Make Your Training Flexible: Towards Efficient Video Foundation Models

Anonymous ICCV submission

Paper ID 2101

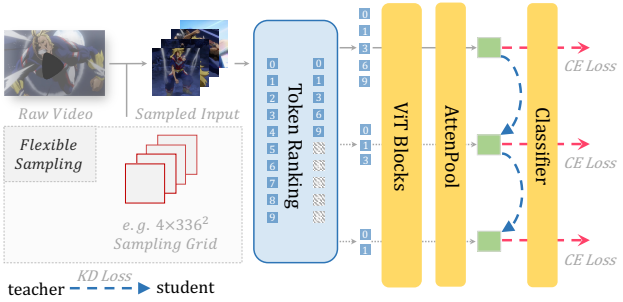

Figure 1. **Overview of Flux-Multi Tuning.**

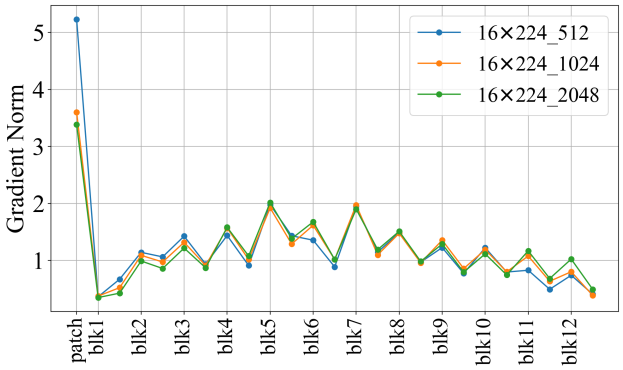

Figure 2. **Gradient norms of main projector modules of Flux-Multi trained InternVideo2 on K400.** We report the L2 gradient norm using bs=32.

1. More Experiments

1.1. More ablation studies

We here provide more ablation studies on Flux, including analyzing Flux's training stability with gradient norm, full results using different spatiotemporal resolutions and a corresponding heuristic TO validation strategy, and convergence analysis and experimenting with possible token merging methods in Flux. This will provide a more in-depth analysis of the whole Flux method.

Training dynamics and convergence analysis As illustrated in Figure 2, we analyze the gradient norms across the main projector layers in the Flux-Multi Tuned InternVideo2-S. The Patch Embedding Layer exhibits notably elevated gradient norm values, particularly when processing higher input settings with a smaller number of input tokens. This gradient magnitude disparity could potentially introduce training instability. Considering this and to generate more stable token-selection masks as introduced before, we use the Dual Patch Norm module, which shares

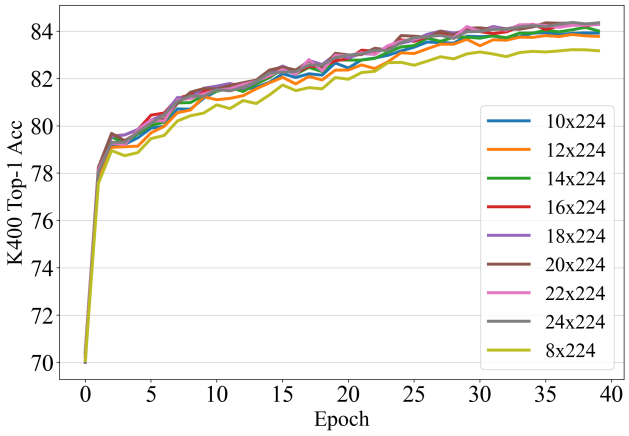

Figure 3. **Convergence analysis of Flux-Single tuning using 3072 tokens but different frame counts directly on K400.**

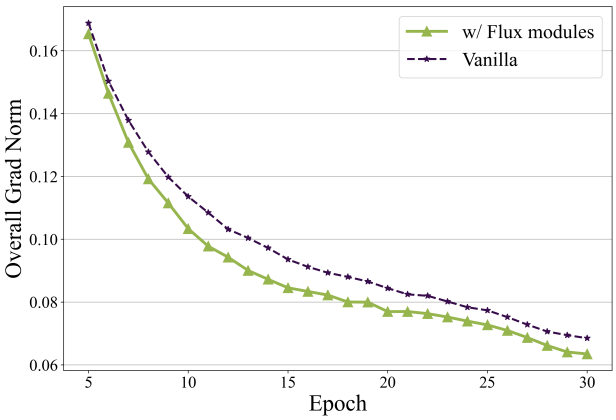

Figure 4. **Overall gradient norm trend during Flux-UMT pertraining.** We report the overall training dynamics with our ablation setting. The FluxViT modules can lower the overall norm.

