VideoPrism: A Foundational Visual Encoder for Video Understanding

Long Zhao [*] **Nitesh B. Gundavarapu** [*] **Liangzhe Yuan** [*] **Hao Zhou** [*] **Shen Yan** [†] **Jennifer J. Sun** [†]
Luke Friedman [†] **Rui Qian** [†] **Tobias Weyand** **Yue Zhao** [§] **Rachel Hornung** **Florian Schroff** **Ming-Hsuan Yang**
David A. Ross **Huisheng Wang** **Hartwig Adam** **Mikhail Sirotenko** [‡] **Ting Liu** [‡] **Boqing Gong** [‡]

Google

Abstract

We introduce VideoPrism, a general-purpose video encoder that tackles diverse video understanding tasks with a single frozen model. We pretrain VideoPrism on a heterogeneous corpus containing 36M high-quality video-caption pairs and 582M video clips with noisy parallel text (*e.g.*, ASR transcripts). The pretraining approach improves upon masked autoencoding by global-local distillation of semantic video embeddings and a token shuffling scheme, enabling VideoPrism to focus primarily on the video modality while leveraging the invaluable text associated with videos. We extensively test VideoPrism on four broad groups of video understanding tasks, from web video question answering to CV for science, achieving state-of-the-art performance on 31 out of 33 video understanding benchmarks.

1. Introduction

Videos are a rich and dynamic archive of real-world perceptual experience, spanning diverse domains from everyday life to scientific observations. Video foundation models (ViFMs) hold enormous potential to unlock new insights within this vast corpus. While prior work has made great progress towards general video understanding (Xu et al., 2021; Wang et al., 2022c; Yan et al., 2022; Tong et al., 2022; Li et al., 2023b; Wang et al., 2023c), building a truly foundational video model is still an elusive goal. Existing models often struggle to balance appearance-heavy tasks with motion-centric reasoning, falling behind task-specialized models across many benchmarks (Yuan et al., 2023).

We introduce VideoPrism, a general-purpose video encoder designed to tackle a wide spectrum of video understanding tasks, including classification, localization, retrieval, captioning, and question answering (QA) (Figure 1). Evaluated extensively on computer vision (CV) datasets and CV for science domains like neuroscience and ecology, VideoPrism achieves state-of-the-art performance with minimal adaptation, using a *single frozen* model. We emphasize this frozen-encoder setting following prior work (Radford et al., 2021; Alayrac et al., 2022; Tang et al., 2023; Li et al., 2023a) and for its practical utility given the otherwise high computational and memory cost of finetuning video models.

The design philosophy behind VideoPrism is as follows. Pretraining data is fundamental to foundation models (FMs) (Bommasani et al., 2021), and the ideal pretraining data for ViFMs would be a representative sample of all videos in the world. Most videos from this sample will have no (or very noisy) parallel text describing the content; however, when such text exists, it provides priceless semantic clues about the video space. Accordingly, our pretraining strategy should focus primarily on the video modality and yet take full advantage of any available video-text pairs.

On the data side, we approximate the desired pretraining corpus by assembling 36M high-quality video-caption pairs and 582M video clips with noisy parallel text (*e.g.*, ASR transcripts, generated captions, and retrieved text). On the modeling side, we first contrastively learn semantic video embeddings (Radford et al., 2021; Jia et al., 2021) from all our video-text pairs of various qualities. Subsequently, we capitalize on the extensive video-only data by distilling the semantic embeddings globally and token-wise, improving upon masked video modeling (Tong et al., 2022; Feichtenhofer et al., 2022; Wang et al., 2023c) described below.

Despite its success for natural language (Devlin et al., 2019; Brown et al., 2020; Anil et al., 2023), masked data modeling remains challenging for CV as raw visual signals lack semantics. Existing works approach this challenge by bor-

[*]Equal primary contribution. [†]Equal core technical contribution. [‡]Equal senior contribution, project leads. [§]This work was done while the author was a student researcher at Google Research. Correspondence to: Long Zhao <longzh@google.com>, Mikhail Sirotenko <msirotenko@google.com>, Ting Liu <liuti@google.com>, Boqing Gong <bgong@google.com>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

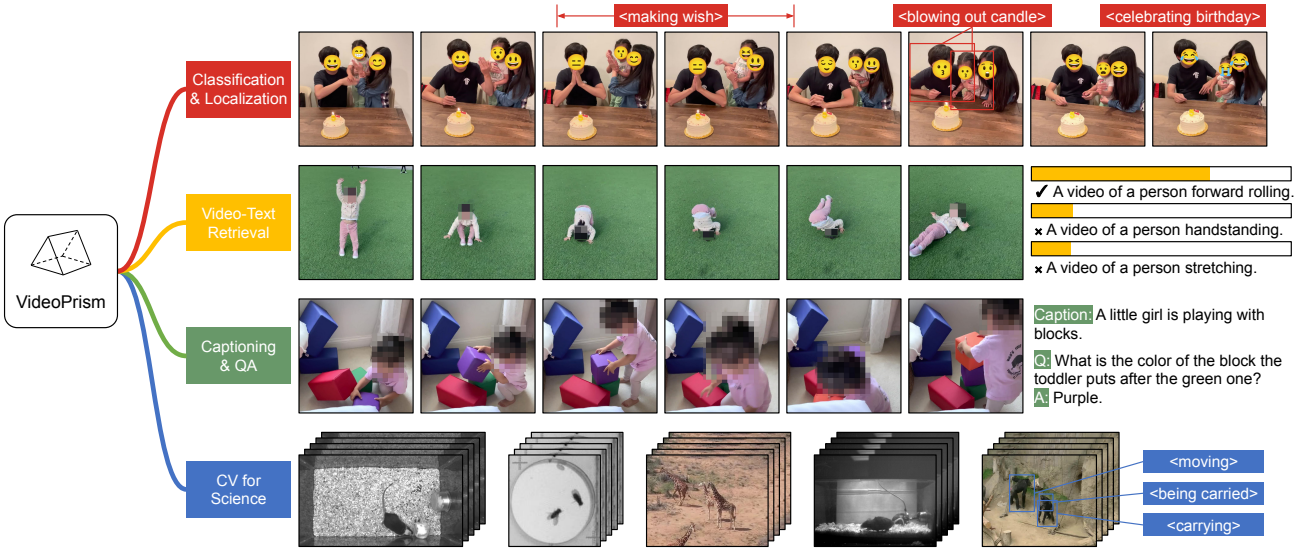


Figure 1. **VideoPrism** is a general-purpose video encoder that enables state-of-the-art results over a wide spectrum of video understanding tasks by producing video representations from one *single frozen* model.

rowing indirect semantics (*e.g.*, using CLIP (Radford et al., 2021) to bootstrap models (Fang et al., 2022; 2023) or tokenizers (Peng et al., 2022)) or implicitly promoting them (*e.g.*, tokenizing visual patches (Zhou et al., 2022; Bao et al., 2022; Oquab et al., 2023), combining a high masking ratio and lightweight decoder (He et al., 2022)).

We build on the above ideas with a two-stage approach tailored to our pretraining data. We first train a video encoder, along with a paired text encoder, over the video-text pairs using a contrastive objective (Gutmann & Hyvärinen, 2010; Radford et al., 2021). Next, we continue training the encoder over all video-only data by masked video modeling with two improvements: (1) the model is required to predict both the video-level global embedding and token-wise embeddings from the first stage based on unmasked input video patches; (2) random shuffle is applied to the encoder's output tokens before they are passed to the decoder to avoid learning shortcuts. Notably, our pretraining utilizes two supervisory signals: a video's text description and its contextual self-supervision, enabling VideoPrism to excel on both appearance- and motion-focused tasks. Indeed, previous works have shown that video captions mainly reveal appearance cues (Wang et al., 2023f), and contextual self-supervision facilitates learning motion (Tong et al., 2022).

Contributions. VideoPrism is a state-of-the-art, general-purpose video encoder. We advocate for a scalable strategy for collecting pretraining videos, combining manually captioned videos with those containing noisy textual descriptions. We design a unique two-stage pretraining approach

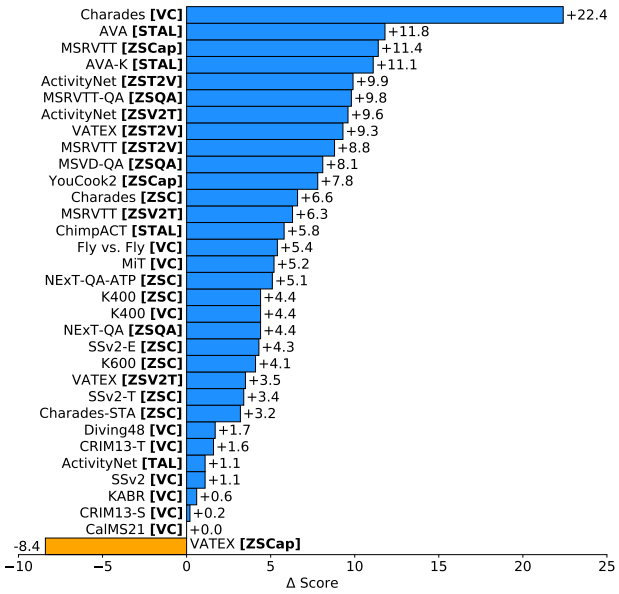


Figure 2. **VideoPrism *vs.* the previous best-performing FMs.** Please find the details of this figure in Appendix D.

tailored to this hybrid data, leveraging video-language contrastive learning to harvest semantics, followed by improved masked video modeling with global-local distillation and token shuffling. Finally, we present a comprehensive evaluation on four broad categories of understanding tasks across 33 diverse benchmarks, including videos from the web, scripted performances, and scientific experiments. Results demonstrate that VideoPrism significantly outperforms ex-

Table 1. **Composition of our pretraining corpus.** We report the numbers of videos and clips we were able to access during pretraining.

Pretraining datasets	Public	Domain	Caption source	Caption quality	# of videos	# of clips
Anonymous-Corpus #1	✗	Web video	Manual labelled	High	36.1M	36.1M
WTS-70M (Stroud et al., 2020)	✓	YouTube video	Metadata	Low	55.1M	55.1M
YT-Temporal-180M (Zellers et al., 2021)	✓	YouTube video	ASR	Low	2.3M	87.8M
VideoCC (Nagrani et al., 2022)	✗	YouTube video	Image captions for mining	Low	133.5M	191.1M
InternVid (Wang et al., 2023e)	✓	YouTube video	Generated by VLM/LLM	Medium	2.8M	7.0M
Anonymous-Corpus #2	✗	YouTube video	ASR	Low	44.6M	170.3M
Anonymous-Corpus #3	✗	YouTube video	Generated by VLM/LLM	Medium	36.7M	71.5M

isting ViFMs on 31 benchmarks (Figure 2). Importantly, no single baseline model consistently achieves second-best performance, indicating VideoPrism's robust generalizability.

2. Approach

2.1. Pretraining data

Our pretraining data consists of 36M clips (sampled from 36M videos) with high-quality manually labelled *captions* and 582M clips (from 275M videos) with noisy parallel *text*, as summarized in Table 1. The 36M high-quality video-caption pairs in Anonymous-Corpus #1 are the largest of its kind for ViFMs, to our knowledge, but they are still an order of magnitude smaller than the image-language data used to fuel image FMs (Radford et al., 2021; Yu et al., 2022). Hence, we also collect large-scale video-text data whose noisy text is generated through ASR, metadata, and large multimodal models (Wang et al., 2023e; Zhao et al., 2024), *etc.* This subset of videos corresponds to the rows from WTS-70M to Anonymous-Corpus #3 in Table 1, and we provide more details in Appendix A.

Importantly, unlike previous works (Tong et al., 2022; Wang et al., 2022c; Li et al., 2023b; Wang et al., 2023b), we do not incorporate any training sets from the evaluation benchmarks, *e.g.*, Kinetics (Kay et al., 2017), for either pretraining or post-pretraining. This conscious choice avoids overly tuning our model towards certain evaluation benchmarks. Moreover, we carefully de-duplicate the pretraining corpus against the videos in all the 33 evaluation benchmarks used in this paper to ensure that there is no data leakage.

2.2. Model architecture

The VideoPrism model architecture stems from the standard Vision Transformer (ViT) (Dosovitskiy et al., 2021), with a factorized design in space and time following ViViT (Arnab et al., 2021). However, we remove the global average pooling layer of ViViT immediately after the spatial encoder so that the spatiotemporal dimensions remain in the output token sequence, facilitating the downstream tasks that require fine-grained features (*e.g.*, spatiotemporal action localization). We experiment with two model configurations: *VideoPrism-g* and *VideoPrism-B*. VideoPrism-g is the ViT-giant network (Zhai et al., 2022a) with 1B parameters in the spatial encoder, and VideoPrism-B is a smaller variant with the ViT-Base network (Dosovitskiy et al., 2021). Appendix B describes the two network architectures in detail.

2.3. Training algorithm

Our goal is to leverage both video-text pairs and the video-only data curated in Section 2.1 to train VideoPrism scalably, so as to make VideoPrism a foundational video encoder capable of capturing both appearance and motion semantics from videos. We highlight the video-only modality rather than solely relying on video-text because the text in our large-scale pretraining corpus is very noisy for a majority of the videos. As shown in Figure 3, the training pipeline of VideoPrism consists of two stages: *video-text contrastive training* and *masked video modeling*.

2.3.1. Stage 1: Video-text contrastive training

In the first stage, we conduct contrastive learning to align a video encoder with a text encoder using all the video-text pairs. Following prior arts (Radford et al., 2021; Jia et al., 2021; Cheng et al., 2023), we minimize a symmetric cross-entropy loss over the similarity scores of all video-text pairs in a mini-batch, initialize the spatial encoding modules using the image model of CoCa (Yu et al., 2022), and include WebLI (Chen et al., 2023c) (about 1B images with alt-text) to the pretraining. The video encoder's features are aggregated through a multi-head attention pooler (MAP) (Lee et al., 2019) before the loss computation. This stage allows the video encoder to learn rich visual semantics from language supervision, and the resulting model supplies semantic video embeddings for the second-stage training.

