This ICCV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Make Your Training Flexible: Towards Deployment-Efficient Video Models

Chenting Wang[1,2] Kunchang Li[2]
Tianxiang Jiang[2,4] Xiangyu Zeng[2,5] Yi Wang[2,3] Limin Wang[1,2,5♠]

[1]Shanghai Jiao Tong University [2]Shanghai AI Laboratory [3]Shanghai Innovation Institute
[4]University of Science and Technology of China [5]State Key Laboratory for Novel Software Technology, Nanjing University

Abstract

Current video training methods rely on fixed spatiotemporal sampling grids to extract a predetermined number of tokens, limiting adaptability to diverse computational budgets and resulting in suboptimal accuracy-computation trade-offs. This rigidity constrains high-performance models trained in resource-rich environments from being efficiently deployed on resource-constrained devices. We hence introduce a novel paradigm for lossless adaptation across scenarios, enabling models to maintain optimal performance under high-resource conditions while seamlessly transferring to low-resource environments. Central to this is ***Token Optimization (TO)****, an adaptive inference framework that dynamically samples and selects input token set to optimize input information under varied computational constraints. To support this, we propose* ***Flux****, an augmentation tool that enables flexible sampling grids and token selection. It integrates seamlessly into popular video training frameworks, significantly enhancing model robustness and adaptability with negligible additional cost. Applied to large-scale video pretraining, our method produces* ***FluxViT****, which achieves state-of-the-art performance across multiple tasks under standard costs. Remarkably, with only 1/4 of the tokens, FluxViT matches prior state-of-the-art models under TO across tasks, achieving nearly 90% computational savings. Code and models are available at* https://github.com/OpenGVLab/FluxViT.

1. Introduction

Video representation learning [20, 45, 71] is a fundamental task in computer vision, as it is critical in developing spatiotemporal perception and procedural reasoning in multimodal large language models [38, 41, 54, 77] and embodied AI systems [24, 78]. However, the deployment of video models is computationally intensive due to the high volume of video tokens compared to image tokens. Popular video training methods work on fixed-count patched tokens sampled from a predefined spatiotemporal grid (e.g., 8×224^2), leading to considerable redundancy in training and deployment, especially for long, high-resolution videos. Moreover, this inflexible and fixed training achieves sub-optimal results for tasks that require dynamic resolution settings, including restricted computation or emphasizing frame counts rather than frame resolution. Researchers have developed different approaches for better trade-offs.

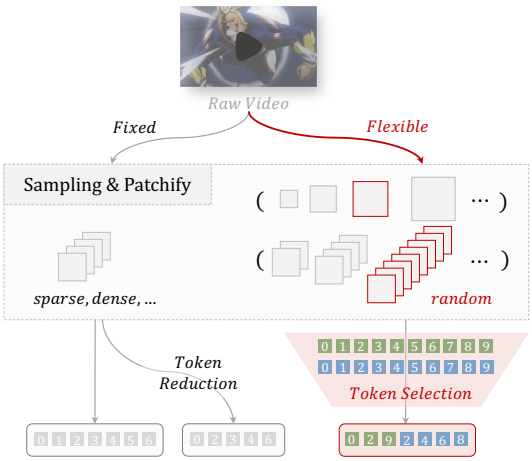

Figure 1. **Flux** (right) employs flexible sampling and token selection to achieve Token Optimization. Common methods(left) use rigid sampling(and use token reduction for applications directly).

First, to fit a limited computation budget in inference, many works [15, 31, 59] use token reduction on the same densely sampled tokens as in training, showing sub-optimal performance. For one thing, there is a trade-off between the complexity (cost) and performance of the token reduction strategy. Current strategies can not perform well with a significant reduction rate. For another, the inflexibly trained models can not generalize with the sparse, masked tokens.

Second, existing methods for flexible network training, such as Resformer [65] and FFN [88], have demonstrated the efficacy of flexible networks operating at different spatial or temporal resolutions but not both simultaneously. Al-

♠ Corresponding author.

though these approaches achieve robust performance across diverse settings, we contend that merely reducing frame counts or resolution under computational constraints fails to optimize token capacity utilization and remains susceptible to redundancy. Notably, FFN exhibits substantial performance degradation when operating with only two frames in their research. We argue that the input token set with a size equal to 2 frames can be optimized. Also, they are not validated in large-scale pre-training to achieve competitive results for real-world applications, and the performance gain with larger resolution input comes with quadratic computation cost increase, which is also insufficient.

We propose a new perspective of the desired optimal computation-and-accuracy and spatial-and-temporal trade-offs across various settings, termed **'Token Optimization'** (TO in short), as shown in Figure 1. It involves selecting an optimized set of input tokens based on budget from more properly sampled videos for information maximization. Since there is no perfect token selector for all settings, we promote a better solution of using flexible sampling, with more densely and finely sampled videos for higher computation and more sparsely sampled videos for lower budget. Spatial and temporal trade-offs are also achieved.

To better achieve TO, we propose a new data augmentation tool for training flexible video models, termed **Flux** for **Fl**exibility in **u**niversal conte**x**ts, by utilizing the combination of flexible sampling and token selector to make training flexible while not introducing costs. We integrate it with the Unmasked Teacher [42] pre-training, supervised-training, multi-modal contrastive training, and chat-centric training to validate its efficiency. Our new recipe brings more robustness caused by Flux augmentation both in standard settings and our TO setting. Two plug-in modules are used for better feature extraction regarding the reduced token set with flexible reduction rates and input spatiotemporal resolutions. We build bottom-up ablation studies in turning the model flexible. The resulting model FluxViT achieves supreme improvements regarding the currently state-of-the-art InternVideo2-Dist series models in fair settings across budgets and across tasks, as in Figure 2.

Our FluxViT-S outperforms the previous state-of-the-art small-scale model, InternVideo2-S [77], by **2.2%** on K400 under standard computation constraints. Moreover, it achieves comparable performance to InternVideo2-S while utilizing only about **10%** of inference cost. FluxViT-B achieves competitive results with state-of-the-art much larger scale models on various tasks, including scene-based action recognition task [34] (**90.0%** top-1 accuracy on K400), motion-intensive [23] (**75.8%** top-1 accuracy on SSv2), long-term task [64] (**94.1%** top-1 accuracy on COIN), zero-shot text-to-video retrieval [1, 12, 28, 61, 81] (**49.9%** R@1 on MSRVTT, **53.5%** R@1 on DiDeMo and **56.7%** R@1 on ActivityNet). Notably, it maintains state-

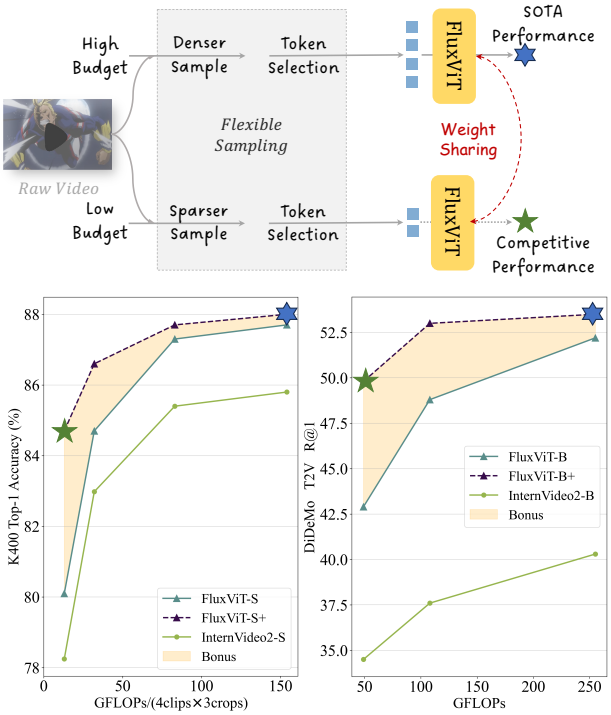

Figure 2. **Overview of our Flux method.** The same-scaled FluxViT and InternVideo2 [77] series models are both pre-trained with the InternVideo2-1b model as the teacher using the same dataset. The "FluxViT+" refers to the results using Token Optimization at test time with the same GFLOPS.