#Frame	Spatial Resolution					Max
	168	196	224	252	280	
4	80.4	81.7	82.3	**82.6**	82.3	82.6
6	83.5	**84.5**	84.3	84.2	83.6	84.5
8	84.4	**84.8**	84.6	84.4	83.7	84.8
10	**85.2**	85.1	85.0	84.5	83.5	85.2
12	**85.3**	85.3	84.9	84.4	83.4	85.3
16	**85.3**	85.1	84.8	84.4	83.5	85.3
20	**85.1**	85.0	84.6	84.0	83.2	85.1
Max	**85.3**	**85.3**	**85.0**	**84.5**	**83.7**	-

Table 1. **Results of FluxViT-S on K400 using 1024 tokens and different spatiotemporal resolutions.** We use 1clip × 1crop for testing. The blue value marks the results of the unmasked setting. The values in bold show the best resolution for each frame count.

Method	Input Size	#Token	
		1024	512
Our selector	4×224^2	82.3	79.5
	8×224^2	84.6	81.3
	12×224^2	84.9	80.7
	16×224^2	84.8	80.7
	20×224^2	84.6	80.3
	24×224^2	84.6	80.3
	Max	84.9	81.3
***w/* Vid-TLDR**	4×224^2	∅	77.4 78.2
	8×224^2	83.9	81.0 79.8
	12×224^2	85.0	81.4 80.6
	16×224^2	85.3	81.5 80.9
	20×224^2	85.2	80.9 80.4
	24×224^2	85.2	80.5 80.3
	Max	85.3	81.5 80.9

Table 2. **Use token merging strategy Vid-TLDR [6] on FluxViT K400 testing.** The increment achieved by ViD-TLDR is sensitive to the hyper-parameter setting, like how many tokens are to be reduced in certain layers.

similar findings with DPN[13]. However, our convergence analysis, presented in Figure 3, reveals that our Flux-Single tuning with InternVideo2-S, utilizing 3072 tokens and direct tuning over 40 epochs on the K400 dataset, demonstrates consistent convergence patterns. Specifically, configurations with varying frame counts but a fixed token number exhibit normal convergence behavior during tuning. This observation suggests that the aforementioned instability may not be the primary concern in Flux training. Consequently, we prioritize DPN's capability to generate robust masks over its stabilization properties in fine-tuning scenarios and prioritize DPN's capability in stabilized training in the pre-training stage, considering the results in Figure 4 and the results in the previous Flux-UMT ablation study.

Full results using different spatiotemporal resolutions. Table 1 shows the results of FluxViT-S on K400 using different spatiotemporal resolutions but with a kept 1024 number. Using lower spatial resolution but with a larger frame count can further strengthen the model's performance, which causes another +0.3% performance gain compared with the best result achieved using standard 224 resolution. This may reflect the dataset's bias towards longer inputs and our method's preference for more dynamic tokens instead of highly informative spatial tokens. Moreover, we find that the best-performing areas are mainly located within a threshold of input tokens, which can be seen as the bolded values of each frame count mainly located within an anti-diagonal line. **This observation validates our approach of imposing a threshold on input token numbers and suggests an optimized evaluation strategy for determining optimal input configurations.** We propose a systematic evaluation procedure: beginning with minimal input settings (e.g., 4×224^2), incrementally increase frame counts until performance plateaus, then progressively reduce spatial resolution while increasing frame count until accuracy improvements cease. This linear complexity evaluation approach efficiently identifies the near-optimal configuration for token optimization.