2.3.2. Stage 2: Masked video modeling

Training solely on vision-text data as in Stage 1 presents challenges: text descriptions can be noisy, and they often capture appearance more than motion (Hendricks & Nematzadeh, 2021; Momeni et al., 2023). To address this, our second-stage training focuses on learning both appearance and motion information from video-only data. Building

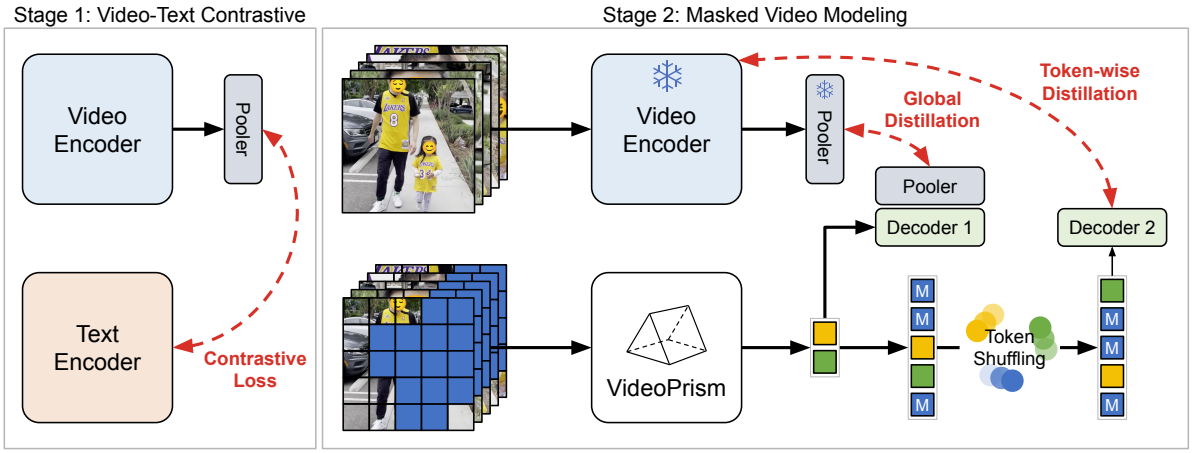


Figure 3. **Illustration of our two-stage pretraining.** Stage 1 trains video and text encoders with contrastive loss on video-text pairs, supplying semantic video embeddings to the next stage. Stage 2 continues to train the video encoder, now called VideoPrism, using improved masked autoencoding on video-only clips. The frozen Stage-1 encoder uses unmasked 3D video patches to produce a global semantic embedding of the whole video and token-wise embeddings. Decoder 2 processes shuffled tokens with positional embedding, while Decoder 1 has no positional embedding.

upon the success of masked autoencoding for motion understanding (Wang et al., 2022c; 2023c), we adapt this approach for the second stage, while ensuring that the model retains the semantic knowledge acquired in the first stage.

In this stage, we continue to train the video encoder on video-only data using improved masked video modeling. These improvements include (1) a novel token shuffling scheme to prevent decoding shortcuts and (2) global and token-wise distillation losses to effectively leverage the knowledge acquired in the first stage. As illustrated in Figure 3, the second-stage (student) model learns to predict the first-stage (teacher) model's embeddings of *all* tokens based on a masked video. The encoder-decoder Transformers are decoupled following He et al. (2022)'s design.

Token shuffling. As we effectively initialize the second-stage model from the first stage, one issue is that the model may create a shortcut for the decoder to copy and paste the unmasked tokens while predicting only the masked ones, making it an easier task to solve than predicting all tokens. To address this issue, we *randomly shuffle* the token sequence output by the encoder before feeding it to the decoder, and the decoder adds positional embeddings to this sequence after the shuffling. Note that this shuffling operation avoids the copy-and-paste shortcut of unmasked tokens that the decoder can potentially explore. One can also view it akin to Jigsaw puzzles (Noroozi & Favaro, 2016) which the decoder tries to solve for the unmasked tokens while it predicts the masked ones.

Global-local distillation. Unlike the masked distillation for images (Fang et al., 2022; 2023), we find that our second-stage model underperforms the first-stage teacher on appearance-heavy tasks when only the masked modeling loss is utilized, probably attributing to catastrophic forgetting (McCloskey & Cohen, 1989) in the two-stage pretraining. To mitigate this issue, we add an additional loss to let the second-stage model distill the global embedding of the full intact video from the first-stage teacher using the visible tokens. Hence, the second-stage training loss combines the token-wise masked video modeling and global distillation. Due to space limit, we refer readers to Appendix C for the detailed implementation and training configurations.

3. Experiments

We evaluate VideoPrism on a wide spectrum of video-centric understanding tasks to demonstrate its capability and generalizability. We group the tasks into four categories: (1) general video-only understanding, including classification and spatiotemporal localization (Section 3.1), (2) zero-shot video-text retrieval (Section 3.2), (3) zero-shot video captioning and QA (Section 3.3), and (4) CV for science (Section 3.4). For all experiments in the main paper, we freeze VideoPrism as a video encoder and only train task-specific components for the tasks in groups (1), (2), and (4) and some adaptation layers connecting VideoPrism to an LLM for (3). In the appendices, we report more results of end-to-end and adapter finetuning. Note that our evaluation strategy, freezing the visual encoder, aligns with prior works (He et al., 2022; Singh et al., 2022; Yuan et al., 2023)

Table 2. **Evaluating FMs on the VideoGLUE benchmark (Yuan et al., 2023) with frozen backbones.** Only weights in the task heads are trained using the downstream tasks' training sets. On all video classification (VC) tasks except Charades, we report top-1 accuracy. On Charades, temporal localization (TAL), and spatiotemporal localization (STAL) tasks, we use mean average precision (mAP) as the evaluation metric. -B, -L, -g indicate that the underlying models are respectively the base, large, and giant ViT (Dosovitskiy et al., 2021).

Methods	VC (A)		VC (M)		VC (ML)	TAL	STAL	
	K400	**MiT**	**SSv2**	**D48**	**Charades**	**ActivityNet**	**AVA**	**AVA-K**
Base-scale models								
CLIP-B (Radford et al., 2021)	75.2	32.6	41.0	44.1	11.2	32.7	21.1	25.9
VATT-B (Akbari et al., 2021)	75.1	32.1	57.8	49.7	33.3	35.3	20.3	22.2
InternVideo-B (Wang et al., 2022c)	69.3	26.3	58.2	55.6	13.0	33.3	13.4	15.7
UMT-B (Li et al., 2023b)	77.1	34.0	47.7	47.8	30.1	35.8	20.7	21.1
VideoPrism-B	**84.2** (↑7.1)	**40.8** (↑6.8)	**63.6** (↑5.4)	**67.4** (↑12.)	**40.4** (↑7.1)	**36.6** (↑0.8)	**30.6** (↑9.5)	**31.8** (↑5.9)
Large-scale models								
VideoMAE-v2-g (Wang et al., 2023b)	82.1	35.0	56.1	60.5	22.4	35.3	21.5	23.3
InternVideo-L (Wang et al., 2022c)	78.6	33.7	67.4	69.6	20.9	35.9	20.8	21.3
UMT-L (Li et al., 2023b)	82.8	40.3	54.5	49.0	39.9	36.7	24.4	26.2
VideoPrism-g	**87.2** (↑4.4)	**45.5** (↑5.2)	**68.5** (↑1.1)	**71.3** (↑1.7)	**62.3** (↑22.)	**37.8** (↑1.1)	**36.2** (↑12.)	**37.3** (↑11.)

and is almost a go-to choice for building VideoLLMs (Tang et al., 2023). It is especially needed for videos because finetuning a ViFM is prohibitively expensive, while a frozen ViFM allows one to amortize the cost of video encoding across multiple tasks. All results in the main text are produced using the same frozen VideoPrism-B/g checkpoint, corresponding to the base/giant model.

3.1. Classification and spatiotemporal localization

We compare VideoPrism with state-of-the-art FMs on a video-only understanding benchmark: VideoGLUE (Yuan et al., 2023). By design, VideoGLUE evaluates FMs through four adaptation methods over eight hallmark datasets, representing appearance-focused action recognition (VC (A)), motion-rich action recognition (VC (M)), multi-label video classification (VC (ML)), temporal action localization (TAL), and spatiotemporal action localization (STAL). This benchmark introduces a VideoGLUE score (VGS), considering the tradeoff between adaptation costs and performance, to provide a holistic view of FMs' capabilities on the video-only understanding tasks. We present the frozen-backbone evaluation results in the main paper and leave the rest to Appendix E. We employ an MAP head (Yuan et al., 2023) in action recognition (MAP probing) and spatiotemporal localization and use G-TAD (Xu et al., 2020) for temporal localization (see Appendix E.1 for details).

Datasets. The eight datasets in VideoGLUE are as follows. For apperance-focused action recognition, Kinetics-400 (K400) (Kay et al., 2017) and Moments-in-Time (MiT) (Monfort et al., 2019) are sourced from web videos. Something-Something v2 (SSv2) (Goyal et al., 2017a) and Diving48 (D48) (Li et al., 2018) are fine-grained motion-rich action recognition datasets. Besides, Charades (Sigurdsson et al., 2016) provides a multi-label classification problem using scripted indoor videos. The temporal localization task entails one dataset, ActivityNet v1.3 (Caba Heilbron et al., 2015), and the spatiotemporal localization contains Atomic Visual Actions (AVA) (Gu et al., 2018) and AVA-Kinetics (AVA-K) (Li et al., 2020).

Main results. Table 2 shows the frozen-backbone results on VideoGLUE. VideoPrism outperforms the baselines on all datasets by a large margin. Besides, increasing VideoPrism's underlying model size from ViT-B to ViT-g significantly improves the performance. Notably, no baselines can perform second best on all benchmarks, indicating the previous methods might be developed towards certain aspects of video understanding, while VideoPrism consistently improves on this wide range of tasks. This result implies that VideoPrism packed various video signals into one encoder: semantics at multiple granularities, appearance *vs.* motion cues, spatiotemporal information, and robustness to diverse video sources (*e.g.*, web videos *vs.* scripted performance).

In Appendix E.3, following the VideoGLUE setup, we conduct experiments on other adaptation methods, including end-to-end and parameter-efficient finetuning, and multi-layer attention pooling. Different adaption methods trade off computational cost with performance, accounting for real-world application considerations, and the VGS aggregates them into a scalar value. VideoPrism achieves VGS 51.25, outperforming all baseline FMs in Table 17 and scoring 13.6% higher than the second best model (UMT).

3.2. Zero-shot video-text retrieval and classification

To enable zero-shot video-text retrieval and video classification capabilities of VideoPrism, we follow LiT (Zhai et al., 2022b) to learn a text encoder producing the text embeddings matched to their corresponding video embeddings out of VideoPrism. We choose the LiT text encoder to mirror the one in the first-stage training and attach an MAP head to the video encoder. The LiT tuning is over the same pretraining data from the first stage. More details are in Appendix F.1.

Table 3. **Results of zero-shot video-text retrieval**. We report the Recall@1 (R@1) and R@5 for all the benchmarks. Note that we follow the 1K-A split of MSRVTT produced by Bain et al. (2021) which contains 1,000 videos for testing. Please refer to Appendix F.3 for the results on the full split of MSRVTT proposed by Xu et al. (2016).

Methods	MSRVTT (1K-A)				VATEX				ActivityNet			
	Text → Video		Video → Text		Text → Video		Video → Text		Text → Video		Video → Text	
	R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5
CLIP-L (Radford et al., 2021)	35.0	-	32.3	-	45.2	-	59.2	-	25.2	-	20.7	-
Singularity-B (Lei et al., 2023)	34.0	56.7	-	-	-	-	-	-	30.6	55.6	-	-
VideoCoCa-g (Yan et al., 2022)	43.9	69.9	45.4	68.6	53.2	83.3	73.6	93.2	34.5	63.2	33.0	61.6
InternVideo-L (Wang et al., 2022c)	40.7	-	39.6	-	49.5	-	69.5	-	30.7	-	31.4	-
UMT-L (Li et al., 2023b)	42.6	64.4	38.6	59.8	-	-	-	-	42.8	69.6	40.7	67.6
VideoPrism-B	51.4 (↑7.5)	74.4 (↑4.5)	50.2 (↑4.8)	73.2 (↑4.6)	57.7 (↑4.5)	88.5 (↑5.2)	76.2 (↑2.6)	93.7 (↑0.5)	49.6 (↑6.8)	76.7 (↑7.1)	47.9 (↑7.2)	75.0 (↑7.4)
VideoPrism-g	**52.7** (↑8.8)	**77.2** (↑7.3)	**51.7** (↑6.3)	**75.2** (↑6.6)	**62.5** (↑9.3)	**91.0** (↑7.7)	**77.1** (↑3.5)	**95.6** (↑2.4)	**52.7** (↑9.9)	**79.4** (↑9.8)	**50.3** (↑9.6)	**77.1** (↑9.5)

Table 4. **Comparison to state-of-the-art results on zero-shot video classification**. Results are reported in Top-1/5 accuracy (%) on Kinetics-400 and Something-Something v2, multi-choice (MC) retrieval accuracy (%) on NExT-QA (ATP-Hard) and Charades-STA, and mean average precision (mAP) on Charades. In line with Ni et al. (2022), we follow the single-view evaluation protocol for simplicity. Models pretrained with extra modalities (*e.g.*, audio) in addition to vision and language are marked in gray.

