B}}\mathbf{u}_k \\ \mathbf{y}_k &= \bar{\mathbf{C}}\mathbf{x}_k, \end{aligned} \tag{3}$$

where $\bar{\mathbf{A}} = (\mathbf{I} + \frac{\Delta\cdot\mathbf{A}}{2})/(\mathbf{I} - \frac{\Delta\cdot\mathbf{A}}{2})$, $\bar{\mathbf{B}} = \Delta\cdot\mathbf{B}/(I - \frac{\Delta\cdot\mathbf{A}}{2})$ and $\bar{\mathbf{C}} = \mathbf{C}$. Equation 3 can be solved using a discrete convolution [23]:

$$\mathbf{y} = \bar{\mathbf{K}} \circledast \mathbf{u}, \tag{4}$$

where $\mathbf{u} = \{u_0, u_1, \ldots, u_{k-1}, u_k\}$ and $\bar{\mathbf{K}} \in \mathbb{R}^{\mathrm{L}} := \{\bar{\mathbf{C}}\bar{\mathbf{B}}, \bar{\mathbf{C}}\bar{\mathbf{A}}\bar{\mathbf{B}}, \ldots, \bar{\mathbf{C}}\bar{\mathbf{A}}^{\mathrm{L}-1}\bar{\mathbf{B}}\}$ is a structured convolutional kernel and L is the sequence length. Equation 4 is the core formulation of S4 model whose computational cost is linear to the input length and can be efficiently computed using fast Fourier transform (FFT) and inverse FFT. Moreover, to control the convolution kernel width, the work in [24] set Δ as a learnable parameter.

3.1.2 ViS4mer Model

By utilizing the S4 model, the ViS4mer [29] achieves promising results in the long-form video understanding tasks. We start with defining some notations to help summarize the adaptation of S4 model in computer vision. Given a video clip $\mathbf{X} \in \mathbb{R}^{\mathrm{H}\times\mathrm{W}\times 3\times\mathrm{T}}$ consisting of T RGB frames sampled from the video, we convert it into a sequence of $\mathrm{S}\cdot\mathrm{T}$ image tokens $\mathbf{x}_s^t \in \mathbb{R}^{\mathrm{D}}$ for $s = 1, \ldots, \mathrm{S}$ and $t = 1, \ldots, \mathrm{T}$. The tokens $\mathbf{z}_s^t$ are obtained by decomposing each frame into S patches which are then projected to a D-dimensional space through a learnable linear transformation. This tokenization can be implemented by linearly mapping the RGB patches of each frame [4, 45]. Separate learnable positional encodings $\mathbf{e}_s$ and $\mathbf{e}^t$ are then applied to the patch embeddings $\mathbf{z}_s^t$ for the spatial and the temporal dimensions: $\mathbf{x}_s^t = \mathbf{z}_s^t + \mathbf{e}_s + \mathbf{e}^t$, formulating $\mathbf{x}_{\mathbf{input}} = \{x_0^0, x_1^0, x_{\mathrm{S}}^0, x_0^1, \ldots, x_{\mathrm{S}}^{\mathrm{T}}\}$.

In ViS4mer [29], a multi-scale S4 decoder is introduced for learning the long-term temporal reasoning. As is mentioned in § 3.1.1, S4 model has a linear computation and

[1] Please refer to [22] for more details and relevant proofs.

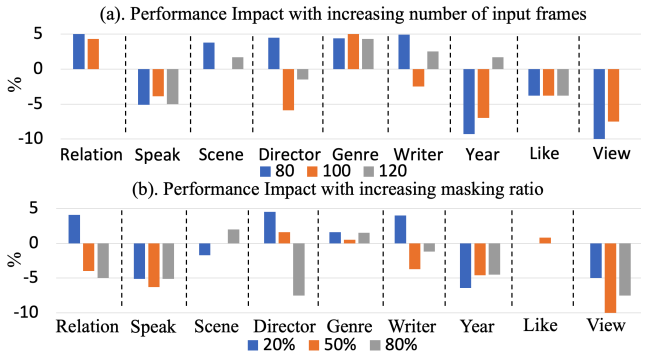

Figure 2. Performance gain/loss of ViS4mer on LVU dataset [67] with different settings of input frames and random masking ratio, where we conclude: **(a).** The performance is not substantially improved with increasing number of input frames. **(b).** Random masking strategy cannot effectively reduce redundant tokens.

memory dependency with respect to the input length, which has significantly lower computational cost than the self-attention in transformers. The formulation of S4 decoder can be written as:

$$\begin{aligned} \mathbf{x}_{s_4} &= \mathrm{S}_4\left(\mathrm{LN}\left(\mathbf{x}_{input}\right)\right) \\ \mathbf{x}_{mlp} &= \mathrm{MLP}\left(\mathrm{P}\left(\mathbf{x}_{s_4}\right)\right) \\ \mathbf{x}_{skip} &= \mathrm{Linear}\left(\mathrm{P}\left(\mathbf{x}_{input}\right)\right) \\ \mathbf{x}_{out} &= \mathbf{x}_{skip} + \mathbf{x}_{mlp}, \end{aligned} \tag{5}$$

Where $\mathrm{LN}(\cdot), \mathrm{MLP}(\cdot), \mathrm{Linear}(\cdot)$ and $\mathrm{P}(\cdot)$ represent the layer normalization [2], the multi-layer perception, linear layer and pooling layer, and $\mathbf{x}_{s_4}$ is the $\mathbf{y}$ in Equation 4.