of-the-art performance among the similar scaled models while using only **1/4** to **1/2** of their tokens, resulting in **70%** to **95%** savings in computation. On chat centric tasks including MVbench [41] for general perception and Dream-1k [70] for fine-grained caption, our model surpasses widely used SigLIP, CLIP and UMT in 'linear prob' setting across computation budgets, where only the projector between the ViT and LLM is trained for comparability regarding the ViT's capability as the common stage-1 training in chat models. This highlights Flux's strong effect across computation constraints and tasks and underscores the robustness afforded by such flexibility.

2. Related Work

Flexible Model Training. Many recent studies focus on developing a method to train a single model that performs well across various settings, facilitating seamless adaptation to diverse downstream deployment scenarios. These methods include training a model as a set of shared weights to be pruned for specific tasks [7, 84], as well as optimizing models for varied spatial resolutions [65], frame counts [88], spatiotemporal resolutions [48, 51, 73, 80], patch sizes [6], and representation dimensions [8, 36]. The Resformer [65] and FFN [88] methods propose co-training with three different resolution settings in a single batch, uti-

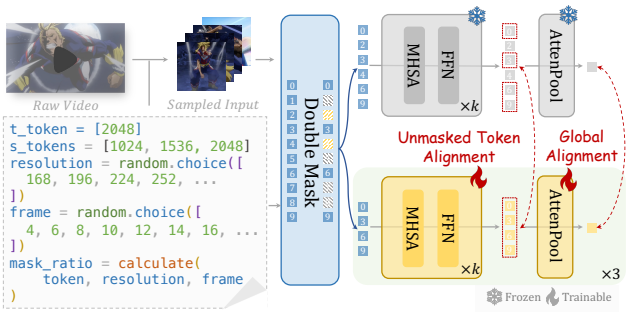

Figure 3. **Our proposed Flux method with UMT framework.** We show that our proposed Flux training is easy to integrate with mainstream video training frameworks.

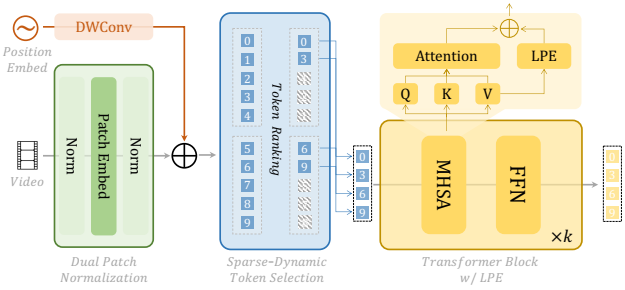

Figure 4. **Our proposed essential modules for Flux.** From the model side, Flux modules include Group-dynamic token selector, dual patch norm, and Global-Local positional embedding.

lizing self-distillation mechanisms to enhance performance under limited input resolutions for spatial and temporal dimensions, although separately. We argue that they can't achieve optimal performance within constrained computational budgets. Simply reducing frame counts and resolutions is suboptimal. Further, performance gains brought by high-resolution inputs lead to significant cost increases, a challenge also noted in Oryx [51]. Token Optimization addresses this by optimizing the set of input tokens anytime.

Masked Vision Modeling. Mask has always been seen as a critical method in training and utilizing scaled vision models, especially in the video domain. Previous methods can be mainly divided into two categories. First, drawing insights from the masked language modeling [16, 17, 52], researchers use masks in vision transformer [18] pre-training as a data augmentation strategy, as seen in masked autoencoder frameworks [21, 27, 66, 72] and BERT-like mask-then-predict frameworks [4, 58, 74, 90]. Works like UMT [42], MVP [79], and Milan [29] further utilize such augmentation strategy to predict the CLIP [60] model's feature instead of native pixels as a Teacher-Student pre-training method. Second, methods like MAR [59] and MCM [31] apply masking techniques during the fine-tuning stage and inference to save computation with minimal performance loss. We here propose that mask can be used as novel data-augmentation tool with our flexible sampling to boost performance with our Token Optimization idea.

3. Method

In this section, we introduce our **Flux** for effectively pre-training video models that represent videos at varying resolution scales using tokens with flexible quantities robustly.

3.1. Flux-Pretraining

We begin with a widely adopted pre-training paradigm in which the representations of a teacher model are employed as soft labels for a student model. Popular approaches following this typically rely on fixed sampling grids and utilize all extracted visual tokens for both the teacher and student models. However, this methodology suffers from substantial redundancy, particularly in video data at finer resolutions. To address this, Unmasked Teacher (UMT) [42] proposes using only a subset of the visual tokens for the student while retaining the full input for the teacher. This masked alignment strategy enhances both efficiency and robustness for the student model. We propose a further examination of the input tokens for the teacher and leverage flexible sampling grids combined with a group-dynamic token selection mechanism. This enables us to select the same number of tokens as the original setting but from a larger pool extracted at flexibly higher spatiotemporal resolutions that are more informative, thus harvesting more diversified and meaningful representations from the teacher without cost, as shown in Figure 3. The other settings, like alignment losses, are the same as the original UMT, which trains the competitive InternVideo2-Series models [77].

Flexi-Sampling. For each video in a batch, we randomly select a frame count ranging from F_{min} to F_{max} with a temporal stride t_s and select a spatial resolution between R_{min} and R_{max} with a spatial stride r_s and enforce a threshold T_{thres} in keeping a reasonable size of visual token pool.

Double Mask Module. To make the training flexible while not introducing new costs, we leverage token selection in the teacher input side. We propose a simple but effective method that prioritizes tokens exhibiting significant changes between frames. We use additional sparse groups for this selection to ensure coverage of the full video and resist several rapid changes in videos, termed **Group-Dynamic Selector**. Formally, with a pre-defined token count K, which is set to keep unchanged teacher forward cost, and a given tokenized frame sequence $\{F_1, F_2, ..., F_T\}$ by teacher's patch embedding layer, we uniformly sample it into N sparse groups where $B_i = \{F_t \mid t \in [iB, (i+1)B-1)\}, i \in [0, N)$. In each group, we compute the dynamic value as $\mathrm{D}(F_{t+1,i}) = ||F_{t+1,i} - F_{t,i}||_p$, where $F_{t,i}$ means the value of i_{th} spatial token in the t_{th} frame and p means the p-dim distance. Then we take the $\mathrm{top}(\frac{K}{N})$ tokens with the highest dynamic values in each group to

form K visible tokens in total. This selector, along with flexible sampling, is utilized for the teacher model. We also employ another student mask based on the teacher's attention score of CLS token adopted in Unmasked Teacher. The usage of the two masks is referred to as the Double Mask Module. Such a module greatly boosts the student model's performance and brings no additional cost, showing a natural integration of Flux in the UMT framework.

3.2. Flux-Tuning

For the stage of supervised tuning, we keep the flexi-sampling method and group-dynamic selector for the trained student model. Table 1 shows such a method's gain even on a competitive model pre-trained with standard setting, showing Flux as a general augmentation tool.