(a) Kinetics-400

Methods	Top-1 Acc	Top-5 Acc
CoCa-g (Yu et al., 2022)	66.4	87.1
VideoCoCa-g (Yan et al., 2022)	72.0	90.5
Text4Vis-L (Wu et al., 2023)	61.0	-
ImageBind-H (Girdhar et al., 2023)	50.0	-
LanguageBind-L (Zhu et al., 2024)	64.0	-
IMP-MoE-L (Akbari et al., 2023)	77.0	-
VideoPrism-B	71.3 (↓0.7)	91.7 (↑1.2)
VideoPrism-g	**76.4** (↑4.4)	**94.3** (↑3.8)

(b) Something-Something v2

Methods	Temporal	Events
VideoCLIP-B (Xu et al., 2021)	9.8	6.4
CoCa-g (Yu et al., 2022)	13.4	10.4
VideoCoCa-g (Yan et al., 2022)	14.1	10.7
VNLI-L (Yarom et al., 2023)	14.6	10.4
VideoCon-L (Bansal et al., 2023)	15.2	11.4
ImageBind-H (Girdhar et al., 2023)	10.5	5.5
VideoPrism-B	16.1 (↑0.9)	11.9 (↑0.5)
VideoPrism-g	**18.6** (↑3.4)	**15.7** (↑4.3)

(c) NExT-QA (ATP-Hard)

Methods	MC Acc
CLIP-B (Radford et al., 2021)	23.8
ATP-B (Buch et al., 2022)	20.2
VideoCoCa-g (Yan et al., 2022)	25.2
TACT-B (Bagad et al., 2023)	27.6
ImageBind-H (Girdhar et al., 2023)	25.4
VideoPrism-B	31.3 (↑3.7)
VideoPrism-g	**32.7** (↑5.1)

(d) Charades

Methods	mAP
CLIP-B (Radford et al., 2021)	19.8
CLIP-Hitchhiker-B (Bain et al., 2022)	21.1
CoCa-g (Yu et al., 2022)	23.1
VideoCoCa-g (Yan et al., 2022)	25.8
MAXI-B (Lin et al., 2023b)	23.8
VideoPrism-B	29.2 (↑3.4)
VideoPrism-g	**32.4** (↑6.6)

(e) Charades-STA

Methods	MC Acc
CoCa-g (Yu et al., 2022)	46.1
VideoCoCa-g (Yan et al., 2022)	47.2
VideoPrism-B	50.0 (↑2.8)
VideoPrism-g	**50.4** (↑3.2)

Datasets. We evaluate VideoPrism's zero-shot video-text retrieval performance on three benchmarks: MSRVTT (Xu et al., 2016; Bain et al., 2021), VATEX (Wang et al., 2019), and ActivityNet (Krishna et al., 2017). For zero-shot video classification tasks, we experiment with Kinetics-400 (Kay et al., 2017), Charades (Sigurdsson et al., 2016), SSv2-Temporal and SSv2-Events (Sevilla-Lara et al., 2021; Bagad et al., 2023), and the ATP-Hard subset of NExT-QA (Buch et al., 2022). SSv2 and NExT-QA (ATP-Hard) focus on motion and temporal reasoning, respectively. Moreover, we adapt Charades-STA (Gao et al., 2017) to the zero-shot classification scenario by reformulating each of its samples in the test set into a multi-choice retrieval problem (see Appendix F.2 for more details). We report results following the standard evaluation metric for each benchmark.

Main results. Tables 3 and 4 summarize the results of video-text retrieval and video classification, respectively. VideoPrism sets the new state of the art on most benchmarks, and the gains over the prior arts are exceptionally substantial on the challenging datasets (*e.g.*, 9.5% on ActivityNet, 4.4% on SSv2-Events, and 6.6 mAP on Charades). Most results from our base-scale VideoPrism-B are actually better than those of existing larger-scale models. Additionally, VideoPrism is on par with or better than the models pretrained with in-domain data and extra modalities (*e.g.*, audios) in Table 4. These improvements in zero-shot retrieval and classification tasks present VideoPrism's strong generalization capabilities.

3.3. Zero-shot video captioning and QA

We further evaluate the inherent capabilities of VideoPrism on generative video-language tasks, *i.e.*, captioning and QA, where we pair VideoPrism with a language decoder, PaLM-2 (Anil et al., 2023). To connect the two models,

Table 5. **Comparison to state-of-the-art methods on zero-shot video captioning**. We report the CIDEr score for all benchmarks.

Methods	MSRVTT	VATEX	YouCook2
Captioning-only models			
VideoCoCa-g (Yan et al., 2022)	27.1	22.8	34.3
DeCap-B (Li et al., 2023d)	18.6	18.7	-
All-in-one models			
Flamingo-3B (Alayrac et al., 2022)	-	**40.1**	55.8
Flamingo-9B (Alayrac et al., 2022)	-	39.5	55.0
VideoPrism-B w/ PaLM-2-1B	**40.3** (↑13.)	24.2 (↓12.)	52.3 (↓3.5)
VideoPrism-B w/ PaLM-2-8B	38.5 (↑11.)	31.7 (↓8.4)	**63.6** (↑7.8)

Table 6. **Comparison to state-of-the-art methods on zero-shot video QA**. We report the WUPS index (Wu & Palmer, 1994) for NExT-QA and Top-1 accuracy for the others. Methods that unfreeze their language models are marked in gray.

Methods	MSRVTT-QA	MSVD-QA	NExT-QA
Question-answering-only models			
FrozenBiLM-L (Yang et al., 2022)	22.2	39.0	-
All-in-one models			
BLIP-B (Li et al., 2022)	19.2	35.2	-
HiTeA-B (Ye et al., 2023)	21.7	37.4	-
mPLUG-2 (Xu et al., 2023)	43.8	55.3	-
Flamingo-3B (Alayrac et al., 2022)	11.0	27.5	21.3
Flamingo-9B (Alayrac et al., 2022)	13.7	30.2	23.0
VideoPrism-B w/ PaLM-2-1B	28.5 (↑6.3)	39.5 (↑0.5)	23.8 (↑0.8)
VideoPrism-B w/ PaLM-2-8B	**32.0** (↑9.8)	**47.1** (↑8.1)	**27.4** (↑4.4)

we introduce and train several gluing layers while keeping both VideoPrism and the language decoder frozen. We then conduct evaluation under the zero-shot configuration on video captioning and QA benchmarks. Note that we do not tune our models separately for captioning and QA tasks. Please refer to Appendix G for implementation details.

Datasets. We evaluate the model in the zero-shot setting on the test splits of a suite of standard video captioning datasets including MSRVTT (Xu et al., 2016), VATEX (Wang et al., 2019), and YouCook2 (Zhou et al., 2018), and video QA benchmarks including MSRVTT-QA (Xu et al., 2017), MSVD-QA (Xu et al., 2017), and NExT-QA (Xiao et al., 2021). For video QA, where it is imperative to match the length and style of the model's answers with groundtruths, we adopt the zero-shot approach of Flamingo (Alayrac et al., 2022) and use two-shot text-only prompts from the training set of the downstream task. Additionally, for MSRVTT-QA and MSVD-QA, we experiment with the closed-vocabulary evaluation configuration (Li et al., 2022; Yang et al., 2022). In this setting, we let the model score candidate answers according to their log-likelihoods and return the top one.

Main results. Tables 5 and 6 show the results of zero-shot video captioning and QA, respectively. Despite the simplicity of our model architecture and the small number of adapter parameters, our models are competitive and top the methods freezing both vision and language models except on VATEX. The results demonstrate that our VideoPrism encoder is able to generalize well to video-to-language generation tasks.

3.4. CV for science tasks

While existing video analysis benchmarks commonly focus on human-centric data, we evaluate VideoPrism on a broad set of videos from scientific datasets to assess its generalizability and potential to be used in scientific applications. These datasets include fields such as ethology (Eyjolfsdottir et al., 2014), behavioral neuroscience (Sun et al., 2021a; Burgos-Artizzu et al., 2012), cognitive science (Ma et al., 2023), and ecology (Kholiavchenko et al., 2024). To the best of our knowledge, this work is the first to study the use of ViFMs on scientific datasets, highlighting their ability to match or surpass the performance of specialized models. We encourage the creation of more open-sourced datasets from real-world scientific experiments to unlock the potential of ViFMs to benefit various fields of science.

Datasets. We focus on large-scale video datasets annotated with domain expertise, captured in scientific experiments. These datasets consist of flies (Fly vs. Fly (Eyjolfsdottir et al., 2014)), mice (CalMS21 (Sun et al., 2021a), CRIM13 (Burgos-Artizzu et al., 2012)), chimpanzees (ChimpACT (Ma et al., 2023)), and Kenyan animals (KABR (Kholiavchenko et al., 2024)). All the datasets are annotated for video classification of behavior, except for the ChimpACT dataset for spatiotemporal action localization. We evaluate CRIM13 from cameras on the side perpendicular to the cage ("S"), as well as a top, overhead view ("T"). We use standard data splits defined in previous works on these datasets, and all datasets are evaluated using the mAP metric, except KABR which uses macro-accuracy. Further implementation details are in Appendix H.

Main results. General ViFMs, using a shared frozen encoder across all evaluations, achieve performance comparable to (or exceeding) domain-specific models specialized for individual tasks (Table 7). In particular, VideoPrism generally performs the best and surpasses domain expert models with the base-scale model. Scaling to large-scale models further improves performance across all datasets. These results demonstrate that ViFMs have the potential to significantly accelerate video analysis across diverse fields.

3.5. Ablation study

The main driving force behind VideoPrism includes the strategy and effort for collecting the pretraining data and the pretraining approach that improves upon masked autoencoding by the two-stage pretraining framework, the global distillation, and token shuffling. We run ablation studies to evalu-

Table 7. **Comparison to state-of-the-art methods and domain experts on CV for Science benchmarks**. We report mean average precision (mAP) for all datasets, except for KABR which uses macro-accuracy.

Methods	Fly vs. Fly	CalMS21	CRIM13 (S/T)	KABR	ChimpACT
Domain experts	88.6	88.9	-	61.9	24.4
Base-scale models					
CoCa-B (Yu et al., 2022)	80.1	89.2	58.2 / 58.4	**62.0**	12.6
InternVideo-B (Wang et al., 2022c)	78.9	89.0	63.2 / 63.6	49.9	24.0
UMT-B (Li et al., 2023b)	84.6	88.7	59.3 / 58.5	58.9	25.0
VideoPrism-B	**89.1** (↑4.5)	**91.1** (↑0.9)	**64.5** (↑1.3) / **64.9** (↑1.3)	61.6 (↓0.4)	**28.8** (↑3.8)
Large-scale models					
InternVideo-L (Wang et al., 2022c)	86.6	**91.5**	65.7 / 65.2	51.4	25.7
UMT-L (Li et al., 2023b)	86.4	89.5	60.5 / 61.4	62.7	24.7
VideoPrism-g	**92.0** (↑5.4)	**91.5** (↑0.0)	**65.9** (↑0.2) / **66.8** (↑1.6)	**63.3** (↑0.6)	**31.5** (↑5.8)

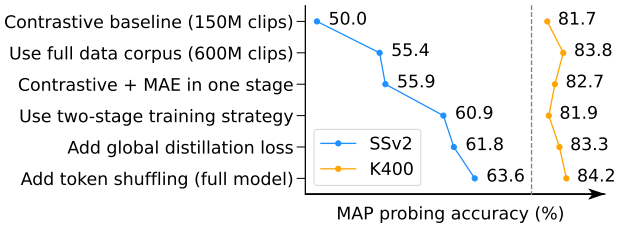


Figure 4. **Ablation study.** From top to bottom: we begin by a video-text contrastive baseline and gradually add our major components to it. Each row is based on a modification of the immediately preceding row. We note that it is difficult to perform well on both K400 and SSv2 using only a single frozen encoder, but our final model with all improvements excels on both datasets.

ate the effectiveness of these components. First, we train a video-text contrastive baseline as presented in Section 2.3.1 over a smaller scale, publicly available corpus (150M video clips in total), including WTS-70M, YT-Temporal-180M, and InternVid. We then add our main components (larger pretraining data, two-stage training, losses, and token shuffling) to the baseline one at a time to see how the model performance evolves along the way. We also experiment with combining contrastive loss with masked autoencoding (Feichtenhofer et al., 2022) in one stage to highlight the effectiveness of our two-stage training pipeline.

Figure 4 exhibits the ablation results, where we observe different performance evolving trajectories on motion-rich SSv2 and appearance-driven K400. Notably, the consistent improvements of VideoPrism on SSv2 suggest the effectiveness of our data curation and model designing efforts for facilitating motion understanding in videos. Although the contrastive baseline has already achieved competitive results on K400, the proposed global distillation and token shuffling further boost the accuracy. We provide more comprehensive ablation studies in Appendix I.

3.6. Limitations

One limitation of our approach is that we leverage a video corpus with noisy text as part of pretraining. This noisy text is potentially incomplete and biased, which could impact model performance. Moreover, long video understanding remains a challenge, since our current focus is on short video clips from which we sample 16 frames as input to VideoPrism. Future work in this direction could leverage our encoder as part of a long video understanding system. Finally, while we advocate for the frozen-backbone evaluation, we acknowledge that there are scenarios that benefit more from end-to-end finetuning and parameter-efficient adaptation. Despite these limitations, the results demonstrate the potential impact of VideoPrism on a range of real-world video understanding tasks.

4. Related work

Foundation models (FMs) (Bommasani et al., 2021) have demonstrated tremendous promise with early work in LLMs (Devlin et al., 2019; Brown et al., 2020). Some ViFMs are built around LLMs (Wang et al., 2022d; Li et al., 2023a; Zhang et al., 2023a; Chen et al., 2023a), analyzing videos by feeding associated text to LLMs, such as ASR transcripts and machine-generated captions. In contrast, VideoPrism takes a video-centric view, and we aim to tackle a broader range of video understanding tasks.

ViFMs. Most recent FMs in CV focus on images (Radford et al., 2021; Yuan et al., 2021; Jia et al., 2021; Yu et al., 2022; Alayrac et al., 2022; Yan et al., 2022; Wang et al., 2022a; Chen et al., 2023c; Xu et al., 2023; Girdhar et al., 2023; Zhang et al., 2023b; Zhu et al., 2024). Their pretraining data contains no or only a small portion of videos, and the model architectures and learning methods are for images by design. While these FMs can accept video frames as input, they fall short on motion and temporal modeling (Yuan et al., 2023). Our work directly addresses this gap by developing a video encoder designed for video-specific applications.