3.2. S4 Model in Long-form Video Understanding

To better understand the S4 model and long-form video understanding tasks, we re-implement ViS4mer [29] with different settings on LVU dataset [67] and demonstrate the result in Figure 2. From the observation that short-form video understanding tasks often benefit from longer input clips [4, 15, 41, 64], we wonder if the performance of S4 model on different long-form video tasks would also be substantially improved with the increasing number of input frames. In Figure 2 (a), we gradually increase the temporal extent from 60 seconds to 120 seconds. Compared to the performance of using 60 second input, we report the impact ratio of using 80, 100, 120 second inputs in each task. From this Figure, we realize that not all long-form video tasks benefit from longer input context, and for those improved tasks, the performance is not necessarily improved with the longer input content. As a result, we raise the hypothesis that capturing long-term relationships is task- and data-dependent, and that additional performance improvements for those temporally-intensive tasks would also be hindered by the redundant spatiotemporal tokens produced by longer input content. Recalling Equation 3 and 4, each output token from S4 model is the result of structured discrete convolution for all previous inputs. Thus, we argue that treating all input token equally as ViS4mer [29] does is not appealing for S4 model to capture effective long-term dependencies, as not all tokens have the temporal relations and each task may also favor tokens in different space-time locations. To naively reduce the redundant tokens, we generate random masks on the 60 second input clips to drop tokens and increase the masking ratio from 20% to 80%. Compared to the performance of un-masked input, we report the impact ratio of using random mask with masking ratio of 20%, 50% and 80% in Figure 2 (b). Despite the minor improvement in some tasks, random masking degenerates the performance of most tasks, so it is not an effective method for reducing the redundancies. To this end, we are motivated to propose a selective S4 model which adaptively pick discriminative image tokens for the S4 model in different long-form video understanding tasks.

3.3. Adaptive Token in Long-form Videos

To pick out discriminative image tokens from the long-form videos among various tasks, we extend the concept of adaptive token learning, formulating our Selective S5 (*i.e.*, selective S4) model. Unlike previous image-based adaptive token learning works [37, 42, 50, 70] that rely on dense self-attention for capturing token-wise relationships, our S5 model avoids the self-attention computation in long-form videos by leveraging S4 features generated from the simulated linear time-invariant (LTI) system. Inherited from the linear complexity of the S4 model, our S5 model can receive long-form video token dependencies with low cost, thus making the adaptive token learning possible in long-form videos. In addition, we propose a momentum updated S4 model to dynamically produce S4 features from the long-form video data in different tasks. Figure 3 (a) demonstrates the pipeline of our S5 model, where the momentum updated S4 model is the moving average of the S4 model.

Specifically, we cast our selective module in the S5 model as an adaptive mask learning problem. Given a mask generator $\mathrm{MG}(\cdot)$ and its input $\mathbf{x}_{s_4}$, the mask generator is a lightweight architecture, which will be ablated in the Section 4. It will be trained for a classification task on predefined category space $\mathbb{C} = \{C_1, \ldots, C_{\mathrm{ST}}\}$, where $\mathrm{S} \cdot \mathrm{T}$ is the total number of image tokens in the video. Let's denote $p(c|\mathbf{x}_{s_4}) \in [0, 1]$ be the normalized probabilistic output of $MG(\mathbf{x}_{s_4})$, so that $\sum_{c=C_1}^{c=C_{\mathrm{ST}}} p(c|\mathbf{x}_{s_4}) = 1$. Then, we sample K categories without replacement from the probabilistic outputs of the mask generator. Finally, the k^{th} selected image tokens can be written as:

$$x_{\mathrm{in}}^{k} = \mathbf{X}^{\mathbf{T}} c^{k} \tag{6}$$

Where $\mathbf{X} \in \mathbb{R}^{\mathrm{ST} \times D}$ represents $\mathrm{S} \cdot \mathrm{T}$ D-dimensional image tokens and c^k is a one-hot vector that select k^{th} token from

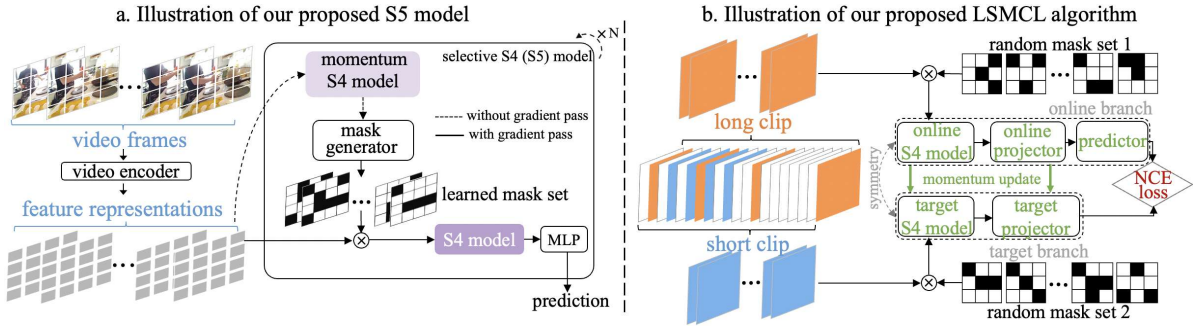

Figure 3. (a) A visualization of our proposed S5 model. Compared to the S4 model, we introduce a selective token picking strategy "mask generator", leverageing the S4 feature from the momentum S4 model. The momentum S4 model is updated by the S4 model in the moving average manner. Both S4 model and momentum S4 model are consisted of a S4 layer [23, 29] and a LN layer [2]. (b) An illustration of the proposed LSMCL pretraining framework, that initializes our S5 model to enrich the robustness.