Combining modern token merging strategy. The integration of state-of-the-art training-free token-merging strategies during inference presents an opportunity to further enhance our Flux method's performance. Table 2 demonstrates the performance achieved by incorporating the advanced Vid-TLDR [6] token reduction approach with our FluxViT-S model on K400. Vid-TLDR implements token merging within the initial network blocks to regulate token count. For configurations targeting 1024 tokens, we apply Vid-TLDR to progressively reduce token counts to [2048, 1536, 1024] across the first three layers. Similarly, for 512-token configurations, we evaluate two reduction sequences: [1024, 512] and an alternative [1536, 1024, 512] (denoted by gray values in the table). While results from the first two reduction strategies demonstrate the potential synergy between advanced token-selection methods and our Flux approach, the latter sequence, despite incorporating more tokens in initial layers and involving more computation overhead, underperforms our baseline that employs a heuristic token-reduction method. This outcome shows Vid-TLDR's limitations in accommodating diverse token reduction requirements and highlights its ongoing need for extensive parameter searching. Thus, we only adopt our heuristic but nearly costless token selection method instead of the heavy, nonflexible, and unstable token merging method in Flux.

1.2. More results

Full retrieval results. Table 3 shows more zero-shot retrieval results on MSRVTT [29], DiDeMo [1], ActivityNet [9], LSMDC [23], and MSVD [5]. We see that MSRVTT and ActivityNet enjoy only marginal performance gain using 2048 tokens, which may be due to the information saturation for these datasets as also observed in InternVideo2 [28] when finding little gain by enlarging the frame count from 4 to 8 and 16. The other three datasets

Method	#Token	Type	MSRVTT			DiDeMo			ActivityNet			LSMDC			MSVD		
			R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
FluxViT-S	2048	T2V	44.4	67.0	75.6	48.3	74.4	82.3	52.4	79.0	87.5	20.8	36.0	44.2	49.3	77.7	85.5
		V2T	44.3	67.7	77.9	50.4	75.1	83.2	53.0	79.4	88.4	21.6	37.6	45.5	78.7	92.8	95.4
	1024	T2V	42.2	64.4	74.0	45.4	72.1	79.5	47.2	73.9	84.8	18.7	35.9	43.8	47.3	76.9	84.4
		V2T	43.1	65.7	74.6	47.0	71.6	80.6	48.0	75.1	85.1	20.3	37.2	45.4	79.3	92.5	95.7
	512	T2V	36.8	59.5	69.6	38.5	65.7	74.7	38.2	65.2	76.1	17.2	33.0	41.7	45.1	74.0	82.3
		V2T	37.0	61.2	70.2	40.0	65.7	75.2	38.5	64.3	76.5	17.8	33.8	41.1	75.5	90.5	93.6
FluxViT-S+	2048	T2V	45.0	67.5	75.8	49.2	74.5	82.8	52.4	79.0	87.5	21.1	38.2	46.0	49.7	77.8	85.8
		V2T	44.9	68.2	76.5	51.2	74.9	82.9	53.8	78.0	89.2	22.4	38.6	46.4	80.2	93.6	95.5
	1024	T2V	44.5	66.4	74.6	49.0	73.9	82.4	50.3	76.9	86.4	20.5	36.6	44.8	49.1	77.0	85.5
		V2T	44.2	67.4	76.4	50.5	74.1	82.4	50.9	77.8	87.3	21.7	38.5	45.3	80.2	92.2	94.5
	512	T2V	40.5	62.7	71.7	45.8	71.4	80.5	44.7	71.8	82.5	19.0	34.3	40.8	46.9	76.2	84.0
		V2T	41.1	63.4	73.1	47.0	71.6	79.4	44.7	71.8	82.8	19.2	35.0	41.7	78.1	90.0	93.6
FluxViT-B	2048	T2V	49.8	72.2	80.1	52.2	77.5	84.5	56.6	81.5	89.6	23.7	41.0	49.3	52.6	80.1	86.7
		V2T	49.3	73.6	81.5	53.0	78.9	86.7	57.6	82.9	91.3	24.8	42.0	49.3	83.3	94.2	96.6
	1024	T2V	48.0	69.6	78.0	48.8	75.5	82.6	51.8	78.4	87.5	22.6	39.9	48.3	51.9	79.5	86.2
		V2T	46.5	70.5	78.0	50.5	76.4	83.5	53.4	79.7	88.4	24.0	41.4	49.1	83.0	94.8	96.9
	512	T2V	42.6	64.4	73.7	42.9	68.9	77.7	42.8	69.3	79.9	20.1	36.6	45.3	49.6	77.6	84.9
		V2T	41.5	65.2	74.0	44.5	70.3	77.9	43.3	70.1	80.6	21.4	37.5	46.0	80.9	92.8	95.4
FluxViT-B+	2048	T2V	49.9	71.0	79.6	53.5	77.3	86.1	56.7	81.6	89.9	25.4	41.7	50.5	54.2	80.9	88.0
		V2T	49.4	73.9	82.4	54.2	78.6	86.8	58.3	83.3	91.4	25.6	42.6	50.4	84.2	93.9	96.6
	1024	T2V	49.1	71.4	79.3	53.0	77.4	84.4	55.2	80.8	88.6	24.1	40.9	49.5	53.4	80.7	87.9
		V2T	48.9	71.4	79.9	54.3	78.6	86.1	57.0	82.2	90.4	25.3	42.8	50.3	84.9	93.9	96.9
	512	T2V	47.2	68.6	77.0	49.8	74.6	82.4	50.3	76.0	85.0	22.5	38.2	47.1	52.1	79.3	86.7
		V2T	46.5	71.1	78.6	51.2	75.5	83.4	50.9	76.7	85.6	23.0	40.3	47.6	83.0	93.9	96.4