For videos, existing works mainly train FMs using self-supervised learning over the video-only modality (Qian et al., 2021; Feichtenhofer et al., 2021; Recasens et al., 2021; Singh et al., 2021; Wei et al., 2022; Yuan et al., 2022;

Qian et al., 2022; Tong et al., 2022; Wang et al., 2023b) or video-language modeling of videos with noisy text (Zellers et al., 2021; Fu et al., 2021; Li et al., 2023c; Wang et al., 2023a; Cheng et al., 2023; Piergiovanni et al., 2023; Xiong et al., 2023). As Wang et al. (2023f) point out, existing video-language models lack knowledge of actions, and yet self-supervised models from video-only data struggle with semantics. We instead bring the best of the two together. Related to our work, InternVideo (Wang et al., 2022c) glues a self-supervised VideoMAE model (Wang et al., 2023b) and a video-language model together using cross-attention modules. Unlike VideoPrism, however, the two models have no mutual influence during pretraining and they redundantly process the same video from scratch simultaneously.

Large-scale video datasets are pivotal for ViFMs and have been a subject of interest. HowTo100M (Miech et al., 2019), YT-Temporal-1B (Zellers et al., 2022), and HD-VILA-100M (Xue et al., 2022) associate speech transcriptions with videos. WebVid2M (Bain et al., 2021) and WTS70M (Stroud et al., 2020) pair alt-text and other metadata with videos. VideoCC3M (Nagrani et al., 2022) retrieves videos that appear similar to images and transfer the image captions to corresponding videos. VAST-27M (Chen et al., 2023b) and InternVid (Wang et al., 2023e) use multimodal and language models to caption videos. Still, these video-text datasets are significantly smaller than their counterparts for images, and many ViFMs adapt pretrained image-text models to the video space (Fang et al., 2021; Luo et al., 2022; Xue et al., 2023; Liu et al., 2023; He et al., 2023; Wu et al., 2024). Our pretraining corpus has text associations from a hybrid mix of ASR transcripts, generated captions, and high-quality manually annotated captions.

Pretraining strategy. Our pretraining integrates vision-language contrastive learning (Radford et al., 2021; Xu et al., 2021; Bain et al., 2022) and masked data modeling (Devlin et al., 2019; He et al., 2022). The former has led to strong late-fusion models like CLIP (Radford et al., 2021), ALIGN (Jia et al., 2021), CoCa (Yu et al., 2022), and the latter is proven effective to learn from single-modality data like language (Devlin et al., 2019; Anil et al., 2023), audio (Borsos et al., 2023), images (He et al., 2022; Wang et al., 2023d; Oquab et al., 2023), and videos (Tong et al., 2022; Wang et al., 2023b). While EVA (Fang et al., 2022; 2023) and UMT (Li et al., 2023b) transfer indirect semantics from CLIP (Radford et al., 2021) to masked modeling, we learn video-native semantics. We also introduce global distillation and token shuffling to the masked video modeling to orchestrate both appearance and motion cues.

5. Conclusion

We present VideoPrism, a foundational video encoder that achieves state-of-the-art performance across a wide range of video understanding tasks. Our design emphasizes both the data and modeling approach: we assemble the largest pretraining dataset of its kind, as well as develop a pretraining strategy that effectively learns appearance and motion information from it. In our comprehensive evaluation, VideoPrism achieves the best results on a majority of benchmarks. Notably, no other baseline models consistently achieve the second best, highlighting our unique generalizability.

Acknowledgements

We sincerely thank David Hendon for their product management efforts, and Alex Siegman, Ramya Ganeshan, and Victor Gomes for their program and resource management efforts. We also thank Hassan Akbari, Sherry Ben, Yoni Ben-Meshulam, Chun-Te Chu, Sam Clearwater, Yin Cui, Ilya Figotin, Anja Hauth, Sergey Ioffe, Xuhui Jia, Yeqing Li, Lu Jiang, Zu Kim, Dan Kondratyuk, Bill Mark, Arsha Nagrani, Caroline Pantofaru, Sushant Prakash, Cordelia Schmid, Bryan Seybold, Mojtaba Seyedhosseini, Amanda Sadler, Rif A. Saurous, Rachel Stigler, Paul Voigtlaender, Pingmei Xu, Chaochao Yan, Xuan Yang, and Yukun Zhu for the discussions, support, and feedback that greatly contributed to this paper. Lastly, we thank Jay Yagnik, Rahul Sukthankar, and Tomas Izo for their enthusiastic support for this project.

Impact statement

Advancements in video understanding have the potential to accelerate progress across various fields, including scientific research, education, robotics, healthcare, and content recommendation. These technologies could empower new scientific discoveries, enhance learning experiences, improve security and safety, and enable more responsive interactive systems. However, it is crucial to address potential biases and misuses before one deploys related models to the real world. This includes mitigating algorithmic biases, safeguarding privacy, and respecting rules and policies of responsible research. To ensure that the benefits of this technology are harnessed responsibly, we encourage continued open discussions in the community around the development of these new technologies.

References

Akbari, H., Yuan, L., Qian, R., Chuang, W.-H., Chang, S.-F., Cui, Y., and Gong, B. VATT: Transformers for multimodal self-supervised learning from raw video, audio and text. In *NeurIPS*, 2021.

Akbari, H., Kondratyuk, D., Cui, Y., Hornung, R., Wang, H., and Adam, H. Alternating gradient descent and mixture-of-experts for integrated multimodal perception.

In *NeurIPS*, 2023.

Alayrac, J.-B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., et al. Flamingo: A visual language model for few-shot learning. In *NeurIPS*, 2022.

Anil, R., Dai, A. M., Firat, O., Johnson, M., Lepikhin, D., Passos, A., Shakeri, S., Taropa, E., Bailey, P., Chen, Z., et al. PaLM 2 technical report. *arXiv preprint arXiv:2305.10403*, 2023.

Arnab, A., Dehghani, M., Heigold, G., Sun, C., Lučić, M., and Schmid, C. ViViT: A video vision transformer. In *ICCV*, 2021.

Bagad, P., Tapaswi, M., and Snoek, C. G. Test of time: Instilling video-language models with a sense of time. In *CVPR*, 2023.

Bain, M., Nagrani, A., Varol, G., and Zisserman, A. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *ICCV*, 2021.

Bain, M., Nagrani, A., Varol, G., and Zisserman, A. A CLIP-Hitchhiker's guide to long video retrieval. *arXiv preprint arXiv:2205.08508*, 2022.

Bansal, H., Bitton, Y., Szpektor, I., Chang, K.-W., and Grover, A. VideoCon: Robust video-language alignment via contrast captions. *arXiv preprint arXiv:2311.10111*, 2023.

Bao, H., Dong, L., Piao, S., and Wei, F. BEiT: BERT pre-training of image transformers. In *ICLR*, 2022.

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

Borsos, Z., Marinier, R., Vincent, D., Kharitonov, E., Pietquin, O., Sharifi, M., Roblek, D., Teboul, O., Grangier, D., Tagliasacchi, M., et al. AudioLM: A language modeling approach to audio generation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:2523–2533, 2023.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. In *NeurIPS*, 2020.

Buch, S., Eyzaguirre, C., Gaidon, A., Wu, J., Fei-Fei, L., and Niebles, J. C. Revisiting the "video" in video-language understanding. In *CVPR*, 2022.

Burgos-Artizzu, X. P., Dollár, P., Lin, D., Anderson, D. J., and Perona, P. Social behavior recognition in continuous video. In *CVPR*, 2012.

Caba Heilbron, F., Escorcia, V., Ghanem, B., and Carlos Niebles, J. ActivityNet: A large-scale video benchmark for human activity understanding. In *CVPR*, 2015.

Carreira, J., Noland, E., Banki-Horvath, A., Hillier, C., and Zisserman, A. A short note about Kinetics-600. *arXiv preprint arXiv:1808.01340*, 2018.

Chen, G., Zheng, Y.-D., Wang, J., Xu, J., Huang, Y., Pan, J., Wang, Y., Wang, Y., Qiao, Y., Lu, T., et al. VideoLLM: Modeling video sequence with large language models. *arXiv preprint arXiv:2305.13292*, 2023a.

Chen, S. and Huang, D. Elaborative rehearsal for zero-shot action recognition. In *ICCV*, 2021.

Chen, S., Li, H., Wang, Q., Zhao, Z., Sun, M., Zhu, X., and Liu, J. VAST: A vision-audio-subtitle-text omni-modality foundation model and dataset. In *NeurIPS*, 2023b.

Chen, X., Wang, X., Changpinyo, S., Piergiovanni, A., Padlewski, P., Salz, D., Goodman, S., Grycner, A., Mustafa, B., Beyer, L., et al. PaLI: A jointly-scaled multilingual language-image model. In *ICLR*, 2023c.

Cheng, F., Wang, X., Lei, J., Crandall, D., Bansal, M., and Bertasius, G. VindLU: A recipe for effective video-and-language pretraining. In *CVPR*, 2023.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 2019.

Dima, D., Doughty, H., Farinella, G. M., Antonino, F., Evangelos, K., Ma, J., Davide, M., Munro, J., Toby, P., Price, W., et al. Rescaling egocentric vision: Collection, pipeline and challenges for EPIC-KITCHENS-100. *IJCV*, 130(1):33–55, 2022.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

Eyjolfsdottir, E., Branson, S., Burgos-Artizzu, X. P., Hoopfer, E. D., Schor, J., Anderson, D. J., and Perona, P. Detecting social actions of fruit flies. In *ECCV*, 2014.

Fang, H., Xiong, P., Xu, L., and Chen, Y. CLIP2Video: Mastering video-text retrieval via image CLIP. *arXiv preprint arXiv:2106.11097*, 2021.

Fang, Y., Wang, W., Xie, B., Sun, Q.-S., Wu, L. Y., Wang, X., Huang, T., Wang, X., and Cao, Y. EVA: Exploring the limits of masked visual representation learning at scale. In *CVPR*, 2022.

Fang, Y., Sun, Q., Wang, X., Huang, T., Wang, X., and Cao, Y. EVA-02: A visual representation for neon genesis. *arXiv preprint arXiv:2303.11331*, 2023.

Feichtenhofer, C., Fan, H., Malik, J., and He, K. SlowFast networks for video recognition. In *ICCV*, 2019.

Feichtenhofer, C., Fan, H., Xiong, B., Girshick, R., and He, K. A large-scale study on unsupervised spatiotemporal representation learning. In *CVPR*, 2021.

Feichtenhofer, C., Fan, H., Li, Y., and He, K. Masked autoencoders as spatiotemporal learners. In *NeurIPS*, 2022.

Fellbaum, C. WordNet and wordnets. In Barber, A. (ed.), *Encyclopedia of Language and Linguistics*, pp. 2–665. Elsevier, 2005.

Fu, T.-J., Li, L., Gan, Z., Lin, K., Wang, W. Y., Wang, L., and Liu, Z. VIOLET: End-to-end video-language transformers with masked visual-token modeling. *arXiv preprint arXiv:2111.12681*, 2021.

Gao, J., Sun, C., Yang, Z., and Nevatia, R. TALL: Temporal activity localization via language query. In *ICCV*, 2017.

Girdhar, R., El-Nouby, A., Liu, Z., Singh, M., Alwala, K. V., Joulin, A., and Misra, I. ImageBind: One embedding space to bind them all. In *CVPR*, 2023.

Goyal, R., Ebrahimi Kahou, S., Michalski, V., Materzynska, J., Westphal, S., Kim, H., Haenel, V., Fruend, I., Yianilos, P., Mueller-Freitag, M., et al. The "something something" video database for learning and evaluating visual common sense. In *ICCV*, 2017a.

Goyal, Y., Khot, T., Summers-Stay, D., Batra, D., and Parikh, D. Making the V in VQA matter: Elevating the role of image understanding in visual question answering. In *CVPR*, 2017b.

Grauman, K., Westbury, A., Byrne, E., Chavis, Z., Furnari, A., Girdhar, R., Hamburger, J., Jiang, H., Liu, M., Liu, X., et al. Ego4D: Around the world in 3,000 hours of egocentric video. In *CVPR*, 2022.

Gu, C., Sun, C., Ross, D. A., Vondrick, C., Pantofaru, C., Li, Y., Vijayanarasimhan, S., Toderici, G., Ricco, S., Sukthankar, R., et al. AVA: A video dataset of spatio-temporally localized atomic visual actions. In *CVPR*, 2018.

Gutmann, M. and Hyvärinen, A. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *AISTATS*, 2010.

He, K., Chen, X., Xie, S., Li, Y., Dollár, P., and Girshick, R. Masked autoencoders are scalable vision learners. In *CVPR*, 2022.

He, X., Chen, S., Ma, F., Huang, Z., Jin, X., Liu, Z., Fu, D., Yang, Y., Liu, J., and Feng, J. VLAB: Enhancing video language pre-training by feature adapting and blending. *arXiv preprint arXiv:2305.13167*, 2023.

Hendricks, L. A. and Nematzadeh, A. Probing image-language transformers for verb understanding. In *ACL*, 2021.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. LoRA: Low-rank adaptation of large language models. In *ICLR*, 2022.

Huang, G., Pang, B., Zhu, Z., Rivera, C., and Soricut, R. Multimodal pretraining for dense video captioning. *arXiv preprint arXiv:2011.11760*, 2020.

Huang, T.-H., Ferraro, F., Mostafazadeh, N., Misra, I., Agrawal, A., Devlin, J., Girshick, R., He, X., Kohli, P., Batra, D., et al. Visual storytelling. In *NAACL-HLT*, 2016.

Jain, P., Kar, P., et al. Non-convex optimization for machine learning. *Foundations and Trends® in Machine Learning*, 10(3-4):142–363, 2017.

Jia, C., Yang, Y., Xia, Y., Chen, Y.-T., Parekh, Z., Pham, H., Le, Q., Sung, Y.-H., Li, Z., and Duerig, T. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021.

Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., et al. The Kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

Kholiavchenko, M., Kline, J., Ramirez, M., Stevens, S., Sheets, A., Babu, R., Banerji, N., Campolongo, E., Thompson, M., Van Tiel, N., et al. KABR: In-situ dataset for kenyan animal behavior recognition from drone videos. In *WACV*, 2024.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Krishna, R., Hata, K., Ren, F., Fei-Fei, L., and Carlos Niebles, J. Dense-captioning events in videos. In *ICCV*, 2017.