Token Optimization As proposed in the introduction, we hope to maximize token information under any given budget, termed as Token Optimization at downstream tasks. Thus, during evaluation, we use a subset of the train-set to find optimized sampling under a given token count for the task. We provide a heuristic searching method in the Appendix, which takes acceptable overhead.

3.3. FluxViT

Conventional ViT architecture struggles with the variable token numbers, flexible spatiotemporal resolution, and sparsity patterns inherent in Flux training. Thus, we propose two useful plug-in augmentation modules: Global-Local Positional Embedding (**GLPE**) and Dual Patch Normalization (**DPN**), as shown in Figure 4.

Global-Local Positional Embedding. As we utilize flexible sampling with various spatiotemporal resolutions and token reduction rates, it is important to encode the position information of the selected input tokens about where they are from and how they are related. Thus, Global-Local Positional Embedding (GLPE) is utilized. We enhance the global learnable positional embedding, initialized using the sine-cosine method for the maximum possible input size, by applying a Depth-Wise Convolution [30]. Then, to encode local positional information at a finer granularity in the position-invariant attention mechanism, we apply a Linear Projection function (**LPE**) to the value vector $\mathbf{V}$. The modified attention is formulated as:

$$\mathbf{Z} = (\mathrm{Softmax}(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{D}}) + LPE)\mathbf{V}. \quad (1)$$

Such LPE can be generated regardless of the input token number by adding a bias term for the permutation-invariant attention. This is vital in Flux to encode the discrete position information. The Value-dependent nature of LPE is more robust in Flux than others, like RoPE, in that the unmasked tokens are from different places from the source. The encoding of the relative relations between them is naturally Value-dependent. RoPE is more useful in scenarios without token selection for length extension.

Dual Patch Normalization. The Patch Embedding layer plays a crucial role in dynamics estimation and it must be robust to various spatiotemporal resolution distributions. We observed that an additional Layer Normalization after the standard Patch Embedding improves the accuracy of dynamic estimation, leading to more effective token selection. Furthermore, the flexible sampling strategy leads to a varied distribution of input tokens. Consequently, we find the gradient norm of the Patch Embedding layer can become excessively large, as shown in the Appendix. To stabilize training, we introduce a second Layer-Normalization layer before the Patch Embedding operation.

3.4. Train with Varied Input Lengths

As we aim to train a flexible video model that can seamlessly adjust to varied token numbers according to budgets, we propose to use a multi-number co-training approach. For Flux-UMT frameworks, we suggest utilizing three distinct input numbers within a single batch for the student model. This strategy maximizes the benefits of the extensive teacher-forward computation while enhancing robustness under varying conditions. Specifically, we forward the student three times using three different token numbers and compute the teacher-student alignment loss for each. In standard training frameworks without a pre-trained teacher, we incorporate additional self-distillation, using the final aggregated features with the largest input as the teacher for the intermediate and the intermediate as the teacher for the smallest. We include training efficiency in the ablation.

4. Experiments

We leverage InternVideo2-1B [77] as the teacher in Flux-UMT due to its strong performance in general video understanding tasks. We use the same data-decomposition and similar training recipe as the currently most competitive InternVideo2-series distilled models as baselines. Note that they are also built using the Unmasked Teacher framework under the supervision of InternVideo2-1B. For notation simplification, we use Flux-Single for Flux training with a single token count and Flux-Multi for Flux training with multi token counts, while Flux-PT is for Flux with UMT pretraining and Flux-FT for supervised tuning.

4.1. Ablation Study

In the ablation study, we ablate the effect of our Flux through a systematic approach, progressively transforming the original UMT framework into our Flux-UMT and getting our competitive FluxViT. We utilize K710 for pre-training, which integrates the K400 [34], K600 [9], and

Method	Input Size	Test Token Number 3072	2048	1024	512	Avg
8×224^2 Direct Tuned	2×224^2	∅	∅	∅	69.5	69.5
	4×224^2	∅	∅	80.5	73.6	77.1
	8×224^2	∅	83.9	81.9	72.7	79.5
	12×224^2	84.6	84.2	80.9	69.8	79.9
	16×224^2	84.5	83.5	79.1	67.0	78.5
	20×224^2	84.1	82.7	77.7	64.3	77.2
	24×224^2	83.6	82.1	76.4	62.3	76.1
	Avg	84.2	83.3	79.4	68.5	-
	Max	84.6	84.3	81.9	73.6	-
2048 fixed count Flux-Single Tuned	2×224^2	∅	∅	∅	68.1	68.1
	4×224^2	∅	∅	79.9	74.7	77.3
	8×224^2	∅	84.3	82.7	76.7	81.2
	12×224^2	85.1	85.1	82.8	75.8	82.2
	16×224^2	85.4	85.2	82.6	75.3	82.1
	20×224^2	85.6	85.1	82.2	74.6	81.9
	24×224^2	85.4	84.8	82.0	74.1	81.6
	Avg	85.4	84.9	82.0	74.2	-
	Max	85.6	85.2	82.8	76.7	-
		↑1.0	**↑0.9**	**↑0.9**	**↑3.1**	-
12×224^2 Direct Tuned	2×224^2	∅	∅	∅	69.3	69.3
	4×224^2	∅	∅	80.5	72.8	76.7
	8×224^2	∅	83.9	81.9	72.7	79.5
	12×224^2	85.0	84.3	80.9	67.8	79.4
	16×224^2	84.9	83.9	78.9	65.2	78.2
	20×224^2	84.7	83.4	77.6	62.4	77.0
	24×224^2	84.3	82.6	76.4	60.4	75.9
	Avg	84.7	83.6	79.4	67.2	-
	Max	85.0	84.3	81.9	72.8	-
(3072, 2048, 1024) Flux-Multi Tuned	2×224^2	∅	∅	∅	72.2	72.2
	4×224^2	∅	∅	81.0	79.3	80.2
	8×224^2	∅	84.4	82.8	80.3	82.5
	12×224^2	85.4	85.2	83.3	79.9	83.5
	16×224^2	85.7	85.1	83.5	79.2	83.4
	20×224^2	85.7	85.3	83.0	78.9	83.2
	24×224^2	85.6	85.0	82.7	78.2	82.9
	Avg	85.6	85.0	82.7	78.3	-
	Max	85.7	85.3	83.5	80.3	-
		↑0.7	**↑1.0**	**↑1.6**	**↑7.5**	-

Table 1. **Directly use Flux-Tuning with the previsou SOTA InternVideo2-S on K400.** Results are reported on K400 using 1clip×1crop. It shows that Flux can be used as an advanced augmentation tool directly in supervised tuning scenario.

K700 [10] datasets with duplicates removed. Performance is evaluated using scene-based K400 and motion-intensive SSv2 [23]. We use 1 clip × 1 crop for test configurations. For flexible sampling, we set the parameters as follows: F_{min}=4, F_{max}=24, t_s=2, R_{min}=168, R_{max}=252, r_s=28 and T_{thres}=(2048, 4096) by default.