the $\mathbf{X}$. The sampling process is important as it prevents the bias in the training that is potentially caused by the top-K selection. To make this sampling differentiable, we adopt the Gumbel-Softmax with Straight-Through tricks [30], which is widely used in [38, 42]. Specifically, we introduce an additional gumbel noise $g \in \mathbb{R}^{1\times \mathrm{ST}}$ into the predicted probability distribution $p \in \mathbb{R}^{1\times \mathrm{ST}}$, where $g = -\log(-\log(u + \epsilon) + \epsilon)$ ($u \sim$ Uniform(0,1) , and ϵ is a small value for arithmetic robustness consideration). Then, we sample the top-K tokens from the re-parameterized distribution $p + g$. During the back-propagation, we estimate the gradient for each selected token c as:

$$G \approx \bigtriangledown_{\mathrm{MG}} \frac{\exp((\log p(c|\mathbf{x}_{s_4}) + g(c))/\rho)}{\sum_{c'=C_1}^{c'=C_{\mathrm{ST}}} \exp((\log p(c'|\mathbf{x}_{s_4}) + g(c'))/\rho)} \quad (7)$$

where ρ is the temperature factor controlling the sharpness.

3.4. Long-Short Mask Contrastive Learning

Previous token reduction/adaptive learning works rarely take model robustness into consideration. Informative tokens might be incorrectly dropped during training, which could hurt the performance of the model. In this paper, in addition to our proposed S5 model that explicitly picks informative tokens for various long-form video understanding tasks, we also propose Long-Short Mask Contrastive Learning (LSMCL) pretraining, which implicitly learns long-form video representations with better generalizability. Specifically, we equip the recent video contrastive learning framework LSTCL [64] with a random masking strategy on both long and short input clips, which mimics all possible scenarios that the selective module could produce in the S5 model. As a result, our S5 model with LSMCL pretraining would be more robust to and tolerant of errors from the selective module. Moreover, the long-short contrastive set-up will further improve the temporal predictability of our S5 model.

Formally, we sample a long clip (x_L) and a short clip (x_S) from each video sequence with largely different sampling strides τ_L and τ_S, where $\tau_S < \tau_L$. Unlike LSTCL [64] and BraVe [51] that apply independent random sampling, in our paper the temporal span of long clips includes the one of short clips, which prevents dissimilar semantics from two clips in long-form videos. Then, we independently generate binary random masks with a masking ratio of η for each clip, which can be written as: $\mathcal{R}_{\mathrm{mask}}(x, \eta), x \in \{x_L, x_S\}$. We set S4 model as the backbone of the query encoder (f_q) and also adopt a momentum key encoder (f_k) in the pipeline, which is widely accepted in MoCo [26], BYOL [21] and LSTCL [64]. Our query encoder and key encoder follow the same design with [21, 26, 64], that consist of the backbone, projection and prediction heads. Denoting the parameter of f_q as θ_q and the one of f_k as θ_k, we have: $\theta_k = m\theta_k + (1-m)\theta_q$, where $m \in [0, 1]$ is a momentum coefficient. Similarly, the LSMCL adoptes similar objective as the InfoNCE [47]:

$$\text{Given: } q = f_q(\mathcal{R}_{\mathrm{mask}}(x_S, \eta)), k = f_k(\mathcal{R}_{\mathrm{mask}}(x_L, \eta))$$

$$\mathcal{L}_{\mathrm{LSMCL}} = \sum_i -\log \frac{\exp({q^i}^\top k^i/\rho)}{\exp({q^i}^\top k^i/\rho) + \sum_{j\neq i} \exp({q^i}^\top k^j/\rho)} \quad (8)$$

where ρ is the temperature hyperparameter. As is commonly done in [6, 9, 10, 21], we symmetrize the loss function by switching x_S and x_L in f_q and f_k. In our LSMCL, the S4 model is learned to find the correct step size Δ and SSM parameters to match the representation of random masked long and short clips. Given our S5 model takes adaptively learned image tokens in the downstream task, we believe the LSMCL could improve the robustness as well as the temporal modeling ability of S5 model when dealing with partially sampled image tokens. In Section 4, our S5 model with LSMCL empirically shows significantly improved results in long-form video understanding.