Lee, J., Lee, Y., Kim, J., Kosiorek, A., Choi, S., and Teh, Y. W. Set Transformer: A framework for attention-based permutation-invariant neural networks. In *ICML*, 2019.

Lei, J., Berg, T. L., and Bansal, M. Revealing single frame bias for video-and-language learning. In *ACL*, 2023.

Li, A., Thotakuri, M., Ross, D. A., Carreira, J., Vostrikov, A., and Zisserman, A. The AVA-Kinetics localized human actions video dataset. *arXiv preprint arXiv:2005.00214*, 2020.

Li, J., Li, D., Xiong, C., and Hoi, S. BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022.

Li, J., Li, D., Savarese, S., and Hoi, S. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023a.

Li, K., Wang, Y., Li, Y., Wang, Y., He, Y., Wang, L., and Qiao, Y. Unmasked teacher: Towards training-efficient video foundation models. In *ICCV*, 2023b.

Li, L., Gan, Z., Lin, K., Lin, C.-C., Liu, Z., Liu, C., and Wang, L. LAVENDER: Unifying video-language understanding as masked language modeling. In *CVPR*, 2023c.

Li, W., Zhu, L., Wen, L., and Yang, Y. DeCap: Decoding CLIP latents for zero-shot captioning via text-only training. In *ICLR*, 2023d.

Li, Y., Li, Y., and Vasconcelos, N. RESOUND: Towards action recognition without representation bias. In *ECCV*, 2018.

Li, Y., Fan, H., Hu, R., Feichtenhofer, C., and He, K. Scaling language-image pre-training via masking. In *CVPR*, 2023e.

Li, Y., Wang, C., and Jia, J. LLaMA-VID: An image is worth 2 tokens in large language models. *arXiv preprint arXiv:2311.17043*, 2023f.

Lin, B., Zhu, B., Ye, Y., Ning, M., Jin, P., and Yuan, L. Video-LLaVA: Learning united visual representation by alignment before projection. *arXiv preprint arXiv:2311.10122*, 2023a.

Lin, K. Q., Wang, J., Soldan, M., Wray, M., Yan, R., XU, E. Z., Gao, D., Tu, R.-C., Zhao, W., Kong, W., et al. Egocentric video-language pretraining. In *NeurIPS*, 2022.

Lin, W., Karlinsky, L., Shvetsova, N., Possegger, H., Kozinski, M., Panda, R., Feris, R., Kuehne, H., and Bischof, H. Match, expand and improve: Unsupervised finetuning for zero-shot action recognition with language knowledge. In *ICCV*, 2023b.

Liu, R., Huang, J., Li, G., Feng, J., Wu, X., and Li, T. H. Revisiting temporal modeling for clip-based image-to-video knowledge transferring. In *CVPR*, 2023.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. In *ICLR*, 2019.

Luo, H., Ji, L., Zhong, M., Chen, Y., Lei, W., Duan, N., and Li, T. CLIP4Clip: An empirical study of clip for end to end video clip retrieval and captioning. *Neurocomputing*, 508:293–304, 2022.

Ma, X., Kaufhold, S. P., Su, J., Zhu, W., Terwilliger, J., Meza, A., Zhu, Y., Rossano, F., and Wang, Y. ChimpACT: A longitudinal dataset for understanding chimpanzee behaviors. *arXiv preprint arXiv:2310.16447*, 2023.

Maaz, M., Rasheed, H., Khan, S., and Khan, F. S. Video-ChatGPT: Towards detailed video understanding via large vision and language models. *arXiv preprint arXiv:2306.05424*, 2023.

McCloskey, M. and Cohen, N. J. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pp. 109–165. Elsevier, 1989.

Miech, A., Zhukov, D., Alayrac, J.-B., Tapaswi, M., Laptev, I., and Sivic, J. HowTo100M: Learning a text-video embedding by watching hundred million narrated video clips. In *ICCV*, 2019.

Momeni, L., Caron, M., Nagrani, A., Zisserman, A., and Schmid, C. Verbs in action: Improving verb understanding in video-language models. In *ICCV*, 2023.

Monfort, M., Andonian, A., Zhou, B., Ramakrishnan, K., Bargal, S. A., Yan, T., Brown, L., Fan, Q., Gutfreund, D., Vondrick, C., et al. Moments in Time dataset: one million videos for event understanding. *IEEE TPAMI*, 42 (2):502–508, 2019.

Monfort, M., Jin, S., Liu, A., Harwath, D., Feris, R., Glass, J., and Oliva, A. Spoken Moments: Learning joint audio-visual representations from video descriptions. In *CVPR*, 2021.

Nagrani, A., Seo, P. H., Seybold, B., Hauth, A., Manen, S., Sun, C., and Schmid, C. Learning audio-video modalities from image captions. In *ECCV*, 2022.

Ni, B., Peng, H., Chen, M., Zhang, S., Meng, G., Fu, J., Xiang, S., and Ling, H. Expanding language-image pretrained models for general video recognition. In *ECCV*, 2022.

Noroozi, M. and Favaro, P. Unsupervised learning of visual representations by solving Jigsaw puzzles. In *ECCV*, 2016.

Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al. DINOv2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Peng, Z., Dong, L., Bao, H., Ye, Q., and Wei, F. BEiT v2: Masked image modeling with vector-quantized visual tokenizers. *arXiv preprint arXiv:2208.06366*, 2022.

Piergiovanni, A., Nobel, I., Kim, D., Ryoo, M. S., Gomes, V., and Angelova, A. Mirasol3B: A multimodal autoregressive model for time-aligned and contextual modalities. *arXiv preprint arXiv:2311.05698*, 2023.

Pitcher-Cooper, C., Seth, M., Kao, B., Coughlan, J. M., and Yoon, I. You Described, We Archived: A rich audio description dataset. *Journal on Technology and Persons with Disabilities*, 2023.

Qian, R., Meng, T., Gong, B., Yang, M.-H., Wang, H., Belongie, S., and Cui, Y. Spatiotemporal contrastive video representation learning. In *CVPR*, 2021.

Qian, R., Li, Y., Yuan, L., Gong, B., Liu, T., Brown, M., Yang, M.-H., Adam, H., and Cui, Y. On temporal granularity in self-supervised video representation learning. In *BMVC*, 2022.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

Recasens, A., Luc, P., Alayrac, J.-B., Wang, L., Strub, F., Tallec, C., Malinowski, M., Pătrăucean, V., Altché, F., Valko, M., et al. Broaden your views for self-supervised video learning. In *ICCV*, 2021.

Regneri, M., Rohrbach, M., Wetzel, D., Thater, S., Schiele, B., and Pinkal, M. Grounding action descriptions in videos. *Transactions of the Association for Computational Linguistics*, 1:25–36, 2013.

Ren, S., He, K., Girshick, R., and Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015.

Sevilla-Lara, L., Zha, S., Yan, Z., Goswami, V., Feiszli, M., and Torresani, L. Only time can tell: Discovering temporal data for temporal modeling. In *WACV*, 2021.

Shazeer, N. and Stern, M. Adafactor: Adaptive learning rates with sublinear memory cost. In *ICML*, 2018.

Sigurdsson, G. A., Varol, G., Wang, X., Farhadi, A., Laptev, I., and Gupta, A. Hollywood in Homes: Crowdsourcing data collection for activity understanding. In *ECCV*, 2016.

Singh, A., Chakraborty, O., Varshney, A., Panda, R., Feris, R., Saenko, K., and Das, A. Semi-supervised action recognition with temporal contrastive learning. In *CVPR*, 2021.

Singh, A., Hu, R., Goswami, V., Couairon, G., Galuba, W., Rohrbach, M., and Kiela, D. FLAVA: A foundational language and vision alignment model. In *CVPR*, 2022.

Stroud, J. C., Lu, Z., Sun, C., Deng, J., Sukthankar, R., Schmid, C., and Ross, D. A. Learning video representations from textual web supervision. *arXiv preprint arXiv:2007.14937*, 2020.

Sun, J. J., Karigo, T., Chakraborty, D., Mohanty, S. P., Wild, B., Sun, Q., Chen, C., Anderson, D. J., Perona, P., Yue, Y., et al. The multi-agent behavior dataset: Mouse dyadic social interactions. In *NeurIPS D&B*, 2021a.

Sun, J. J., Kennedy, A., Zhan, E., Anderson, D. J., Yue, Y., and Perona, P. Task programming: Learning data efficient behavior representations. In *CVPR*, 2021b.

Tan, J., Wang, C., Li, B., Li, Q., Ouyang, W., Yin, C., and Yan, J. Equalization loss for long-tailed object recognition. In *CVPR*, 2020.

Tang, Y., Ding, D., Rao, Y., Zheng, Y., Zhang, D., Zhao, L., Lu, J., and Zhou, J. COIN: A large-scale dataset for comprehensive instructional video analysis. In *CVPR*, 2019.

Tang, Y., Bi, J., Xu, S., Song, L., Liang, S., Wang, T., Zhang, D., An, J., Lin, J., Zhu, R., et al. Video understanding with large language models: A survey. *arXiv preprint arXiv:2312.17432*, 2023.

Tong, Z., Song, Y., Wang, J., and Wang, L. VideoMAE: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In *NeurIPS*, 2022.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *NeurIPS*, 2017.

Voigtlaender, P., Changpinyo, S., Pont-Tuset, J., Soricut, R., and Ferrari, V. Connecting vision and language with video localized narratives. In *CVPR*, 2023.

Wang, J., Chen, D., Wu, Z., Luo, C., Zhou, L., Zhao, Y., Xie, Y., Liu, C., Jiang, Y.-G., and Yuan, L. OmniVL: One foundation model for image-language and video-language tasks. In *NeurIPS*, 2022a.

Wang, J., Ge, Y., Yan, R., Ge, Y., Lin, K. Q., Tsutsui, S., Lin, X., Cai, G., Wu, J., Shan, Y., et al. All in one: Exploring unified video-language pre-training. In *CVPR*, 2023a.

Wang, L., Huang, B., Zhao, Z., Tong, Z., He, Y., Wang, Y., Wang, Y., and Qiao, Y. VideoMAE v2: Scaling video masked autoencoders with dual masking. In *CVPR*, 2023b.

Wang, R., Chen, D., Wu, Z., Chen, Y., Dai, X., Liu, M., Jiang, Y.-G., Zhou, L., and Yuan, L. BEVT: BERT pre-training of video transformers. In *CVPR*, 2022b.

Wang, R., Chen, D., Wu, Z., Chen, Y., Dai, X., Liu, M., Yuan, L., and Jiang, Y.-G. Masked video distillation: Rethinking masked feature modeling for self-supervised video representation learning. In *CVPR*, 2023c.

Wang, W., Bao, H., Dong, L., Bjorck, J., Peng, Z., Liu, Q., Aggarwal, K., Mohammed, O. K., Singhal, S., Som, S., et al. Image as a foreign language: BEiT pretraining for vision and vision-language tasks. In *CVPR*, 2023d.

Wang, X., Wu, J., Chen, J., Li, L., Wang, Y.-F., and Wang, W. Y. VATEX: A large-scale, high-quality multilingual dataset for video-and-language research. In *ICCV*, 2019.

Wang, Y., Li, K., Li, Y., He, Y., Huang, B., Zhao, Z., Zhang, H., Xu, J., Liu, Y., Wang, Z., et al. InternVideo: General video foundation models via generative and discriminative learning. *arXiv preprint arXiv:2212.03191*, 2022c.

Wang, Y., He, Y., Li, Y., Li, K., Yu, J., Ma, X., Chen, X., Wang, Y., Luo, P., Liu, Z., et al. InternVid: A large-scale video-text dataset for multimodal understanding and generation. *arXiv preprint arXiv:2307.06942*, 2023e.

Wang, Z., Li, M., Xu, R., Zhou, L., Lei, J., Lin, X., Wang, S., Yang, Z., Zhu, C., Hoiem, D., et al. Language models with image descriptors are strong few-shot video-language learners. In *NeurIPS*, 2022d.

Wang, Z., Blume, A., Li, S., Liu, G., Cho, J., Tang, Z., Bansal, M., and Ji, H. Paxion: Patching action knowledge in video-language foundation models. In *NeurIPS*, 2023f.

Wei, C., Fan, H., Xie, S., Wu, C.-Y., Yuille, A., and Feichtenhofer, C. Masked feature prediction for self-supervised visual pre-training. In *CVPR*, 2022.

Wu, C., Huang, L., Zhang, Q., Li, B., Ji, L., Yang, F., Sapiro, G., and Duan, N. Godiva: Generating open-domain videos from natural descriptions. *arXiv preprint arXiv:2104.14806*, 2021.

Wu, W., Sun, Z., and Ouyang, W. Revisiting classifier: Transferring vision-language models for video recognition. In *AAAI*, 2023.

Wu, Z. and Palmer, M. Verb semantics and lexical selection. In *ACL*, 1994.

Wu, Z., Weng, Z., Peng, W., Yang, X., Li, A., Davis, L. S., and Jiang, Y.-G. Building an open-vocabulary video CLIP model with better architectures, optimization and data. *IEEE TPAMI*, 2024.

Xiao, J., Shang, X., Yao, A., and Chua, T.-S. NExT-QA: Next phase of question-answering to explaining temporal actions. In *CVPR*, 2021.

Xiong, Y., Zhao, L., Gong, B., Yang, M.-H., Schroff, F., Liu, T., Hsieh, C.-J., and Yuan, L. Spatiotemporally discriminative video-language pre-training with text grounding. *arXiv preprint arXiv:2303.16341*, 2023.

Xu, D., Zhao, Z., Xiao, J., Wu, F., Zhang, H., He, X., and Zhuang, Y. Video question answering via gradually refined attention over appearance and motion. In *ACM MM*, 2017.

Xu, H., Ghosh, G., Huang, P.-Y., Okhonko, D., Aghajanyan, A., Metze, F., Zettlemoyer, L., and Feichtenhofer, C. VideoCLIP: Contrastive pre-training for zero-shot video-text understanding. In *EMNLP*, 2021.