Flux-Single Pre-training and Tuning. Figure 5 illustrates the performance of three models differing in their pre-training and tuning approaches. The first model, termed the *FixRes Distilled FixRes Tuned* model, employs a fixed resolution of 8×224^2 during both the pre-training stage and fine-tuning stage, the same as the original setting to get InternVideo2-S. We evaluate this tuned model using a fixed input token count of 2048 by our token-selector on

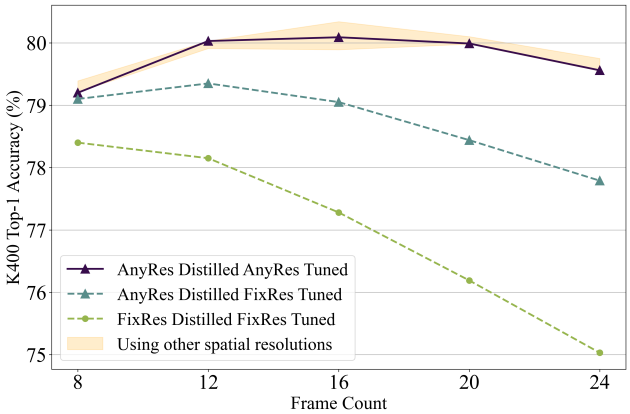

Figure 5. **Comparison between different training methods on K400 using a fixed number of 2048 tokens.** Note the three lines and all the points share similar training and inference costs. The shaded part shows results for the *AnyRes Distilled AnyRes Tuned* model with spatial resolution in range (196, 252), while others use a fixed spatial resolution at 224.

Mask Type	K400	*w/* TO	SSv2	*w/* TO
∅ (single res)	78.4	78.4	65.4	65.4
Random	78.6	79.0 (↑0.6)	65.3	65.9 (↑0.5)
Tube	78.8	80.0 (↑1.6)	65.7	66.7 (↑1.3)
Dynamic(L1)	78.7	79.8 (↑1.4)	65.7	66.6 (↑1.2)
Dynamic(L2)	78.8	80.0 (↑1.6)	65.8	66.7 (↑1.3)
Group-Dynamic(L2)-2	78.8	80.2 (↑1.8)	66.0	67.0 (↑1.6)
Group-Dynamic(L2)-4	79.2	80.3 (↑1.9)	66.3	67.3 (↑1.9)

Table 2. **Ablation on using token selection strategies** We validate the effect of different token selection methods on Flux-Single-Pre-training and Tuning on ViT. A group size of 4 works well.

the K400 dataset, which corresponds to the number of input tokens in the standard 8×224^2 configuration, but with varied frame counts to assess the model's capability to leverage additional temporal information in fixed-number tokens for enhanced performance. Thus, **models trained and tuned with fixed sampling grid can hardly harvest costless performance gain with Token Optimization.**

In contrast, the *AnyRes Distilled FixRes Tuned* model utilizes Flux-UMT-Single upon the original setting. Through Flux token selection, we maintain consistent pre-training computational costs by regulating a fixed quota of 2048 input tokens. The tuning method also remains unchanged. The diversified features from the teacher by the flexible sampling accumulate more valued representations for the student, gaining a +0.7% improvement at the original 8×224^2 test setting. Notably, the new model exhibits greater robustness in processing a fixed number of tokens from larger spatiotemporal resolution, achieving another +0.3% gain with Token Optimization costlessly.

We further test tuning the any-res trained model using Flux-FT with a fixed input token number of 2048 on K400, as termed *AnyRes Distilled AnyRes Tuned* in the figure. As

Method & Arch	K400	*w/* TO	SSv2	*w/* TO
Vanilla + ViT	78.4	78.4	65.4	65.4
Vanilla + FluxViT	79.3	79.6 (↑1.2)	66.0	66.4 (↑1.0)
Flux-Single + ViT	79.2	80.3 (↑1.9)	66.3	67.3 (↑1.9)
With new positional embeddings				
w/ RoPE	79.5	80.7 (↑0.4)	66.5	67.5 (↑0.2)
w/ GPE	79.4	80.5 (↑0.2)	66.4	67.4 (↑0.1)
w/ LPE	79.7	81.0 (↑0.7)	66.8	68.3 (↑1.0)
w/ GLPE	79.9	81.3 (↑1.0)	67.0	68.6 (↑1.3)
With DPN				
w/ DPN	79.8	81.2 (↑0.9)	66.9	68.4 (↑1.1)
Combining the two modules				
Flux-Single + FluxViT	80.5	81.7 (↑3.3)	67.6	69.3 (↑3.9)
Flux-Multi + FluxViT	81.4	82.4 (↑4.0)	68.4	70.0 (↑4.6)

Table 3. Results with 8×224^2(and TO w/ 2048 tokens, which is the same token count as in 8×224^2) on K400 and SSv2. (↑) for absolute improvement and (↑) for relative gain. Vanilla means PT and FT without Flux augmentation.

Settings	8×224^2		2048 + TO	
	K400	SSv2	K400	SSv2
Baseline	**80.5**	67.6	**81.7**	**69.3**
Change T_{thres} to (2048, 6144)	80.1	67.2	81.2	68.7
w/o varied spatial resolution	**80.5**	**67.7**	81.4	69.1
enlarge F_{max} to 32	80.4	67.4	81.6	69.1
enlarge Res_{max} to 336	80.2	67.3	81.4	69.0

Table 4. **Ablations on hyper-parameter on sampling designs for Flux-Single training with FluxViT.** Most settings regarding flexible sampling cause only minor influences except T_{thres}.

Pre-training Length	Fine-tuning Length	Test Length		
		2048	1024	512
∅(*w/o* Flux)	∅	79.3	74.7	62.1
Single	Single	80.5	77.4	65.8
Single	Multi(*w/* align)	80.3	79.0	75.0
Multi	Single	81.0	78.8	70.2
Multi	Multi	80.9	79.1	73.3
Multi	Multi(*w/* align)	**81.4**	**80.3**	**76.6**

Table 5. **Ablation on using varied input lengths in training FluxViT on K400.** Results are tested with fixed 8×224^2 input but varied test token lengths based on our selector.

shown, this tuning method can harvest a +1.9% top-1 accuracy gain on K400 compared with the normal distilled and normal tuned method while nearly bringing no computation increment throughout pre-training and fine-tuning.

Table 1 further shows that only combining Flux-Tuning and Token Optimization on a competitive model pre-trained with standard setting can achieve non-trivial gain, highlighting that the sampling and selection module in Flux can effectively serve as novel augmentation tools.

Token Selection Module. We test four token selection methods for Flux-Single, including random, tube, dynamic

Model	Extra Data	#P(M)	GFLOPs	Top-1
TimeSformer-L [5]	-	121	2380×3	80.7
VideoSwin-L [50]	IN-21K	197	604×12	83.1
VideoMAE-L [66]	-	305	3958×21	86.1
CoVeR-L [87]	JFT-3B+SMI	431	5860×3	87.1
UniFormerV2-L [39]	CLIP-400M+K710	354	12550×6	90.0
UMT-L [42]	K710	431	5860×3	90.6
VideoMAE2-H [72]	UnlabeledHybrid	633	1192×15	88.6
ViViT-H [2]	JFT-300M	654	3981×12	84.9
MTV-H [82]	IN-21K+WTS-60M	1000+	6130×12	89.9
CoCa-G [85]	JFT-3B+ALIGN-1.8B	1000+	N/A×12	88.9
MViTv1-B [19]	-	37	70×5	80.2
MViTv2-B [44]	-	37	255×5	81.2
ST-MAE-B [21]	K600	87	180×21	81.3
VideoMAE-B [66]	-	87	180×15	81.5
VideoSwin-B [50]	IN-21k	88	282×12	82.7
UniFormer-B [40]	IN-1k	50	259×12	83.0
UMT-B [42]	K710	87	180×12	87.4
InternVideo2-B [77]	K710+MASH	96	440×12	88.4
FluxViT-B$_{e200}$	-	97	440×12	88.7 **89.4**
			49×12	84.0 **86.7**
FluxViT-B$_{e100}$	K710+MASH	97	440×12	89.6 **90.0**
			255×12	89.3 **89.7**
			108×12	87.3 **88.9**
			49×12	84.7 **87.4**
UniFormer-S [40]	IN-1k	21	42×4	80.8
MViTv2-S [44]	-	35	64×5	81.0
VideoMAE-S [66]	-	22	57×15	79.0
VideoMAE2-S [72]	-	22	57×15	83.7
InternVideo2-S [77]	K710+MASH	23	154×12	85.8
FluxViT-S$_{e200}$	-	24	154×12	86.4 **87.3**
			13×12	79.7 **84.0**
FluxViT-S$_{e100}$	K710+MASH	24	154×12	87.7 **88.0**
			83×12	87.3 **87.7**
			32×12	84.7 **86.6**
			13×12	80.1 **84.7**