Mask Generator	Content (↑)			Metadata (↑)				User (↓)		
	Relation	Speak	Scene	Director	Genre	Writer	Year	Like	View	
No Mask (ViS4mer [29])	57.14	40.79	67.44	62.61	54.71	48.80	44.75	0.26	3.63	
Random	54.81	38.22	67.44	63.60	54.97	47.00	42.70	0.25	4.00	
Single TX	57.85	40.79	68.66	63.98	55.12	48.85	43.46	0.26	3.82	
Single TX_{S4}	**60.54**	**41.21**	**69.83**	**66.43**	**57.55**	**49.47**	**44.15**	**0.25**	**3.51**	+3.4
Stacked TXs	59.51	41.21	69.83	64.91	55.12	51.83	47.55	0.25	3.42	
Stacked TXs_{S4}	**61.98**	**41.75**	**70.94**	**67.34**	**59.16**	**51.83**	**47.55**	**0.24**	**3.42**	+2.5
Linear	54.81	40.28	67.44	63.90	54.97	48.17	42.77	0.26	3.95	
$Linear_{S4}$	**61.98**	**41.75**	**69.88**	**66.40**	**58.80**	**50.60**	**47.70**	**0.25**	**3.51**	+6.7

Table 1. Performance of various mask generators in LVU [67] dataset, where we adopt 60 frames per clip and 50% masking ratio. The bold results demonstrate the performance of using S4 feature (x_{S_4} in Equation 5). We also provide the average improvement ratio (in green) of nine jobs using S4 features compared to ViT features at the conclusion of each bold row.

4. Experiments

4.1. Dataset

LVU dataset [67]: is constructed from Movie Clip dataset [55]. It contains $\sim 30K$ videos from $\sim 3K$ movies. Each video lasts one to three minutes. The benchmark contains nine tasks covering a wide range of long-form video understanding tasks, which are further folded into three main categories: (i) content understanding, consisting of ('relationship', 'speaking style', 'scene/place') prediction, (ii) metadata prediction, including ('director', 'genre', 'writer', and 'movie release year') classification, and (iii) user engagement, predicting ('YouTube like ratio', and 'YouTube popularity'). For classification and regression tasks, we report accuracy (for content understanding and metadata prediction) and mean-squared error (MSE) (for user engagement) as the evaluation metrics.

COIN [56, 57]: consists of 11,827 videos with 180 distinct procedural tasks, which are all collected from YouTube. These videos cover 12 domains, such as nursing & caring, vehicles, leisure & performance, gadgets, electric appliances, household items, science & craft, plants & fruits, snacks & drinks dishes, sports, and housework. The average length of a video is 2.36 minutes.

Breakfast [36]: contains 1,712 videos of 10 complex cooking activities, which are performed by 52 different individuals in 18 different kitchens, resulting in over 77 hours of video footage. The averaged length of video in this dataset is around 2.7 minutes. Ten cooking activities include: making coffee, chocolate milk, juice, tea, cereals, fried egg, pancakes, fruit salad, sandwich and scrambled egg.

4.2. Implementation Details

Following [29, 67], we stack three structure blocks, which share similar structure to that described in Equation 5, and sample video frames at 1 fps. Unlike previous work, we include an adaptive mask generator to effectively pick image tokens before feeding the input into S4 model. As the advantages of our S5 model will naturally be diminished on less redundant sequences, we follow the same architecture of ViS4mer [29] but adopt the S5 model as the first block. For data argumentation, we resize each video frame to the spatial resolution of 224×224 and use a patch size of 16×16. In addition, we use ViT-L [14] pretrained on ImageNet-21K [35] as the feature extractor in the LVU dataset; Swin-B [40] pretrained on Kinetics-600 [32] as the feature extractor in COIN and Breakfast datasets. The size of the input in each dataset is also the same as [29]: we adopt 60-second input for the LVU dataset and 64-second input for the COIN and Breakfast datasets. In the LSMCL, we adopt the setting from LSTCL [64] and apply independent global random masking on long and short clips, which share the same masking ratio with the adaptive mask generator. Unless otherwise noted, we conduct our ablation studies on the LVU dataset due to its diverse tasks in the long-form video understanding. Finally, we report the best performance of our model on all three datasets and compare with the previous state-of-the-art works.

4.3. Ablation Study

a. Our S5 is better than S4 and random masking: To demonstrate the effectiveness of our proposed S5 model, we compare the performance of S4 models with no mask, random mask, and mask generators of different architectures. Specifically, we utilize one Transformer (TX), two stacked Transformers (TXs), and one linear layer as the mask generator and evaluate on 9 tasks on the LVU dataset (Table 1). In addition, we also evaluate the effectiveness of using S4 features from the momentum-updated S4 model. For each architecture, we compare the result of using ViT features and S4 features as the mask generator input. As can be seen from the Table 1, the performance of each task substantially increases with the computational complexity of the mask generator. Results show our design significantly outperforms ViS4mer [29] and the random masking strategy, and the performance of each task is further improved by using S4 features. Notably, the mask generator with one

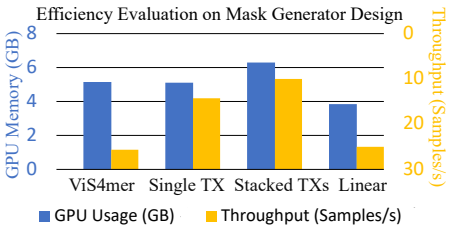

Figure 4. Efficiency evaluation of each method in Table 1, which demonstrates the GPU memory usage as well as throughput. Our proposed S5 model with linear mask generator saves 25% memory cost and achieves on par throughput with ViS4mer [29].

linear layer achieves on par performance to one of the more complex transformer architectures.

b. Our S5 reduces up to 25% memory usage: In Figure 4, we also demonstrate the efficiency of our S5 model with the different masking architectures mentioned previously. Compared to ViS4mer (the one without masking strategies) using same number of input frames, our S5 model with linear mask generator reduces the memory footprint by 25% while maintaining the same level of throughput. Memory consumption and throughput are not improved by the intricate transformer mask generators. Since the linear mask generator has a smaller memory footprint and performs tasks more effectively overall, we use it in our S5 model in the following experiments.