Xu, H., Ye, Q., Yan, M., Shi, Y., Ye, J., Xu, Y., Li, C., Bi, B., Qian, Q., Wang, W., et al. mPLUG-2: A modularized multi-modal foundation model across text, image and video. In *ICML*, 2023.

Xu, J., Mei, T., Yao, T., and Rui, Y. MSR-VTT: A large video description dataset for bridging video and language. In *CVPR*, 2016.

Xu, M., Zhao, C., Rojas, D. S., Thabet, A., and Ghanem, B. G-TAD: Sub-graph localization for temporal action detection. In *CVPR*, 2020.

Xue, H., Hang, T., Zeng, Y., Sun, Y., Liu, B., Yang, H., Fu, J., and Guo, B. Advancing high-resolution video-language representation with large-scale video transcriptions. In *CVPR*, 2022.

Xue, H., Sun, Y., Liu, B., Fu, J., Song, R., Li, H., and Luo, J. CLIP-ViP: Adapting pre-trained image-text model to video-language representation alignment. In *ICLR*, 2023.

Yan, S., Zhu, T., Wang, Z., Cao, Y., Zhang, M., Ghosh, S., Wu, Y., and Yu, J. VideoCoCa: Video-text modeling with zero-shot transfer from contrastive captioners. *arXiv preprint arXiv:2212.04979*, 2022.

Yang, A., Miech, A., Sivic, J., Laptev, I., and Schmid, C. Zero-shot video question answering via frozen bidirectional language models. In *NeurIPS*, 2022.

Yarom, M., Bitton, Y., Changpinyo, S., Aharoni, R., Herzig, J., Lang, O., Ofek, E., and Szpektor, I. What you see is what you read? improving text-image alignment evaluation. In *NeurIPS*, 2023.

Ye, Q., Xu, G., Yan, M., Xu, H., Qian, Q., Zhang, J., and Huang, F. HiTeA: Hierarchical temporal-aware video-language pre-training. In *ICCV*, 2023.

Yu, J., Wang, Z., Vasudevan, V., Yeung, L., Seyedhosseini, M., and Wu, Y. CoCa: Contrastive captioners are image-text foundation models. *TMLR*, 2022. ISSN 2835-8856. URL https://openreview.net/forum?id=Ee277P3AYC.

Yuan, L., Chen, D., Chen, Y.-L., Codella, N., Dai, X., Gao, J., Hu, H., Huang, X., Li, B., Li, C., et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021.

Yuan, L., Qian, R., Cui, Y., Gong, B., Schroff, F., Yang, M.-H., Adam, H., and Liu, T. Contextualized spatio-temporal contrastive learning with self-supervision. In *CVPR*, 2022.

Yuan, L., Gundavarapu, N. B., Zhao, L., Zhou, H., Cui, Y., Jiang, L., Yang, X., Jia, M., Weyand, T., Friedman, L., et al. VideoGLUE: Video general understanding evaluation of foundation models. *arXiv preprint arXiv:2307.03166*, 2023.

Zellers, R., Lu, X., Hessel, J., Yu, Y., Park, J. S., Cao, J., Farhadi, A., and Choi, Y. MERLOT: Multimodal neural script knowledge models. In *NeurIPS*, 2021.

Zellers, R., Lu, J., Lu, X., Yu, Y., Zhao, Y., Salehi, M., Kusupati, A., Hessel, J., Farhadi, A., and Choi, Y. MERLOT Reserve: Neural script knowledge through vision and language and sound. In *CVPR*, 2022.

Zeng, A., Attarian, M., Ichter, B., Choromanski, K., Wong, A., Welker, S., Tombari, F., Purohit, A., Ryoo, M., Sindhwani, V., et al. Socratic models: Composing zero-shot multimodal reasoning with language. *arXiv preprint arXiv:2204.00598*, 2022.

Zhai, X., Kolesnikov, A., Houlsby, N., and Beyer, L. Scaling vision transformers. In *CVPR*, 2022a.

Zhai, X., Wang, X., Mustafa, B., Steiner, A., Keysers, D., Kolesnikov, A., and Beyer, L. LiT: Zero-shot transfer with locked-image text tuning. In *CVPR*, 2022b.

Zhang, H., Li, X., and Bing, L. Video-LLaMA: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint arXiv:2306.02858*, 2023a.

Zhang, X., Chen, J., Yuan, J., Chen, Q., Wang, J., Wang, X., Han, S., Chen, X., Pi, J., Yao, K., Han, J., Ding, E., and Wang, J. CAE v2: Context autoencoder with CLIP latent alignment. *TMLR*, 2023b. ISSN 2835-8856. URL https://openreview.net/forum?id=f36LaK7M0F.

Zhao, Y., Zhao, L., Zhou, X., Wu, J., Chu, C.-T., Miao, H., Schroff, F., Adam, H., Liu, T., Gong, B., et al. Distilling vision-language models on millions of videos. *arXiv preprint arXiv:2401.06129*, 2024.

Zhou, J., Wei, C., Wang, H., Shen, W., Xie, C., Yuille, A., and Kong, T. iBOT: Image BERT pre-training with online tokenizer. In *ICLR*, 2022.

Zhou, L., Xu, C., and Corso, J. Towards automatic learning of procedures from web instructional videos. In *AAAI*, 2018.

Zhu, B., Lin, B., Ning, M., Yan, Y., Cui, J., Wang, H., Pang, Y., Jiang, W., Zhang, J., Li, Z., et al. LanguageBind: Extending video-language pretraining to N-modality by language-based semantic alignment. In *ICLR*, 2024.

A. Pretraining data

A.1. Data curation

Section 2.1 and Table 1 have described the pretraining corpus of videos, and the following provides more details about the three in-house datasets.

Anonymous-Corpus #1 consists of about 36M commercially licensed stock video-caption pairs, where the videos and text are manually uploaded by professional contributors. Hence, the quality of the videos and captions is high in this corpus compared with the rest. Note that we do not do any filtering on this set.

Anonymous-Corpus #2 contains 170M (video, ASR transcript) pairs from 44.6M YouTube videos. Its construction process is similar to HowTo100M (Miech et al., 2019), but the whole corpus is larger and more diverse. Furthermore, view counts and video lengths are filtered using simple metadata. ASR sentence boundaries define the clip boundaries. The clip-text pairs are filtered based on a groundedness score similar to CLIP's similarity score (Wu et al., 2021).

Anonymous-Corpus #3 includes 71.5M (clip, machine-generated caption) pairs from 36.7M YouTube videos. The clips are captioned using vision-language models (Chen et al., 2023c) and further summarized using an LLM (Anil et al., 2023). The corpus is similar to InternVid (Wang et al., 2023e) in terms of construction but a magnitude larger in size and diversity. The initial video selection of this dataset process is similar to Anonymous-Corpus #2, but additional filters are applied to exclude videos composed of primarily talking heads using a face detection model. Also, static clips are eliminated by ensuring semantic feature embeddings from the frames are not static. Hence, the video content is more diverse than Anonymous-Corpus #2.

A.2. Corpus analysis

We randomly sample 100K videos from our video-text pretraining data and show the breakdown analysis in Figure 5. We notice that most of our clips are between 5 to 10 seconds in length and contain 10 to 20 words in the parallel text. In addition, a considerable proportion of clips has duration longer than 10 seconds or captions longer than 20 words. We further show the the CLIP similarity score (Wu et al., 2021) of our corpus in Figure 5c. The large variations of the CLIP similarity scores demonstrate the diverse caption quality of our training data, which we believe is a byproduct of the various ways used to harvest the text.

Furthermore, we provide an in-depth analyses on each Anonymous dataset and contrast them with the other datasets we used in Table 8. We find that the datasets with captions generated by VLMs and LLMs (*e.g.*, InternVid and Anonymous-Corpus #3) perform the best. In addition, retrieval performance is correlated with dataset size, groundingness (CLIP score), "dynamic degree" (optical flow), and the presence of humans. Finally, we compute the top-50 object categories represented by our dataset obtained by running an open-source Tensorflow object detection API on the center-frame of 100K clips from our pretraining data. We find *person* to be the top category, followed by *car*, *chair*, *TV*, *bottle*, *book*, *potted plant*, and *bowl*. We hope this gives more insights into the composition of our datasets.

B. Model architecture

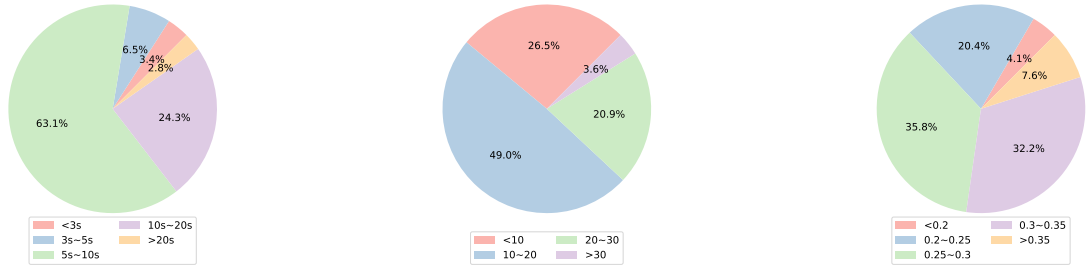
Table 9 shows the VideoPrism model architecture. As mentioned in Section 2.2, the architecture follows the factorized design of ViViT (Arnab et al., 2021). It consists of two separate Transformer modules: a spatial module and a temporal module. After an input video is partitioned into several non-overlapping patches (*i.e.*, tokens), the spatial module first models interactions between tokens from the same temporal index. Then the output sequence of token embeddings are forwarded through the temporal module to model interactions between tokens from different temporal indices. The temporal module shares the same setup of the spatial counterpart, except that its number of layers is fixed to four because no performance improvements are observed with more layers added to our largest VideoPrism model. The positional embeddings of our models are learnable (Devlin et al., 2019) and decoupled in spatial and temporal dimensions. They are utilized to encode the position information of the input tokens in space and time, respectively. When we add image-text data to the first-stage pretraining, the images are treated as one-frame videos, and we crop the temporal positional embeddings when handling the image input. Following CoCa (Yu et al., 2022), we pretrain the model with spatial resolution of 288×288 and patch size 18×18. We uniformly sample 8 frames from each video for pretraining and 16 frames for evaluation by interpolating the temporal positional embedding of our video encoder.

C. Implementation details

In this section, we describe the implementation details and training setups of VideoPrism. We summarize the pretraining configurations in Table 10.

C.1. Stage 1

Model design. The text encoder of the first-stage model is a standard Transformer (Vaswani et al., 2017). Together with the spatial module in our encoder, it is initialized from the unimodal text decoder of CoCa (Yu et al., 2022). We attach a MAP layer (Lee et al., 2019; Yu et al., 2022) to the end of the video encoder to extract the global embedding from the encoder output. For the text encoder, we append



(a) Duration (seconds) (b) Caption length (# of words) (c) CLIP similarity scores

Figure 5. **Analysis on the video-text pretraining corpus.**

Table 8. **More details about different characteristics of each dataset.** The noun/verb ratio and the presence of humans in the video are inferred from the captions using WordNet synsets (Fellbaum, 2005) and the NTLK library. Optical flow is computed at 8 FPS and 256×256 resolution, and the average optical flow across videos is presented. The last column, in addition to the data analyses, is the average of zero-shot R@1 text-video retrieval results on MSRVTT (full-split) and VATEX.

Datasets	Caption source	# of clips	Noun / Verb ratio	% of videos with humans	Optical flow magnitude (px)	Grounding score	Average ZS R@1
InternVid (Wang et al., 2023e)	Generated	7.0M	3.4	79.9	2.47	0.32	45.7
YT-Temporal-180M (Zellers et al., 2021)	ASR	87.8M	1.6	81.8	1.73	0.24	35.8
VideoCC (Nagrani et al., 2022)	Retrievals	133.5M	8.0	69.0	3.21	0.27	32.6
WTS-70M (Stroud et al., 2020)	Metadata	55.1M	1.3	18.5	3.67	0.26	19.4
Anonymous-Corpus #1	Manual	36.1M	5.9	76.9	1.55	0.30	37.0
Anonymous-Corpus #2	ASR	170.3M	2.8	62.2	2.15	0.26	38.6
Anonymous-Corpus #3	Generated	71.5M	3.2	92.0	3.17	0.30	49.4

Table 9. **Encoder architecture of VideoPrism-g**. When describing the output shape, we use {temporal, spatial, and channel} as the order of dimensions when applicable, and we omit the batch size for simplicity. We highlight the dimension that a step applies to by underline. Note that the drop token or masking ratio ρ is set to 0.5 in Stage 1 and 0.65 in Stage 2.

Step	Block	Output shape
Data	-	$8 \times 288 \times 288 \times 3$
Preprocess	Patchify [1, 18, 18]	$8 \times \underline{256} \times 1408$
Drop token / Mask	Tube / BEVT	$\underline{[8 \times (1-\rho)]} \times 256 \times 1408$
Spatial encoder	MSA (6144) $\times 40$	$\underline{[8 \times (1-\rho)]} \times \underline{256} \times 1408$
Normalization	LayerNorm	$\underline{[8 \times (1-\rho)]} \times \underline{256} \times 1408$
Transpose	Switch dimension	$\underline{256} \times \underline{[8 \times (1-\rho)]} \times 1408$
Temporal encoder	MSA (6144) $\times 4$	$256 \times \underline{[8 \times (1-\rho)]} \times 1408$
Normalization	Layer Norm	$256 \times \underline{[8 \times (1-\rho)]} \times 1408$
Transpose	Switch dimension	$\underline{[8 \times (1-\rho)]} \times \underline{256} \times 1408$
Reshape	Merge dimension	$\underline{[2048 \times (1-\rho)]} \times 1408$

a learnable class token at the end of the input sentence and use its corresponding output as the text embedding.