Table 6. **Comparison with the state-of-the-art methods with on scene-related Kinetics-400.** #P is short for the number of parameters. The blue values of FluxViT show results using larger spatiotemporal resolutions but keeping fixed input token count to 3072, 2048, 1024, and 512 respectively, corresponding to the four GFLOPs listed. SMI is short for the train set of SSv2, MiT and ImageNet and MASH for MiT, ANet, SSv2 and HACS.

(as also directly used in [31]) and our group-dynamic method. We further ablate the token dynamics measurement and the sparse group size. Table 2 shows that our group-dynamic strategy with token dynamics measured by L2-distance of tokens within adjacent frames in sparsely divided four groups is the most robust among the token-selection methods. More advanced token selectors can be utilized for better results but with increased complexity, cost, and tedious hyperparameters.

FluxViT Modules. Table 3 analyzes the impact of our proposed plug-in modules. The inclusion of Global-Local positional embedding can strengthen the model's robustness when processing sparse tokens derived from larger spatiotemporal resolutions. We compare our GLPE with ROPE [63] within our Flux scheme and show better results

Model	Extra Data	GFLOPs	Top-1	Top-5
TimeSformer-L [5]	IN-21k	2380×3	62.3	-
MViTv1-B [19]	K400	455×3	67.7	70.9
MViTv2-B [44]	K400	255×3	70.5	92.7
VideoMAE-B [66]	K400	180×6	69.7	92.3
VideoMAE-L [66]	K400	596×6	74.0	94.6
UniFormerV2-B [39]	CLIP-400M	375×3	70.7	93.2
UniFormerV2-L [39]	CLIP-400M	1718×3	73.0	94.5
UMT-B [42]	K710	180×6	70.8	92.4
InternVideo2-B [77]	K710+MASH	253×6	73.5	94.4
FluxViT-B	K710+MASH	440×6	75.3 **75.6**	95.1 **95.1**
		255×6	75.1 **75.5**	94.9 **95.1**
		108×6	72.0 **75.1**	93.3 **94.8**
		49×6	56.8 **73.9**	84.8 **94.4**
UniFormer-S [40]	IN-1K	42×3	67.7	91.4
VideoMAE-S [66]	K600	57×6	66.8	90.3
InternVideo2-S [77]	K710+MASH	83×6	71.5	93.4
FluxViT-S	K710+MASH	154×6	73.4 **73.8**	94.1 **94.1**
		83×6	72.9 **73.4**	94.0 **94.1**
		32×6	70.0 **72.5**	93.4 **93.8**
		13×6	55.3 **70.9**	83.7 **93.1**

Table 7. **Comparison with the state-of-the-art methods with on motion-intensive SSv2.** Our model achieves far better results.

with LPE illustrated in the Method section. Furthermore, adding a dual patch norm module optimizes the performance gains in TO. Combining the two modules achieves +3.3% and +3.9% on K400 and SSv2 respectively with TO.

Hyper-parameters of flexi-sampling Table 4 gives ablation studies on the results of different hyper-parameters regarding sampling. Most settings regarding flexible sampling cause only minor influences except T_{thres}. A proper scale of the flexible space is the most powerful.

Flux-Multi training. We compare our proposed method of co-training with three input numbers(2048, 1024, 512) in both pretraining and tuning in Table 5 and also validate the tuning effects on well-pretrained InternVideo2-S in table 1(3072, 2048, 1024). As shown, pre-training with multiple token numbers can not only boost performance under standard settings but also increase model consistency in input token sparsity. Tuning with different token numbers with a smoothed-L1-loss based self-distillation mechanism can further enhance such alignment consistently.

4.2. Single Modality Results

We scale up the training data using the K-MASH[77] dataset of 1.1 million samples, aligning with the data employed in the Internvideo2-Distilled series models, including K710, SSv2[23], ANet[28], HACS[89], and MiT[56]. We train the FluxViT model using a total batch size of 2048 for 100 epochs. FluxViT's performance is validated using scene-based K400, motion-intensive SSv2, and long-term COIN with Token Optimization.

K400. Table 6 reports our results compared with the previous solutions on K400. We set 3072, 2048, and 1024 as our token numbers in Flux-Multi-Tuning. For our FluxViT-S, we report **88.0%**(+2.2%) on K400 compared with InternVideo2-S, which is the previous SOTA solution, while still achieving **84.7%** with nearly the same computation as the lightweight network UniFormer [40] but with competitive accuracy. For fair comparison without K-MASH, we pretrain FluxViT using only K400 in Flux-UMT, which also shows much better results.

Model	*e2e*	BackBone	Top-1
Distant Supervision [47]	✗	TimeSformer	90.0
ViS4mer [32]	✗	Swin-B	88.4
Turbo$_{f32}$ [25]	✓	VideoMAE-B	87.5
VideoMamba$_{f64}$ [43]	✓	VideoMamba-S	88.7
VideoMamba$_{f64}$ [43]	✓	VideoMamba-M	90.4
InternVideo2$_{f12}$ [77]	✓	InternVideo2-S	90.0
MA-LMM [26]	✓	MLLM	93.2
HERMES [33]	✓	MLLM	93.5
FluxViT$_{3072}$	✓	FluxViT-S	91.8 **92.1**
FluxViT$_{2048}$	✓	FluxViT-S	91.5 **91.9**
FluxViT$_{1024}$	✓	FluxViT-S	89.8 **91.0**
FluxViT$_{3072}$	✓	FluxViT-B	93.9 **94.1**
FluxViT$_{2048}$	✓	FluxViT-B	93.7 **93.9**
FluxViT$_{1024}$	✓	FluxViT-B	92.5 **93.2**

Table 8. **Comparison with the state-of-the-art on long-form video classification COIN dataset.** We report the results based on our preset token number, with the left line using unmasked 12, 8, 4 frames, and 224 spatial resolution while the blue values show results that can be achieved using more informative tokens.

SSv2. Table 7 shows our model's performance in dealing with motion-intensive video understanding tasks. Our FluxViT-S and FluxViT-B models set new state-of-the-art performance on SSv2 with either standard or limited computation cost. For FluxViT-B, we achieve 75.6% with standard computation while still achieving 75.1% with only 25% cost of the standard setting. Previous work [88] instead gets a huge performance drop using 2 frames, which corresponds to the least computation listed.

COIN. Table 8 shows our model's performance in dealing with long-form video understanding tasks. Our FluxViT-S and FluxViT-B models set new state-of-the-art performance on COIN with either standard or limited computation cost.