c. Impact of Masking Ratio and Sequence Length: In Figure 5a and 5b, we study the effect of masking ratio and sequence length with our S5 model. We set ViS4mer [29] (60 frames without mask generator) as baseline and report the average improvement percentage of 9 tasks on LVU dataset by using S5 model with variant masking ratio/sequence length. To demonstrate the effectiveness of our S5 model, we also compare the performance of ViS4mer [29] with different settings in these two figures. Figure 5a clearly shows that the performance of our S5 model increases initially as the masking ratio increases, which indicates that our selective model effectively picks informative image tokens for the S4 model. However, the performance starts to drop dramatically when the masking ratio is over 50%. This is because when the masking ratio increases to be above certain level, the informative tokens are forced to be dropped. As a result, we adopt 50% masking ratio in our following experiments. In Figure 5b, we observe substantial improvement of S5 model with increasing number of input frames. In contrast to the performance of ViS4mer [29], our proposed S5 model is indeed able to capture longer term dependencies while reducing the spatial-temporal redundancy in the input.

d. Effect of Multiple S5 models: As shown in Figure 3, multiple S5 models can be stacked in the pipeline, similar to what is commonly done in Transformer [4, 14, 68] and ViS4mer [29]. In the previous setup, we only adopt one S5 model, leaving the remaining blocks as S4 models. By stacking multiple S5 models, we find a further **0.5**% average improvement on the LVU dataset. Less redundant sequences will inevitably reduce the performance gain from our S5 model, decreasing the benefit from stacking additional S5 blocks. As a result, we utilize only one S5 model after the video encoder for maximum memory efficiency gain and throughput.

e. Ablation on LSMCL: In Figure 5c and 5d, we evaluate the effectiveness of our proposed LSMCL with different sampling strides and random masking ratios. For both figures, we set the performance of ViS4mer [29] as the baseline and report the average improvement ratio (in percentage) of 9 tasks from LVU with different settings. From Figure 5c, our S5 model with LSMCL can achieve better performance even when $\tau_L = \tau_S$, which suggests that LSMCL can increase the robustness of our S5 model and help it handle incorrectly picked tokens. When we gradually increase the $\frac{\tau_L}{\tau_S}$, the performance of S5 model is further improved as the model is able to capture longer temporal context via the proposed LSMCL. Indeed, the performace using LSMCL approaches the performance without LSMCL **with 66% more input frames** (shown in Figure 5b both around 6% boost). In Figure 5d, we further ablate the random masking ratio used in LSMCL. When the masking ratio of LSMCL is over 50%, the benefit from LSMCL is insignificant as the input does not provide sufficient information. Thus, we consider 50% masking ratio in LSMCL for better efficiency in the long-form video contrastive learning.

4.4. Comparison with the State-Of-The-Arts

In Table 2, we compare our method on LVU dataset with previous state-of-the-art methods. Specifically, the LST [29] adopt the same architecture with ours, but substitutes the S5/S4 model to the transformer architecture. Whereas the Performer [11] and Orthoformer [49] apply the efficient attention in the transformer architecture, that do not require quadratic complexity *w.r.t.* the input length. When compared to baseline ViS4mer [29], we achieve up to **9.6**% improvement. When compared to other methods, ours outperforms by an even more significant margin. This shows that our method is consistently more effective in understanding the long-form videos.

To demonstrate the generalizability of our method, we evaluate our S5 model on COIN [56, 57] and Breakfast [36] datasets, which are challenging long-range procedural activity classification datasets. Our proposed method achieves **2.4%** and **5.5%** over the ViS4mer [29] and outperforms the other state-of-the-arts by **0.81%** and **0.80%** respectively. Notice that D-Sprv. [39] leverages HowTo100M dataset [44] for pretraining, which volume is much larger

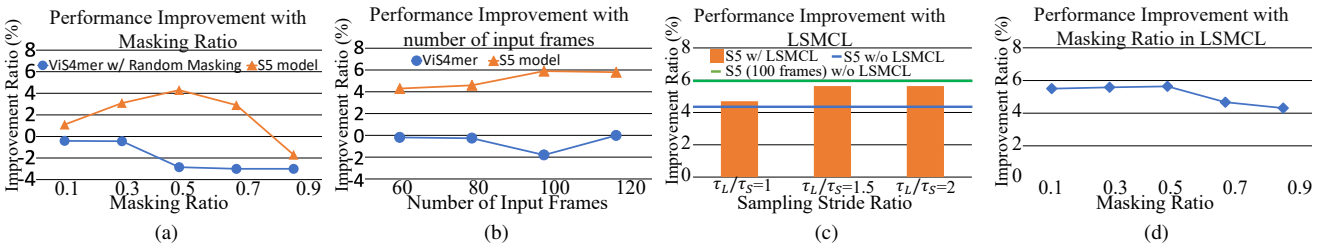

Figure 5. Compared to the baseline performance, average improvement performance of our method on LVU dataset. Unless otherwise noted, the default number of input frame and masking ratio is 60 and 50%. **(a).** We compared our S5 model and S4 model with random masking with increasing masking ratio; **(b).** We compare our S5 model and S4 model with increasing number of input frames; **(c).** We show the effect of LSMCL pretraining with different long-short sampling stride ratio. In addition, we provide the performance of S5 model without LSMCL and S5 model with 100 input frames; **(d).** We show the impact of the increasing masking ratio in the LSMCL pretraining.