Training. In contrast to existing methods that use batch mixing, we adopt alternating gradient descent (AGD) (Jain et al., 2017) to contrastively train our first-stage model with multiple datasets. It alternates samples from different datasets as mini-batches during training, shown effective in a multi-task and multi-dataset scenario (Akbari et al., 2023). This is particularly useful for our model to avoid easy negatives within a batch, since samples from the same dataset usually follow the same distribution and are harder to distinguish. Furthermore, we observe that the AGD approach scales well as we add more datasets or increase the size of the corpus.

Table 10. **Summary of our pretraining configurations.**

Configuration	Stage 1	Stage 2
Optimizer	AdaFactor	AdaFactor
Base learning rate	5×10^{-4}	5×10^{-4}
Learning rate schedule	linear decay	cosine decay
Warmup iterations	2×10^4	2.5×10^4
Training iterations	2×10^5	3×10^5
Weight decay	1×10^{-4}	0.05
Batch size	4096	4096
Drop token or Mask	0.5 (Tube mask)	0.65 (BEVT mask)

The training of the first-stage model follows the conventional setup of vision-language contrastive learning (Radford et al., 2021). To reduce the memory cost during pretraining, we drop 50% of video tokens as in Li et al. (2023e) and the tube masking strategy (Feichtenhofer et al., 2022) is employed for dropping tokens. The teacher model is optimized using Adafactor (Shazeer & Stern, 2018) with the batch size of 4096. We set the learning rate to 1×10^{-4} for our base model and 5×10^{-5} for the giant model. We train the first-stage model for 2×10^5 steps with 2×10^4 warm up steps and linear learning rate decay. A symmetric cross-entropy loss (Gutmann & Hyvärinen, 2010; Radford

Algorithm 1 Token shuffling pseudo-implementation.

```
# token_emb    : visible token embedding [b, m, dim]
# pos_emb      : positional embedding [n, dim]
# mask_emb     : mask embedding [dim]
# b            : mini-batch size
# m            : sequence length of visible tokens
# n            : full sequence length of input video

z = expand_dims(mask_emb, axis=[0, 1]) # [1, 1, dim]
z = tile(z, reps=[b, n - m, 1]) # [b, n - m, dim]
out_emb = concat([token_emb, z], axis=1)
out_emb = shuffle(out_emb, axis=1)

x = expand_dims(pos_emb, axis=0) # [1, n, dim]
out_emb = out_emb + x # feed out_emb to decoder
```

Table 11. **Decoder architectures of VideoPrism-g**. We highlight the dimension that a step applies to by underline. Note that the masking ratio ρ is set to 0.65.

Step	Block	Decoder output shape	
		Local	Global
Data	-	2048×1408	$[2048 \times (1-\rho)] \times 1408$
Projector	MLP	$\underline{2048} \times 512$	$[2048 \times (1-\rho)] \times 512$
Decoder	MSA (2048) $\times 4$	$\underline{2048} \times 512$	$[2048 \times (1-\rho)] \times 512$
Projector	MLP	$\underline{2048} \times 1408$	$[2048 \times (1-\rho)] \times 1408$

et al., 2021; Jia et al., 2021; Cheng et al., 2023) is used in the first-stage training.

C.2. Stage 2

Token-wise distillation. As discussed in Section 2.3.2, after training the first-stage model with contrastive learning, we train the VideoPrism video encoder with masked modeling to reconstruct the spatiotemporal embeddings from the first-stage model. As shown in Figure 3, the training pipeline of the second stage is similar to MVD (Wang et al., 2023c). After patchifying the input video sequence to a set of tokens, we apply BEVT masking (Wang et al., 2022b) with a masking ratio of 0.65 to randomly remove some of the tokens. The second-stage video encoder, which is initialized from the first-stage encoder, takes the remaining visible tokens as input and predict their embeddings. A learnable MASK token is then used to fill in the position of the masked tokens to form a full sequence together with these visible embeddings. The full sequence of embeddings is then randomly shuffled and added with positional embedding before being fed into a shallow decoder which is a four-layer Transformer. A linear layer is then used to align the output of the decoder with the embeddings of the first-stage video encoder by minimizing their cosine distance. Algorithm 1 presents a pseudocode implementation of the proposed token shuffling for masked video modeling.

Global distillation. To distill the global visual embedding from the first-stage model, we employ a four-layer Transformer decoder followed by a MAP layer to take the visible embeddings from the second-stage video encoder as input and output a global embedding. We do not apply token shuffling or add positional embedding for this decoder. We then align this second-stage global embedding to the global visual embedding from the first-stage model using a cosine distance loss. Please note that the global visual embedding from the first-stage model is predicted by the same MAP head of contrastive training in the first stage. Table 11 shows the decoder architectures in this stage.

Training. We train the second-stage video encoder using the same video clips for the first-stage model, excluding WebLI (Chen et al., 2023c), the image-based dataset. We use Adafactor (Shazeer & Stern, 2018) for optimization. The second-stage video encoder is trained with batch size 4096 and a starting learning rate of 5×10^{-4}. The learning rate is decayed to 1×10^{-5} with a cosine annealing schedule. 2.5×10^4 warm up steps are also used to linearly increase the learning rate from 0 to 5×10^{-4} at the beginning. The second-stage video encoder is trained for 3×10^5 steps. We apply the same weight for token-wise distillation loss and global distillation loss in the second-stage training.

D. Evaluation data

Table 12 summarizes all the datasets and their corresponding metrics utilized for evaluation in this paper. The evaluation datasets are categorized into four parts: general video-only understanding (VideoGLUE (Yuan et al., 2023)), video-text retrieval, captioning & QA, and CV for science. Within each category, we select representative datasets and report the standard metric on each of them.

We compare the performance of VideoPrism to the previous best-performing foundation models in Figure 2. For each dataset and task, we compute the performance gain (ΔScore) with respect to the best reported number achieved by an image or video foundation model. We collect all of them and plot in descending order.

E. VideoGLUE

E.1. Tasks and task heads for VideoPrism

We follow the VideoGLUE (Yuan et al., 2023) setup for video-only evaluations. Given a video clip of shape $T \times H \times W \times 3$, VideoPrism produces a set of visual tokens of shape $T \times \frac{H}{18} \times \frac{W}{18} \times D$, where T, H, and W are the number of frames, image height, and image width, respectively, and D is the feature length.

In all our video classification tasks, we employ a multi-head attention pooling (MAP) layer as our task head, which consists of Transformer layers with 12 heads and hidden size 768. A class token is prepended to cross-attend to all

Table 12. **Summary of evaluation datasets.** We report the corresponding standard metric for each dataset, including Top-1/5 Accuracy (Acc.) for classification and question answering, mean Average Precision (mAP) for multi-label classification, Recall@1/5 for retrieval, multi-choice retrieval accuracy (MC Acc.) for multi-choice retrieval, CIDEr score for captioning, Wu-Palmer Similarity (WUPS) index for question answering, and macro-accuracy (Macro Acc.) for the KABR dataset.

Datasets	Tasks	Zero-shot	Abbr.	Metrics
Kinetics-400 (Kay et al., 2017)	Video Classification	✗	VC	Top-1 Acc.
MiT (Monfort et al., 2019)	Video Classification	✗	VC	Top-1 Acc.
SSv2 (Goyal et al., 2017a)	Video Classification	✗	VC	Top-1 Acc.
Diving48 (Li et al., 2018)	Video Classification	✗	VC	Top-1 Acc.
Charades (Sigurdsson et al., 2016)	Video Classification	✗	VC	mAP
ActivityNet (Caba Heilbron et al., 2015)	Temporal Action Localization	✗	TAL	mAP
AVA (Gu et al., 2018)	Spatiotemporal Action Localization	✗	STAL	mAP
AVA-Kinetics (Li et al., 2020)	Spatiotemporal Action Localization	✗	STAL	mAP
MSRVTT (Xu et al., 2016)	Text-to-Video Retrieval	✓	ZST2V	Recall@1, Recall@5
MSRVTT (Xu et al., 2016)	Video-to-Text Retrieval	✓	ZSV2T	Recall@1, Recall@5
VATEX (Wang et al., 2019)	Text-to-Video Retrieval	✓	ZST2V	Recall@1, Recall@5
VATEX (Wang et al., 2019)	Video-to-Text Retrieval	✓	ZSV2T	Recall@1, Recall@5
ActivityNet (Caba Heilbron et al., 2015)	Text-to-Video Retrieval	✓	ZST2V	Recall@1, Recall@5
ActivityNet (Caba Heilbron et al., 2015)	Video-to-Text Retrieval	✓	ZSV2T	Recall@1, Recall@5
Kinetics-400 (Kay et al., 2017)	Video Classification	✓	ZSC	Top-1 & Top-5 Acc.
Kinetics-600 (Carreira et al., 2018)	Video Classification	✓	ZSC	Top-1 & Top-5 Acc.
SSv2-Temporal (Sevilla-Lara et al., 2021)	Video Classification	✓	ZSC	Top-1 Acc.
SSv2-Events (Bagad et al., 2023)	Video Classification	✓	ZSC	Top-1 Acc.
NExT-QA (ATP-Hard) (Xiao et al., 2021)	Video Classification	✓	ZSC	MC Acc.
Charades (Sigurdsson et al., 2016)	Video Classification	✓	ZSC	mAP
Charades-STA (Gao et al., 2017)	Video Classification	✓	ZSC	MC Acc.
MSRVTT (Xu et al., 2016)	Video Captioning	✓	ZSCap	CIDEr
VATEX (Wang et al., 2019)	Video Captioning	✓	ZSCap	CIDEr
YouCook2 (Zhou et al., 2018)	Video Captioning	✓	ZSCap	CIDEr
MSRVTT-QA (Xu et al., 2017)	Video Question Answering	✓	ZSQA	Top-1 Acc.
MSVD-QA (Xu et al., 2017)	Video Question Answering	✓	ZSQA	Top-1 Acc.
NExT-QA (Xiao et al., 2021)	Video Question Answering	✓	ZSQA	WUPS
Fly vs. Fly (Eyjolfsdottir et al., 2014)	Video Classification	✗	VC	mAP
CalMS21 (Sun et al., 2021a)	Video Classification	✗	VC	mAP
CRIM13 (Side view) (Burgos-Artizzu et al., 2012)	Video Classification	✗	VC	mAP
CRIM13 (Top view) (Burgos-Artizzu et al., 2012)	Video Classification	✗	VC	mAP
KABR (Kholiavchenko et al., 2024)	Video Classification	✗	VC	Macro Acc.
ChimpACT (Ma et al., 2023)	Spatiotemporal Action Localization	✗	STAL	mAP

Table 13. **Results of FM adaptation using frozen features on video understanding tasks.** The model backbones are frozen and only weights in the task heads are updated using the downstream tasks' training sets. * indicates the model is evaluated under the setting with trainable FLOPs alignment.

Methods	VC (A)		VC (M)		VC (ML)	TAL	STAL		Trainable
	K400	**MiT**	**SSv2**	**D48**	**Charades**	**ActivityNet**	**AVA**	**AVA-K**	FLOPs (B)
CLIP-B (Radford et al., 2021)	75.2	32.6	41.0	44.1	11.2	32.7	21.1	25.9	3.72
VATT-B (Akbari et al., 2021)	75.1	32.1	57.8	49.7	33.3	35.3	20.3	22.2	3.72
CoCa-B (Yu et al., 2022)	73.1	32.0	41.5	34.1	8.8	33.0	23.3	24.7	3.72
FLAVA-B (Singh et al., 2022)	71.3	29.7	40.6	45.9	12.6	32.2	18.8	21.5	3.72
VideoMAE-B (Tong et al., 2022)	65.1	23.0	53.9	59.5	11.3	33.0	16.0	19.9	3.72
InternVideo-B (Wang et al., 2022c)	69.3	26.3	58.2	55.6	13.0	33.3	13.4	15.7	3.72
UMT-B (Li et al., 2023b)	77.1	34.0	47.7	47.8	30.1	35.8	20.7	21.1	3.72
VideoPrism-B*	82.8	40.0	61.8	59.5	38.7	36.6	29.9	32.0	3.72
VideoPrism-B	84.2	40.8	63.6	67.4	40.4	36.6	30.6	31.8	9.71

Table 14. **Results of FM adaptation using frozen backbones with MLAP heads on video understanding tasks.** MLAP takes multiple frozen features from an FM as inputs and map them hierarchically for the final task prediction. Only the MLAP layer weights are updated using the downstream tasks' training sets. * indicates the model is evaluated under the setting with trainable FLOPs alignment.

Methods	VC (A)		VC (M)		VC (ML)	TAL	STAL		Trainable
	K400	**MiT**	**SSv2**	**D48**	**Charades**	**ActivityNet**	**AVA**	**AVA-K**	FLOPs (B)
CLIP-B (Radford et al., 2021)	77.1	39.0	50.1	55.8	41.5	33.9	27.7	29.6	14.9
VATT-B (Akbari et al., 2021)	75.1	35.6	58.7	60.1	58.2	35.0	22.9	24.1	14.9
CoCa-B (Yu et al., 2022)	74.2	37.2	45.9	48.4	19.6	33.3	24.4	27.0	14.9
FLAVA-B (Singh et al., 2022)	71.5	34.5	43.1	58.5	38.2	32.4	21.3	23.2	14.9
VideoMAE-B (Tong et al., 2022)	71.7	32.2	57.4	69.6	35.9	33.4	19.6	22.1	14.9
InternVideo-B (Wang et al., 2022c)	73.7	34.7	60.3	71.9	40.5	33.6	15.9	17.7	14.9
UMT-B (Li et al., 2023b)	77.5	38.0	51.2	55.5	55.8	36.0	24.6	25.8	14.9
VideoPrism-B*	83.7	43.9	64.6	70.7	56.6	37.2	31.5	33.1	14.9
VideoPrism-B	84.5	43.8	66.3	73.6	58.6	37.2	31.4	33.0	38.8

Table 15. **Results of FM adaptation using frozen backbones with low-rank adapters and task heads.** Only the weights of the low-rank adapters and task heads are updated using downstream tasks' training sets. * indicates the model is evaluated under the setting with trainable FLOPs alignment.