4.3. Multi Modality Retrieval Results

We use the pre-trained FluxViT with the CLIP [60] framework and our Flux method to train a video clip model using 27M corpus, far lower than most baseline models, including 25M coarse level caption data: Webvid10M [3], CC3M [62], COCO [46], VG [35], SBU [57], CC12M [11] and 2M high-quality data: S-MiT [55], InternVid-2M-Recap [76]. We leverage MobileClip-B [67] as the text encoder and only use the vanilla VTC loss, which is short for Video-Text Contrastive loss in training. We only unfreeze the ViT projector for the first stage using the 25M coarse

Model	MSR	DDM	ANet	LSMDC	MSVD
Internvideo2-S$_{2048}$ [77]	35.6	33.7	34.5	14.7	41.8
Frozen-B [3]	18.7	20.2	-	-	-
VIOLET-B [22]	25.9	23.5	-	-	-
Singularity-B [37]	34.0	37.1	30.6	-	-
OmniVL-B [69]	34.6	33.3	-	-	-
CLIP4Clip-B [53]	30.6	-	-	13.6	36.2
UMT-B [42]	35.2	41.2	35.5	19.1	42.3
Internvideo2-B$_{2048}$ [77]	40.3	40.3	41.5	18.7	49.1
VINDLU-L [14]	32.0	36.9	30.9	-	-
InternVideo-L [75]	40.7	31.5	30.7	17.6	43.4
UMT-L [42]	40.7	48.6	41.9	24.9	49.0
ViClip-L [76]	42.4	18.4	15.1	20.1	49.1
InternVideo2-L [77]	42.1	42.8	43.6	21.4	-
LanguageBind-L [91]	42.8	39.7	38.4	-	54.1
LanguageBind-H [91]	44.8	39.9	41.0	-	53.9
VideoCoCa-G [83]	34.3	-	34.5	-	-
VideoPrism-G [83]	39.7	-	52.7	-	-
VAST-G [13]	49.3	55.5	-	-	-
FluxViT-S$_{2048}$	44.4	48.3	52.4	20.8	49.4
	45.0	**49.3**	**52.4**	**22.4**	**49.7**
FluxViT-S$_{1024}$	42.2	45.4	47.2	18.7	48.1
	44.5	**49.0**	**50.3**	**20.5**	**48.5**
FluxViT-S$_{512}$	36.8	38.5	38.2	17.7	45.5
	40.5	**45.8**	**44.7**	**19.0**	**46.9**
FluxViT-B$_{2048}$	49.8	52.2	56.6	23.7	53.8
	49.9	**53.5**	**56.7**	**25.4**	**54.2**
FluxViT-B$_{1024}$	48.0	48.8	51.8	22.6	52.8
	49.1	**53.0**	**54.8**	**24.1**	**53.4**
FluxViT-B$_{512}$	42.6	42.9	42.8	20.1	50.7
	47.2	**49.8**	**50.3**	**22.8**	**52.1**

Table 9. **Zero-shot text-to-video retrieval on MSRVTT ("MSR"), DiDeMo ("DDM"), AcitivityNet ("ANet"), LSMDC, and MSVD.** We only report the R@1 accuracy. The upper line regarding FluxViT shows results with non-masked 8×224^2, 4×224^2 and 2×224^2 input setting as indicated by the token count while each lower **bold** line shows results further using more informative tokens. We employ Dual Softmax Loss for the results.

caption data for 3 epochs. Then we unfreeze all the modules for the second stage using the 2M high-quality data for one epoch to fully boost the ViT's capacity. By default, we train the clip model with a batch size of 4096. Also, we train the clip model with three input token numbers, 2048, 1024, and 512, and utilize smoothed-L1-loss on the final aggregated vision features to perform self-distillation and compute contrastive loss for each vision feature of the three numbers. Table 9 indicates that the clip model trained with our FluxViT method outperforms the Internvideo2-Series model with a large margin and further surpasses most of the top-performing models in Large or even Giant scale. Results on zero-shot action recognition results are included in Appendix for full validation of FluxViT-CLIP.

Encoder	#Tokens	*w/* TO	MVbench	Dream1k-F1
Clip-L [60]	8×256	✗	45.6	28.4
SigLIP$_{336}$-L [86]	8×576	✗	46.7	29.2
InternVideo2-L [77]	8×224	✗	47.0	28.7
SigLIP$_{336}$-L [86]	4×576	✗	44.5	25.4
UMT-L [42]	4×256	✗	45.0	24.6
FluxViT-L	8×256	✗	48.3	29.0
	4×256		46.9	27.9
	2×256		46.0	25.6
FluxViT-L	2048	✓	49.0 (↑0.7)	29.5 (↑0.5)
	1024		47.7 (↑0.8)	28.5 (↑0.6)
	512		47.6 (↑1.6)	27.5 (↑1.9)

Table 10. **Results on Chat-Centric benchmarks MVbench (General perception) and Dream1k(Fine-grained caption).** Models are trained in a multimodal 'linear prob' setting where both the Encoder and the LLM are frozen. #Tokens for the number of visual tokens by the vision encoder.

4.4. Chat-Centric Evaluation Results

We scale the pre-trained model size to FluxViT-Large and evaluate its performance under the VideoChat framework. We employ a single-layer MLP projector between ViT and an LLM(Qwen2-7B[73]) and train only the projector to ensure an unbiased evaluation of the ViT's performance, termed 'linear probe' setting in multimodal scenarios. Such a probing strategy is widely used in the Stage-1 training of multi-modal chat models. We utilize a large-scaled trainset including LLava-558K[49], S-MiT, 700k filtered subset of WebVid-10M, VidLN[68], and SSv2-open-ended. We then compare the models' performance on two benchmarks: MVbench, which assesses general spatiotemporal perception capabilities, and Dream1k, which evaluates the model's fine-grained ability to generate detailed captions. The other models tested include CLIP-L, SigLIP-L, and UMT-L, which are widely used in Video MLLMs. Given that ViT and LLM remain frozen, we maintain the original input settings as their pretraining phases to ensure comparability. We will scale the corpus size and fully tune the chat model for full performance in future work.

5. Conclusion and Future work

We present a token optimization process to maximize information in limited tokens within any budget. It integrates seamlessly with mainstream frameworks, offering a simple and scalable method to enhance real-world applications and facilitate future video foundation models. Future work could explore more advanced token selection methods for improvement, as the group setting can ensure full video coverage but not fully resist consistent camera motions. We test advanced Vid-TLDR [15] in Appendix, but with increased cost, tedious hyper-parameter tuning, and unstable improvement. For all current tasks tested, including fine-grained captioning, our method is still simple and effective.

Acknowledgement

This work is supported by the National Key R&D Program of China (No. 2022ZD0160900), Jiangsu Frontier Technology Research and Development Program (No. BF2024076), and the Collaborative Innovation Center of Novel Software Technology and Industrialization.