Model	Content (↑)			Metadata (↑)				User (↓)		GPU Usage (GB) (↓)
	Relation	Speak	Scene	Director	Genre	Writer	Year	Like	View	
Obj. T4mer [67]	54.76	33.17	52.94	47.66	52.74	36.30	37.76	0.30	3.68	N/A
Performer [11]	50.00	38.80	60.46	58.87	49.45	48.21	41.25	0.31	3.93	5.93
Orthoformer [49]	50.00	38.30	66.27	55.14	55.79	47.02	43.35	0.29	3.86	5.56
VideoBERT [53]	52.80	37.90	54.90	47.30	51.90	38.50	36.10	0.32	4.46	N/A
LST [29]	52.38	37.31	62.79	56.07	52.70	42.26	39.16	0.31	3.83	41.38
ViS4mer [29]	57.14	40.79	67.44	62.61	54.71	48.80	44.75	0.26	3.63	5.15
Ours$_{60\ frames}$	**61.98**	**41.75**	**69.88**	**66.40**	**58.80**	**50.60**	**47.70**	**0.25**	**3.51**	**3.85**
Ours$_{60\ frames+LSMCL}$	**61.98**	**41.75**	**72.53**	**66.40**	**61.34**	**50.60**	**47.70**	**0.24**	**3.51**	**3.85**
Ours$_{100\ frames}$	**66.71**	**41.78**	**73.28**	**66.64**	**63.65**	**50.60**	**47.85**	**0.25**	**3.51**	**3.95**
Ours$_{100\ frames+LSMCL}$	**67.11**	**42.12**	**73.49**	**67.32**	**65.41**	**51.27**	**47.95**	**0.24**	**3.51**	**3.95**

Table 2. Comparison to the state-of-the-art methods on LVU dataset testing set.

Method	P.T. Dataset	P.T. Samples	Accuracy
TSN [57]	Kinetics-400	306K	73.40
D-Sprv. [39]	HowTo100M	136M	90.00
ViS4mer [29]	Kinetics-600	495K	88.41
Ours	Kinetics-600	495K	**90.42**
Ours$_{+LSMCL}$	Kinetics-600	495K	**90.81**

Table 3. Comparison to the state-of-the-art methods on COIN dataset. P.T. stands for pretraining.

Method	P.T. Dataset	P.T. Samples	Accuracy
VideoGraph [28]	Kinetics-400	306K	69.50
Timeception [27]	Kinetics-400	306K	71.30
GHRM [73]	Kinetics-400	306K	75.50
D-Sprv. [39]	HowTo100M	136M	89.90
ViS4mer [29]	Kinetics-600	495K	85.10*
Ours	Kinetics-600	495K	**90.14**
Ours$_{+LSMCL}$	Kinetics-600	495K	**90.70**

Table 4. Comparison to the state-of-the-art methods on Breakfast dataset. P.T. stands for pretraining. *We were not able to reproduce the 88.17% baseline result reported in [29], but our proposed S5 model still largely improves from 85.10%, and achieves the new state-of-the-art result.

than our pre-training dataset (Kinetics-600 [7]). Putting together the aforementioned performance gain and memory efficiency gain, our S5 model successfully demonstrates its efficiency and effectiveness in learning discriminative representation via selecting informative image tokens from long-form video sequences.

5. Conclusion

In this paper, we proposed a selective structured state-space sequence (S5) model for long-form video understanding, where we adopt a lightweight mask generator to adaptively pick informative tokens from long-form videos. Our mask generator avoids dense self-attention computation as what is applied in previous works. It leverages the sequential output of the simulated linear time invariant (LTI) system, and benefits from the momentum distillation of S4 model, enabling our S5 model to dynamically learn from informative tokens for different long-form video tasks. To mitigate the negative impact of picking less informative tokens, we also propose a LSMCL pretraining to improve the robustness and further broaden the temporal horizon of our model. Through extensive experiments, we demonstrate the effectiveness of each proposed component in our S5 model, achieving the new state-of-the-art performance in three challenging long-form video understanding benchmarks.

References

[1] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6836–6846, October 2021. 1

[2] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 4, 5

[3] Moez Baccouche, Franck Mamalet, Christian Wolf, Christophe Garcia, and Atilla Baskurt. Sequential deep learning for human action recognition. In *International workshop on human behavior understanding*, pages 29–39. Springer, 2011. 1

[4] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 813–824. PMLR, 2021. 1, 3, 4, 7

[5] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 2

[6] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *arXiv preprint arXiv:2006.09882*, 2020. 5

[7] Joao Carreira, Eric Noland, Andras Banki-Horvath, Chloe Hillier, and Andrew Zisserman. A short note about kinetics-600. *arXiv preprint arXiv:1808.01340*, 2018. 8

[8] 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

[9] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. *arXiv preprint arXiv:2011.10566*, 2020. 3, 5

[10] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised visual transformers. *arXiv preprint arXiv:2104.02057*, 2021. 5

[11] Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, et al. Rethinking attention with performers. *arXiv preprint arXiv:2009.14794*, 2020. 2, 7, 8