Table 3. **Full Zero-shot retrieval results on MSRVTT, DiDeMo, AcitivityNet, LSMDC, and MSVD.**

Method	K400		K600		UCF101	MiTv1
	Top1	Top5	Top1	Top5		
FluxViT-S_{2048}	66.7	88.5	65.2	87.3	85.8	28.2
FluxViT-S_{2048}+	67.0	88.8	65.5	87.5	87.5	28.6
FluxViT-S_{1024}	64.2	86.6	62.8	85.7	85.1	27.2
FluxViT-S_{1024}+	65.6	87.9	64.3	86.5	87.2	27.8
FluxViT-S_{512}	59.0	82.6	57.4	81.2	81.5	25.5
FluxViT-S_{512}+	62.6	85.5	61.1	84.2	84.2	26.5
FluxViT-B_{2048}	70.2	90.6	68.9	89.5	88.7	31.2
FluxViT-B_{2048}+	70.7	90.9	69.3	89.8	89.1	31.5
FluxViT-B_{1024}	68.6	89.6	67.2	88.4	87.8	30.4
FluxViT-B_{1024}+	69.6	90.1	68.3	89.0	89.1	31.0
FluxViT-B_{512}	64.2	86.1	62.5	84.9	84.9	28.8
FluxViT-B_{512}+	67.4	87.4	65.8	87.7	87.6	29.8

Table 4. **Full Zero-shot Action Recognition Results.**

config	SthSth V2	Others
optimizer	AdamW [17]	
optimizer momentum	β_1, β_2=0.9, 0.98	
weight decay	0.05	
learning rate schedule	cosine decay [18]	
learning rate	1e-3	
batch size	2048	
warmup epochs [8]	20	
total epochs	100	
teacher input token	2048	
student input tokens	2048, 1536, 1024	
input frame	(4, 26, stride=2)	
spatial resolution	(168, 280, stride=28)	
drop path [10]	0.05	
flip augmentation	*no*	*yes*
augmentation	MultiScaleCrop [0.66, 0.75, 0.875, 1]	

Table 5. **Flux-UMT pre-training settings.**

highlight our Flux method's effects more with 2048 tokens, while all these datasets demonstrate our costless performance improvement with 1024 and 512 token inputs.

More zero-shot action recognition results. Table 4 shows more zero-shot retrieval results of our FluxViT on K400 [11], K600 [3], UCF101 [25], and MiTv1 [20].

2. More implementation details

In this section, we introduce the detailed training hyperparameters and report the training dataset details in Table 8.

Flux-UMT pre-training. In combining Flux and UMT[15] framework to get our single modality FluxViT model, we follow most settings as used in deriving InternVideo2 models. Details are shown in Table 5.

Single modality fine-tuning. We adopt the Flux-UMT pre-trained video encoder and add an extra classification layer for fine-tuning. Input settings are kept the same, and the details of hyperparameters are given in Table 6.