References

[1] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *ICCV*, 2017. 2

[2] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. In *ICCV*, 2021. 6

[3] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *ICCV*, 2021. 7, 8

[4] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: Bert pre-training of image transformers. In *ICLR*, 2021. 3

[5] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *ICML*, 2021. 6, 7

[6] Lucas Beyer, Pavel Izmailov, Alexander Kolesnikov, Mathilde Caron, Simon Kornblith, Xiaohua Zhai, Matthias Minderer, Michael Tschannen, Ibrahim M. Alabdulmohsin, and Filip Pavetic. Flexivit: One model for all patch sizes. In *CVPR*, 2022. 2

[7] Han Cai, Chuang Gan, Tianzhe Wang, Zhekai Zhang, and Song Han. Once-for-all: Train one network and specialize it for efficient deployment. In *ICLR*, 2019. 2

[8] Mu Cai, Jianwei Yang, Jianfeng Gao, and Yong Jae Lee. Matryoshka multimodal models. *ArXiv*, abs/2405.17430, 2024. 2

[9] João Carreira, Eric Noland, Andras Banki-Horvath, Chloe Hillier, and Andrew Zisserman. A short note about kinetics-600. *ArXiv*, abs/1808.01340, 2018. 4

[10] João Carreira, Eric Noland, Chloe Hillier, and Andrew Zisserman. A short note on the kinetics-700 human action dataset. *ArXiv*, abs/1907.06987, 2019. 5

[11] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *CVPR*, 2021. 7

[12] David L. Chen and William B. Dolan. Collecting highly parallel data for paraphrase evaluation. In *ACL*, 2011. 2

[13] Sihan Chen, Handong Li, Qunbo Wang, Zijia Zhao, Mingzhen Sun, Xinxin Zhu, and Jing Liu. Vast: A vision-audio-subtitle-text omni-modality foundation model and dataset. In *NIPS*, 2023. 8

[14] Feng Cheng, Xizi Wang, Jie Lei, David J. Crandall, Mohit Bansal, and Gedas Bertasius. Vindlu: A recipe for effective video-and-language pretraining. *ArXiv*, abs/2212.05051, 2022. 8

[15] Joonmyung Choi, Sanghyeok Lee, Jaewon Chu, Minhyuk Choi, and Hyunwoo J. Kim. vid-tldr: Training free token merging for light-weight video transformer. In *CVPR*, 2024. 1, 8

[16] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *ArXiv*, abs/1810.04805, 2018. 3

[17] Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. Unified language model pre-training for natural language understanding and generation. *NeurIPS*, 32, 2019. 3

[18] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 3

[19] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. In *ICCV*, 2021. 6, 7

[20] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, 2019. 1

[21] Christoph Feichtenhofer, Haoqi Fan, Yanghao Li, and Kaiming He. Masked autoencoders as spatiotemporal learners. *NeurIPS*, 2022. 3, 6

[22] Tsu-Jui Fu, Linjie Li, Zhe Gan, Kevin Lin, William Yang Wang, Lijuan Wang, and Zicheng Liu. Violet: End-to-end video-language transformers with masked visual-token modeling. *ArXiv*, abs/2111.12681, 2021. 8

[23] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense. In *ICCV*, 2017. 2, 5, 7

[24] Jing Gu, Eliana Stefani, Qi Wu, Jesse Thomason, and Xin Wang. Vision-and-language navigation: A survey of tasks, methods, and future directions. In *ACL*, 2022. 1

[25] Tengda Han, Weidi Xie, and Andrew Zisserman. Turbo training with token dropout. In *BMVC*, 2022. 7

[26] Bo He, Hengduo Li, Young Kyun Jang, Menglin Jia, Xuefei Cao, Ashish Shah, Abhinav Shrivastava, and Ser-Nam Lim. Ma-lmm: Memory-augmented large multimodal model for long-term video understanding. In *CVPR*, 2024. 7

[27] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Doll'ar, and Ross B. Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022. 3

[28] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, 2015. 2, 7

[29] Zejiang Hou, Fei Sun, Yen-Kuang Chen, Yuan Xie, and S. Y. Kung. Milan: Masked image pretraining on language assisted representation. *ArXiv*, abs/2208.06049, 2022. 3

[30] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, M. Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional

neural networks for mobile vision applications. *ArXiv*, abs/1704.04861, 2017. 4

[31] Sun-Kyoo Hwang, Jaehong Yoon, Youngwan Lee, and Sung Ju Hwang. Everest: Efficient masked video autoencoder by removing redundant spatiotemporal tokens. In *ICML*, 2022. 1, 3, 6

[32] Md. Mohaiminul Islam and Gedas Bertasius. Long movie clip classification with state-space video models. In *ECCV*, 2022. 7

[33] Faure Gueter Josmy, Yeh Jia-Fong, Chen Min-Hung, Su Hung-Ting, Hsu Winston H, and Lai Shang-Hong. Bridging episodes and semantics: A novel framework for long-form video understanding. In *ECCV workshops*, 2024. 7

[34] Will Kay, João Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Apostol Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset. *ArXiv*, abs/1705.06950, 2017. 2, 4

[35] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 2017. 7

[36] Aditya Kusupati, Gantavya Bhatt, Aniket Rege, Matthew Wallingford, Aditya Sinha, Vivek Ramanujan, William Howard-Snyder, Kaifeng Chen, Sham M. Kakade, Prateek Jain, and Ali Farhadi. Matryoshka representation learning. In *NIPS*, 2022. 2

[37] Jie Lei, Tamara L Berg, and Mohit Bansal. Revealing single frame bias for video-and-language learning. *ArXiv*, abs/2206.03428, 2022. 8

[38] Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, 2022. 1

[39] Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Limin Wang, and Y. Qiao. Uniformerv2: Spatiotemporal learning by arming image vits with video uniformer. *ArXiv*, abs/2211.09552, 2022. 6, 7

[40] Kunchang Li, Yali Wang, Gao Peng, Guanglu Song, Yu Liu, Hongsheng Li, and Yu Qiao. Uniformer: Unified transformer for efficient spatial-temporal representation learning. In *ICLR*, 2022. 6, 7

[41] Kunchang Li, Yinan He, Yi Wang, Yizhuo Li, Wen Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. Videochat: Chat-centric video understanding. *ArXiv*, abs/2305.06355, 2023. 1, 2

[42] Kunchang Li, Yali Wang, Yizhuo Li, Yi Wang, Yinan He, Limin Wang, and Yu Qiao. Unmasked teacher: Towards training-efficient video foundation models. In *ICCV*, 2023. 2, 3, 6, 7, 8

[43] Kunchang Li, Xinhao Li, Yi Wang, Yinan He, Yali Wang, Limin Wang, and Yu Qiao. Videomamba: State space model for efficient video understanding. *ArXiv*, abs/2403.06977, 2024. 7

[44] Yanghao Li, Chaoxia Wu, Haoqi Fan, Karttikeya Mangalam, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. Improved multiscale vision transformers for classification and detection. *ArXiv*, abs/2112.01526, 2021. 6, 7

[45] Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. In *ICCV*, 2019. 1

[46] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 7

[47] Xudong Lin, Fabio Petroni, Gedas Bertasius, Marcus Rohrbach, Shih-Fu Chang, and Lorenzo Torresani. Learning to recognize procedural activities with distant supervision. *CVPR*, 2022. 7

[48] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *ArXiv*, abs/2310.03744, 2023. 2

[49] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023. 8

[50] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. In *CVPR*, 2022. 6

[51] Zuyan Liu, Yuhao Dong, Ziwei Liu, Winston Hu, Jiwen Lu, and Yongming Rao. Oryx mllm: On-demand spatial-temporal understanding at arbitrary resolution. *ArXiv*, abs/2409.12961, 2024. 2, 3

[52] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *NeurIPS*, 2019. 3

[53] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. Clip4clip: An empirical study of clip for end to end video clip retrieval and captioning. *Neurocomputing*, 2022. 8

[54] Muhammad Maaz, Hanoona Abdul Rasheed, Salman Khan, and Fahad Shahbaz Khan. Video-chatgpt: Towards detailed video understanding via large vision and language models. *ArXiv*, abs/2306.05424, 2023. 1