[12] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*, 2019. 2

[13] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 2

[14] 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. *arXiv preprint arXiv:2010.11929*, 2020. 1, 6, 7

[15] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6824–6835, October 2021. 1, 4

[16] Christoph Feichtenhofer, Haoqi Fan, Yanghao Li, and Kaiming He. Masked autoencoders as spatiotemporal learners. *arXiv preprint arXiv:2205.09113*, 2022. 3

[17] Christoph Feichtenhofer, Haoqi Fan, Bo Xiong, Ross Girshick, and Kaiming He. A large-scale study on unsupervised spatiotemporal representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3299–3309, 2021. 2

[18] Christoph Feichtenhofer, Haoqi Fan, Bo Xiong, Ross B. Girshick, and Kaiming He. A large-scale study on unsupervised spatiotemporal representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 3299–3309. Computer Vision Foundation / IEEE, 2021. 3

[19] Christoph Feichtenhofer, Axel Pinz, and Richard P Wildes. Spatiotemporal multiplier networks for video action recognition. In *CVPR*, 2017. 1

[20] Christoph Feichtenhofer, Axel Pinz, and Richard P Wildes. Temporal residual networks for dynamic scene recognition. In *CVPR*, 2017. 1

[21] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020. 3, 5

[22] Albert Gu, Tri Dao, Stefano Ermon, Atri Rudra, and Christopher Ré. Hippo: Recurrent memory with optimal polynomial projections. *Advances in Neural Information Processing Systems*, 33:1474–1487, 2020. 3

[23] Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. *arXiv preprint arXiv:2111.00396*, 2021. 1, 2, 3, 5

[24] Albert Gu, Isys Johnson, Karan Goel, Khaled Saab, Tri Dao, Atri Rudra, and Christopher Ré. Combining recurrent, convolutional, and continuous-time models with linear state space layers. *Advances in neural information processing systems*, 34:572–585, 2021. 3

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

[26] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Con-*

ference on Computer Vision and Pattern Recognition, pages 9729–9738, 2020. 3, 5

[27] Noureldien Hussein, Efstratios Gavves, and Arnold WM Smeulders. Timeception for complex action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 254–263, 2019. 8

[28] Noureldien Hussein, Efstratios Gavves, and Arnold WM Smeulders. Videograph: Recognizing minutes-long human activities in videos. *arXiv preprint arXiv:1905.05143*, 2019. 8

[29] Md Mohaiminul Islam and Gedas Bertasius. Long movie clip classification with state-space video models. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 1, 2, 3, 4, 5, 6, 7, 8

[30] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016. 2, 5

[31] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International Conference on Machine Learning*, pages 5156–5165. PMLR, 2020. 2

[32] 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. The Kinetics-400 dataset is licensed under the Creative Commons Attribution-NonCommercial 4.0 International License. 6

[33] Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer. *arXiv preprint arXiv:2001.04451*, 2020. 2

[34] Bruno Korbar, Du Tran, and Lorenzo Torresani. Scsampler: Sampling salient clips from video for efficient action recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6232–6242, 2019. 2

[35] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105, 2012. 6

[36] Hilde Kuehne, Ali Arslan, and Thomas Serre. The language of actions: Recovering the syntax and semantics of goal-directed human activities. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 780–787, 2014. 1, 6, 7

[37] Youwei Liang, Chongjian Ge, Zhan Tong, Yibing Song, Jue Wang, and Pengtao Xie. Not all patches are what you need: Expediting vision transformers via token reorganizations. *arXiv preprint arXiv:2202.07800*, 2022. 2, 3, 4

[38] Xudong Lin, Gedas Bertasius, Jue Wang, Shih-Fu Chang, Devi Parikh, and Lorenzo Torresani. Vx2text: End-to-end learning of video-based text generation from multimodal inputs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7005–7015, 2021. 5

[39] Xudong Lin, Fabio Petroni, Gedas Bertasius, Marcus Rohrbach, Shih-Fu Chang, and Lorenzo Torresani. Learning to recognize procedural activities with distant supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13853–13863, 2022. 7, 8

[40] 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. *arXiv preprint arXiv:2103.14030*, 2021. 2, 6

[41] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. *arXiv preprint arXiv:2106.13230*, 2021. 1, 2, 4

[42] Lingchen Meng, Hengduo Li, Bor-Chun Chen, Shiyi Lan, Zuxuan Wu, Yu-Gang Jiang, and Ser-Nam Lim. Adavit: Adaptive vision transformers for efficient image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12309–12318, 2022. 2, 4, 5

[43] Yue Meng, Chung-Ching Lin, Rameswar Panda, Prasanna Sattigeri, Leonid Karlinsky, Aude Oliva, Kate Saenko, and Rogerio Feris. Ar-net: Adaptive frame resolution for efficient action recognition. In *European Conference on Computer Vision*, pages 86–104. Springer, 2020. 2

[44] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2630–2640, 2019. 7

[45] Daniel Neimark, Omri Bar, Maya Zohar, and Dotan Asselmann. Video transformer network. *arXiv preprint arXiv:2102.00719*, 2021. 1, 3

[46] Eric Nguyen, Karan Goel, Albert Gu, Gordon W Downs, Preey Shah, Tri Dao, Stephen A Baccus, and Christopher Ré. S4nd: Modeling images and videos as multidimensional signals using state spaces. *Advances in neural information processing systems*, 2022. 2