Flux-CLIP per-training. In combining Flux and CLIP[22] framework to get our multi-modality FluxViT model, we show the details in Table 7. We freeze all the

config	Kinetics	COIN
optimizer	AdamW [17]	
optimizer momentum	β_1, β_2=0.9, 0.999	
weight decay	0.05	
learning rate schedule	cosine decay [18]	
learning rate	2e-4	5e-4
batch size	1024+512	512
warmup epochs [8]	5+1	5
total epochs	35+5 (S), 20+3 (B)	40(S), 25 (B)
drop path [10]	0.1	
flip augmentation	*yes*	
label smoothing [26]	0.0	
augmentation	RandAug(9, 0.5) [7]	

Table 6. **Action recognition fine-tuning settings.** The training epochs A+B on Kinetics include A epochs on K710 and B epochs on K400, the same notation for warmup-epochs and batch size.

config	25M+2.5M
optimizer	AdamW [17]
optimizer momentum	β_1, β_2=0.9, 0.98
weight decay	0.02
learning rate schedule	cosine decay [18]
learning rate	4e-4 (25M), 2e-5 (2.5M)
batch size	4096 (image), 4096 (video)†
warmup epochs [8]	0.6 (25M), 0 (2.5M)
total epochs	3 (25M), 1 (2.5M)
input frame	(4, 26, stride=2)
spatial resolution	(168, 280, stride=28)
token threshold	(2048, 4096)
augmentation	MultiScaleCrop [0.5, 1]

Table 7. **Flux-CLIP pre-training settings.** †: For FluxViT-B, we lower the batch size to 2048 for the 2.5M data training.

Dataset	**#image/video**	**#text**	**Type**
Kinetics-710 [14]	658K	0	Video
COCO [16]	113K	567K	image
Visual Genome [12]	100K	768K	image
SBU Captions [21]	860K	860K	image
CC3M [24]	2.88M	2.88M	image
CC12M [4]	11.00M	11.00M	image
S-MiT0.5M [19]	0.5M	0.5M	video
WebVid-2M [2]	2.49M	2.49M	video
WebVid-10M [2]	10.73M	10.73M	video
InternVid2M [27]	2.0M	2.0M	video
25M corpus = CC3M+CC12M +WebVid-10M+Visual Genome +SBU+COCO	25.68M	26.81M	video + image
2.5M corpus = S-MiT+InternVid2M+COCO	2.56M	2.62M	video + image

Table 8. **Statistics of pre-training datasets.**

modules in 25M data pretraining as Stage 1, except the vision projector. We unfreeze all the modules for the Stage 2 training on the 2.5M dataset.

Chat Centric Training We freeze both the LLM and the Vision Encoder in the common stage-1 training of a chat model. We use a learning rate of 1e-3, a batch size of 512, a single training epoch, and a cosine learning rate schedule with a 0.03 warmup rate.

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] 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. 4

[3] João Carreira, Eric Noland, Andras Banki-Horvath, Chloe Hillier, and Andrew Zisserman. A short note about kinetics-600. *ArXiv*, abs/1808.01340, 2018. 3

[4] 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. 4

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

[6] 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. 2

[7] Ekin Dogus Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V. Le. Randaugment: Practical automated data augmentation with a reduced search space. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020. 4

[8] Priya Goyal, Piotr Dollár, Ross B. Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour. *ArXiv*, abs/1706.02677, 2017. 3, 4

[9] 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

[10] Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q Weinberger. Deep networks with stochastic depth. In *ECCV*, 2016. 3, 4

[11] 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. 3

[12] 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. 4

[13] Manoj Kumar, Mostafa Dehghani, and Neil Houlsby. Dual patchnorm. *ArXiv*, abs/2302.01327, 2023. 2

[14] 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. 4

[15] 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. 3

[16] 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. 4

[17] I. Loshchilov and F. Hutter. Fixing weight decay regularization in adam. *ArXiv*, abs/1711.05101, 2017. 3, 4

[18] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *ICLR*, 2017. 3, 4

[19] Mathew Monfort and SouYoung Jin. Spoken moments: Learning joint audio-visual representations from video descriptions. In *CVPR*, 2021. 4

[20] 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. 3

[21] Vicente Ordonez, Girish Kulkarni, and Tamara Berg. Im2text: Describing images using 1 million captioned photographs. In *NeurIPS*, 2011. 4

[22] 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

[23] 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

[24] 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. 4

[25] K Soomro. Ucf101: A dataset of 101 human actions classes from videos in the wild. *ArXiv*, abs/1212.0402, 2012. 3

[26] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, 2016. 4

[27] 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. 4

[28] 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. 2

[29] 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