[55] Mathew Monfort and SouYoung Jin. Spoken moments: Learning joint audio-visual representations from video descriptions. In *CVPR*, 2021. 7

[56] Mathew Monfort, Bolei Zhou, Sarah Adel Bargal, Alex Andonian, Tom Yan, Kandan Ramakrishnan, Lisa M. Brown, Quanfu Fan, Dan Gutfreund, Carl Vondrick, and Aude Oliva. Moments in time dataset: One million videos for event understanding. *TPAMI*, 2020. 7

[57] Vicente Ordonez, Girish Kulkarni, and Tamara Berg. Im2text: Describing images using 1 million captioned photographs. In *NeurIPS*, 2011. 7

[58] Zhiliang Peng, Li Dong, Hangbo Bao, Qixiang Ye, and Furu Wei. Beit v2: Masked image modeling with vector-quantized visual tokenizers. *ArXiv*, abs/2208.06366, 2022. 3

[59] Zhiwu Qing, Shiwei Zhang, Ziyuan Huang, Xiang Wang, Yuehuang Wang, Yiliang Lv, Changxin Gao, and Nong Sang. Mar: Masked autoencoders for efficient action recognition. In *ACMMM*, 2022. 1, 3

[60] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual

models from natural language supervision. In *ICML*, 2021. 3, 7, 8

[61] Anna Rohrbach, Atousa Torabi, Marcus Rohrbach, Niket Tandon, Christopher Joseph Pal, H. Larochelle, Aaron C. Courville, and Bernt Schiele. Movie description. *International Journal of Computer Vision*, 2016. 2

[62] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL*, 2018. 7

[63] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 2024. 6

[64] 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 *CVPR*, 2019. 2

[65] Rui Tian, Zuxuan Wu, Qi Dai, Han Hu, Yu Qiao, and Yu-Gang Jiang. Resformer: Scaling vits with multi-resolution training. In *CVPR*, 2023. 1, 2

[66] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. VideoMAE: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In *NeurIPS*, 2022. 3, 6, 7

[67] Pavan Kumar Anasosalu Vasu, Hadi Pouransari, Fartash Faghri, Raviteja Vemulapalli, and Oncel Tuzel. Mobileclip: Fast image-text models through multi-modal reinforced training. In *CVPR*, 2024. 7

[68] Paul Voigtlaender, Soravit Changpinyo, Jordi Pont-Tuset, Radu Soricut, and Vittorio Ferrari. Connecting vision and language with video localized narratives. *CVPR*, 2023. 8

[69] Junke Wang, Dongdong Chen, Zuxuan Wu, Chong Luo, Luowei Zhou, Yucheng Zhao, Yujia Xie, Ce Liu, Yu-Gang Jiang, and Lu Yuan. Omnivl: One foundation model for image-language and video-language tasks. *ArXiv*, abs/2209.07526, 2022. 8

[70] Jiawei Wang, Liping Yuan, Yuchen Zhang, and Haomiao Sun. Tarsier: Recipes for training and evaluating large video description models. *arXiv*, abs/2407.00634, 2024. 2

[71] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 2016. 1

[72] 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 *CVPR*, 2023. 3, 6

[73] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *ArXiv*, abs/2409.12191, 2024. 2, 8

[74] 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. *CVPR*, 2022. 3

[75] Yi Wang, Kunchang Li, Yizhuo Li, Yinan He, Bingkun Huang, Zhiyu Zhao, Hongjie Zhang, Jilan Xu, Yi Liu, Zun Wang, Sen Xing, Guo Chen, Junting Pan, Jiashuo Yu, Yali Wang, Limin Wang, and Yu Qiao. Internvideo: General video foundation models via generative and discriminative learning. *ArXiv*, abs/2212.03191, 2022. 8

[76] Yi Wang, Yinan He, Yizhuo Li, Kunchang Li, Jiashuo Yu, Xin Jian Ma, Xinyuan Chen, Yaohui Wang, Ping Luo, Ziwei Liu, Yali Wang, Limin Wang, and Y. Qiao. Internvid: A large-scale video-text dataset for multimodal understanding and generation. *ArXiv*, 2023. 7, 8

[77] Yi Wang, Kunchang Li, Xinhao Li, Jiashuo Yu, Yinan He, Chenting Wang, Guo Chen, Baoqi Pei, Ziang Yan, Rongkun Zheng, Jilan Xu, Zun Wang, Yansong Shi, Tianxiang Jiang, Songze Li, Hongjie Zhang, Yifei Huang, Yu Qiao, Yali Wang, and Limin Wang. Internvideo2: Scaling foundation models for multimodal video understanding. In *ECCV*, 2024. 1, 2, 3, 4, 6, 7, 8

[78] Zun Wang, Jialu Li, Yicong Hong, Yi Wang, Qi Wu, Mohit Bansal, Stephen Gould, Hao Tan, and Yu Qiao. Scaling data generation in vision-and-language navigation. In *ICCV*, 2023. 1

[79] Longhui Wei, Lingxi Xie, Wen gang Zhou, Houqiang Li, and Qi Tian. Mvp: Multimodality-guided visual pre-training. In *ECCV*, 2022. 3

[80] Chao-Yuan Wu, Ross Girshick, Kaiming He, Christoph Feichtenhofer, and Philipp Krahenbuhl. A multigrid method for efficiently training video models. In *CVPR*, 2020. 2

[81] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*, 2016. 2

[82] Shen Yan, Xuehan Xiong, Anurag Arnab, Zhichao Lu, Mi Zhang, Chen Sun, and Cordelia Schmid. Multiview transformers for video recognition. In *CVPR*, 2022. 6

[83] Shen Yan, Tao Zhu, Zirui Wang, Yuan Cao, Mi Zhang, Soham Ghosh, Yonghui Wu, and Jiahui Yu. Video-text modeling with zero-shot transfer from contrastive captioners. *ArXiv*, abs/2212.04979, 2022. 8

[84] Jiahui Yu, Pengchong Jin, Hanxiao Liu, Gabriel Bender, Pieter-Jan Kindermans, Mingxing Tan, Thomas Huang, Xiaodan Song, Ruoming Pang, and Quoc Le. Bignas: Scaling up neural architecture search with big single-stage models. In *ECCV*, 2020. 2

[85] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *Transactions on Machine Learning Research*, 2022. 6

[86] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. *CVPR*, 2023. 8

[87] Bowen Zhang, Jiahui Yu, Christopher Fifty, Wei Han, Andrew M. Dai, Ruoming Pang, and Fei Sha. Co-training transformer with videos and images improves action recognition. *ArXiv*, abs/2112.07175, 2021. 6

[88] Yitian Zhang, Yue Bai, Chang Liu, Huan Wang, Sheng Li, and Yun Fu. Frame flexible network. In *CVPR*, 2023. 1, 2, 7

[89] Hang Zhao, Antonio Torralba, Lorenzo Torresani, and Zhicheng Yan. Hacs: Human action clips and segments

dataset for recognition and temporal localization. In *ICCV*, 2019. 7

[90] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Loddon Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. *ArXiv*, abs/2111.07832, 2021. 3

[91] Bin Zhu, Bin Lin, Munan Ning, Yang Yan, Jiaxi Cui, Wang HongFa, Yatian Pang, Wenhao Jiang, Junwu Zhang, Zongwei Li, Cai Wan Zhang, Zhifeng Li, Wei Liu, and Li Yuan. Languagebind: Extending video-language pretraining to n-modality by language-based semantic alignment. In *ICLR*, 2024. 8