[47] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 5

[48] Zizheng Pan, Bohan Zhuang, Jing Liu, Haoyu He, and Jianfei Cai. Scalable vision transformers with hierarchical pooling. In *Proceedings of the IEEE/cvf international conference on computer vision*, pages 377–386, 2021. 2

[49] Mandela Patrick, Dylan Campbell, Yuki Asano, Ishan Misra, Florian Metze, Christoph Feichtenhofer, Andrea Vedaldi, and João F Henriques. Keeping your eye on the ball: Trajectory attention in video transformers. *Advances in neural information processing systems*, 34:12493–12506, 2021. 1, 7, 8

[50] Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. Dynamicvit: Efficient vision transformers with dynamic token sparsification. *Advances in neural information processing systems*, 34:13937–13949, 2021. 2, 4

[51] Adrià Recasens, Pauline Luc, Jean-Baptiste Alayrac, Luyu Wang, Florian Strub, Corentin Tallec, Mateusz Malinowski, Viorica Patraucean, Florent Altché, Michal Valko, et al.

Broaden your views for self-supervised video learning. *arXiv preprint arXiv:2103.16559*, 2021. 2, 3, 5

[52] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. *arXiv preprint arXiv:1406.2199*, 2014. 1

[53] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7464–7473, 2019. 8

[54] Yuchong Sun, Bei Liu, Hongwei Xue, Ruihua Sone, Huan Yang, and Jianlong Fu. Long-form video-language pre-training with multimodal temporal contrastive learning. *Advances in neural information processing systems*, 2022. 1, 2

[55] Yansong Tang, Dajun Ding, Yongming Rao, Yu Zheng, Danyang Zhang, Lili Zhao, Jiwen Lu, and Jie Zhou. Coin: A large-scale dataset for comprehensive instructional video analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 6

[56] Yansong Tang, Dajun Ding, Yongming Rao, Yu Zheng, Danyang Zhang, Lili Zhao, Jiwen Lu, and Jie Zhou. Coin: A large-scale dataset for comprehensive instructional video analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1207–1216, 2019. 6, 7

[57] Yansong Tang, Jiwen Lu, and Jie Zhou. Comprehensive instructional video analysis: The coin dataset and performance evaluation. *IEEE transactions on pattern analysis and machine intelligence*, 43(9):3138–3153, 2020. 6, 7, 8

[58] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *arXiv preprint arXiv:2203.12602*, 2022. 3

[59] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015. 1

[60] Du Tran, Heng Wang, Lorenzo Torresani, and Matt Feiszli. Video classification with channel-separated convolutional networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5552–5561, 2019. 1

[61] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6450–6459, 2018. 1

[62] Arnold Tustin. A method of analysing the behaviour of linear systems in terms of time series. *Journal of the Institution of Electrical Engineers-Part IIA: Automatic Regulators and Servo Mechanisms*, 94(1):130–142, 1947. 3

[63] Vivek Veeriah, Naifan Zhuang, and Guo-Jun Qi. Differential recurrent neural networks for action recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 4041–4049, 2015. 1

[64] Jue Wang, Gedas Bertasius, Du Tran, and Lorenzo Torresani. Long-short temporal contrastive learning of video transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14010–14020, 2022. 2, 3, 4, 5, 6

[65] Jue Wang and Lorenzo Torresani. Deformable video transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14053–14062, 2022. 1, 2

[66] Junke Wang, Xitong Yang, Hengduo Li, Zuxuan Wu, and Yu-Gang Jiang. Efficient video transformers with spatial-temporal token selection. *arXiv preprint arXiv:2111.11591*, 2021. 2

[67] Chao-Yuan Wu and Philipp Krähenbühl. Towards Long-Form Video Understanding. In *CVPR*, 2021. 4, 6, 8

[68] Chao-Yuan Wu, Yanghao Li, Karttikeya Mangalam, Haoqi Fan, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. Memvit: Memory-augmented multiscale vision transformer for efficient long-term video recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13587–13597, 2022. 1, 2, 7

[69] Zuxuan Wu, Caiming Xiong, Chih-Yao Ma, Richard Socher, and Larry S Davis. Adaframe: Adaptive frame selection for fast video recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1278–1287, 2019. 2

[70] Hongxu Yin, Arash Vahdat, Jose Alvarez, Arun Mallya, Jan Kautz, and Pavlo Molchanov. Adavit: Adaptive tokens for efficient vision transformer. *arXiv preprint arXiv:2112.07658*, 2021. 2, 4

[71] Hongxu Yin, Arash Vahdat, Jose M Alvarez, Arun Mallya, Jan Kautz, and Pavlo Molchanov. A-vit: Adaptive tokens for efficient vision transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10809–10818, 2022. 2

[72] Pengfei Zhang, Cuiling Lan, Junliang Xing, Wenjun Zeng, Jianru Xue, and Nanning Zheng. View adaptive recurrent neural networks for high performance human action recognition from skeleton data. In *Proceedings of the IEEE international conference on computer vision*, pages 2117–2126, 2017. 1

[73] Jiaming Zhou, Kun-Yu Lin, Haoxin Li, and Wei-Shi Zheng. Graph-based high-order relation modeling for long-term action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8984–8993, 2021. 8