Methods	VC (A)		VC (M)		VC (ML)	TAL	STAL		Trainable
	K400	**MiT**	**SSv2**	**D48**	**Charades**	**ActivityNet**	**AVA**	**AVA-K**	FLOPs (B)
CLIP-B (Radford et al., 2021)	80.2	39.7	56.0	77.2	44.2	-	24.5	28.0	6.44
VATT-B (Akbari et al., 2021)	75.0	36.5	63.5	68.9	53.5	-	22.3	25.8	6.44
CoCa-B (Yu et al., 2022)	80.9	41.4	56.1	67.1	45.8	-	26.6	28.7	6.44
FLAVA-B (Singh et al., 2022)	74.7	34.1	52.1	68.4	40.8	-	17.9	23.8	6.44
VideoMAE-B (Tong et al., 2022)	73.6	30.6	61.4	76.0	43.0	-	16.6	23.3	6.44
InternVideo-B (Wang et al., 2022c)	75.5	31.3	63.9	73.6	46.2	-	19.2	25.5	6.44
UMT-B (Li et al., 2023b)	81.5	40.4	61.8	78.5	50.0	-	27.8	29.4	6.44
VideoPrism-B*	84.5	44.0	66.3	83.0	57.8	-	33.6	35.7	8.71
VideoPrism-B	85.7	43.9	68.8	85.1	60.6	-	34.1	35.8	22.8

Table 16. **Results of FM adaptation by end-to-end fine-tuning.** All the model weights are updated using the downstream tasks' training sets. * indicates the model is evaluated under the setting with trainable FLOPs alignment.

Methods	VC (A)		VC (M)		VC (ML)	TAL	STAL		Trainable
	K400	**MiT**	**SSv2**	**D48**	**Charades**	**ActivityNet**	**AVA**	**AVA-K**	FLOPs (B)
CLIP-B (Radford et al., 2021)	81.0	39.0	46.6	75.7	54.3	-	27.1	28.9	367
VATT-B (Akbari et al., 2021)	77.1	34.8	65.1	77.6	55.7	-	27.0	28.4	371
CoCa-B (Yu et al., 2022)	82.6	43.6	66.8	79.6	55.0	-	27.7	31.0	367
FLAVA-B (Singh et al., 2022)	79.1	38.3	61.1	72.0	48.6	-	22.0	25.6	367
VideoMAE-B (Tong et al., 2022)	78.7	36.1	65.5	75.5	51.4	-	23.5	26.2	367
InternVideo-B (Wang et al., 2022c)	80.1	35.9	67.0	75.8	52.2	-	27.2	29.8	367
UMT-B (Li et al., 2023b)	83.3	38.7	67.0	79.2	57.1	-	28.8	30.9	367
VideoPrism-B*	84.4	43.9	68.2	82.3	58.1	-	33.3	35.3	374
VideoPrism-B	85.7	44.2	70.0	84.9	60.1	-	33.4	35.9	977

Table 17. **Stratified average scores under four adaptation methods and the final VGS.** * indicates the model is evaluated under the setting with trainable FLOPs alignment.

Methods	Frozen	MLAP	Adapter	E2E	VGS
CLIP-B (Radford et al., 2021)	32.8	43.3	49.3	52.8	41.5
VATT-B (Akbari et al., 2021)	39.4	46.3	49.9	52.7	45.1
CoCa-B (Yu et al., 2022)	31.2	36.3	49.0	55.2	39.7
FLAVA-B (Singh et al., 2022)	31.7	39.3	44.1	49.4	38.5
VideoMAE-B (Tong et al., 2022)	32.6	40.9	45.9	51.0	39.9
InternVideo-B (Wang et al., 2022c)	33.1	42.2	47.7	52.5	41.0
UMT-B (Li et al., 2023b)	38.0	45.6	52.4	55.3	45.3
VideoPrism-B*	45.6	51.5	57.8	57.9	51.3

visual tokens from VideoPrism for final classifications. We use batch size 256 when training the task heads. We apply the same data augmentation strategies and training recipes for each individual dataset as described in VideoGLUE and perform multi-view evaluations.

Spatiotemporal action localization requires to localize person instances in an input video and recognize their actions. In our experiments, instance-level features are first RoIPooled (Ren et al., 2015) from visual tokens by using corresponding instance boxes. These features are then used to query all other visual tokens through cross-attention layers. We use a Transformer layer with 12 heads and hidden size 768 as the task heads. Final query tokens are classified via a linear classifier. We use the groundtruth instance boxes with their associated action labels for training. At test time, we use the same pretrained person detector as in Feichtenhofer et al. (2019) for person detection on AVA. On AVA-Kinetics, we use the detector described in Li et al. (2020). We train the models with batch size 256.

For temporal action localization, we only apply VideoPrism under frozen and multi-layer attention pooler (MLAP) settings, since the long video samples do not allow end-to-end tuning. In the MLAP setting, we pool features and input them to a G-TAD head (Xu et al., 2020). We use batch size 32 and train G-TAD on ActivityNet v1.3 for 10 epochs.

We employ AdamW (Loshchilov & Hutter, 2019) optimizer and cosine learning rate decay in all video-only experiments. For more details on the experiment setups, we refer readers to the VideoGLUE paper (Yuan et al., 2023).

We experiment VideoPrism under two configurations regarding different input video sizes. In the first configuration (marked by asterisk "*" in Tables 13 to 16), we use 8 frames and 252×252 image resolution for feature extraction when training video classification task heads, which results in a sequence of $8 \times 14 \times 14$ tokens. On AVA and AVA-Kinetics, video clips of shape $8 \times 288 \times 288$ (*i.e.*, token length $8 \times 16 \times 16$) are used for both training and evaluation. This configuration aligns the trainable FLOPs of VideoPrism with the other baseline models reported in Yuan et al. (2023). In the second configuration, we use video clips of shape $16 \times 288 \times 288$ as input for all the experiments. This configuration aligns with the pretraining setup with higher trainable FLOPs and accounts for the results in Section 3.1.

E.2. Adaptations

We follow Yuan et al. (2023) to report model performances under four adaptation settings, namely frozen model backbone with simple MAP heads, with MLAP heads, and with low-rank adapters, and finally end-to-end finetuning.

Frozen features with simple MAP heads update a one-layer task head only, which pools over the visual tokens from the backbone. The MLAP head upgrades from the one-layer MAP pooler by taking a stack of visual tokens as input and mapping them hierarchically for final task prediction. In our experiments, the attention pooler in MLAP has four cross-attention layers, following Yuan et al. (2023). The low-rank adaptation inserts low-rank adapter modules (Hu et al., 2022) with trainable weights into the pretrained video encoders, and uses a one-layer MAP task head. During training, both the adapter weights and task heads are updated, and the other weights from video backbones are kept frozen. We set the inner dimension of the adapter layers to be 64 for all our experiments. Finally, end-to-end finetuning is done with a one-layer MAP task head while we update weights in both the backbones and task heads.

E.3. Results

In Tables 13 to 16, we report the detailed benchmark results using the aforementioned four adaptation settings[1]. In Table 17, the stratified average scores for each FM under four adaptations are reported. We also report the final VideoGLUE score (VGS) for each FM, which weights their absolute performances with the respective adaptation costs.

Notably, from Tables 13 to 16, we can see that when evaluated under the same trainable FLOPs, VideoPrism consistently and significantly outperforms other FMs across different benchmarks and tasks. Aligning with the pretraining setup further improves the performance. The strong results under both configurations indicate the efficacy of the learned representations by VideoPrism.

From the overall benchmark results, we note that VideoPrism achieves the best across the board, surpassing existing FMs by a large margin. VideoPrism-B performs strongly on both appearance-based video understanding datasets and motion-aware recognition tasks, thanks to our two-stage pretraining design. More interestingly, VideoPrism-B improves upon baseline FMs more significantly on the regime of low adaptation costs according to Table 17.

F. Zero-shot video-text retrieval

F.1. Implementation details

In general, LiT (Zhai et al., 2022b) can be viewed as an efficient way to equip any pretrained vision encoder with zero-shot classification and retrieval capabilities. Here, we follow LiT to pair VideoPrism with a text encoder to assess its zero-shot performance on discriminative video-language tasks: video-text retrieval and video classification as text

[1]The UMT-B/16-25M checkpoint is obtained from https://github.com/OpenGVLab/unmasked_teacher/blob/main/multi_modality/MODEL_ZOO.md.

Table 18. **Zero-shot video-text retrieval on MSRVTT**. We follow the full split produced by Xu et al. (2016) which contains $2,990$ videos for testing.

Methods	MSRVTT			
	Text → Video		Video → Text	
	R@1	R@5	R@1	R@5
CLIP-B (Radford et al., 2021)	23.3	44.2	43.3	73.3
SM-B (Zeng et al., 2022)	-	-	46.9	73.5
CoCa-g (Yu et al., 2022)	30.0	52.4	49.9	73.4
VideoCoCa-g (Yan et al., 2022)	34.3	57.8	64.7	85.2
VideoPrism-B	37.0 (↑2.7)	61.5 (↑3.7)	67.7 (↑3.0)	87.5 (↑2.3)
VideoPrism-g	**39.7** (↑5.4)	**63.7** (↑5.9)	**71.0** (↑6.3)	**90.0** (↑4.8)

Table 19. **Zero-shot video classification on Kinetics-600.** Models pretrained with extra modalities in addition to vision and language (*e.g.*, audio) are marked in gray.

Methods	Top-1 Acc	Top-5 Acc
ER-ZSAR-B (Chen & Huang, 2021)	42.1	73.1
CLIP-B (Radford et al., 2021)	63.5	86.8
CoCa-g (Yu et al., 2022)	65.1	87.1
VideoCoCa-g (Yan et al., 2022)	70.1	88.9
X-CLIP-B (Ni et al., 2022)	65.2	86.1
X-Florence-B (Ni et al., 2022)	68.8	88.4
Text4Vis-L (Wu et al., 2023)	68.9	-
MAXI-B (Lin et al., 2023b)	71.5	92.5
LanguageBind-L (Zhu et al., 2024)	61.9	-
IMP-MoE-L (Akbari et al., 2023)	76.8	-
VideoPrism-B	69.7 (↓1.8)	90.6 (↓1.9)
VideoPrism-g	**75.6** (↑4.1)	**93.2** (↑0.7)

retrieval. We let the text encoder mirror the corresponding text encoder from the first-stage model and attach a MAP head to VideoPrism. Both the text encoder and MAP head are initialized from the teacher model pretrained in Stage 1. Note that as LiT, the video encoder is locked (frozen) during training.

We use exactly the same pretraining data and configurations of the first-stage model to tune the model in this stage. To further boost the model performance, we tune our model only with Anonymous-Corpus #3 in the last training epoch, whose captions are produced following Zhao et al. (2024). When evaluating the tuned models on zero-shot video classification tasks, we turn the groundtruth class labels into text descriptions with the text prompts introduced in CLIP (Radford et al., 2021).

F.2. Zero-shot classification on Charades-STA

As mentioned in Section 3.2, in order to evaluate the fine-grained temporal reasoning capability of VideoPrism, we adapt Charades-STA (Gao et al., 2017) to the zero-shot video classification task. Charades-STA is originally proposed for temporal grounding where multiple sequential descriptions are annotated with their start and end timestamps for a video. We repurpose Charades-STA for multi-choice video-to-text retrieval by trimming the video into multiple clips using the annotated timestamps. The multi-choice video-to-text retrieval then is performed by retrieving the correct description for a video clip from all sequential descriptions of this video.

Table 20. **Datasets included in Academic-Corpus.**

Datasets	# of clips
Video Story Telling (Huang et al., 2016)	3K
TACoS (Regneri et al., 2013)	4K
YouDescribe (Pitcher-Cooper et al., 2023)	19K
Charades (Sigurdsson et al., 2016; Gao et al., 2017)	20k
COIN (Tang et al., 2019)	24K
VITT (Huang et al., 2020)	35K
VLN (Voigtlaender et al., 2023)	37K
EPIC-Kitchens-100 (Dima et al., 2022)	67K
Spoken Moments in Time (Monfort et al., 2021)	481K
Ego4D (Grauman et al., 2022; Lin et al., 2022)	3.8M

F.3. Additional results on MSRVTT

In Table 18, we report zero-shot video-text retrieval results on the full split of MSRVTT produced by Xu et al. (2016), which contains $2,990$ videos for testing. We observe that VideoPrism outperforms previous methods by a large margin. More importantly, our base-scale model is also better than existing larger-scale models (*e.g.*, CoCa-g and VideoCoCa-g). These findings are consistent with the results in Table 3, which confirms the strong capability of VideoPrism on zero-shot video-text retrieval tasks. *Note that, in the appendix, all video-text retrieval results on MSRVTT are calculated using this full split, unless otherwise stated.*

F.4. Additional results on Kinetics-600

In addition to Table 4 in the main paper, we provide zero-shot video classification results on Kinetics-600 (K600) (Carreira et al., 2018) in this section. As shown in Table 19, we can find that VideoPrism achieves the best results compared with state-of-the-art FMs that are pretrained with vision and language modalities. Although LanguageBind (Zhu et al., 2024) and IMP (Akbari et al., 2023) use additional modalities (*e.g.*, audio) during pretrainig, our results are still comparable to them. More importantly, our base-scale model is able to outperform a majority of methods with even larger scales. These observations are consistent with the ones we draw from Table 4 in the main text.

G. Gluing VideoPrism with PaLM-2

In Section 3.3, we provided evidence of the strength and generalizability of VideoPrism by showing that we can easily

Table 21. **More detailed comparison to state-of-the-art methods on zero-shot video question answering.** We include additional results under the two-shot prompting and closed-vocabulary settings. We report Top-1 accuracy for both MSRVTT-QA and MSVD-QA. Methods that unfreeze their language models during training are marked in gray.

Methods	Two-shot prompting	Closed-vocab.	**MSRVTT-QA**	**MSVD-QA**
Question-answering-only models				
FrozenBiLM-L (Yang et al., 2022)	✗	✓	22.2	39.0
All-in-one models				
BLIP-B (Li et al., 2022)	✗	✓	19.2	35.2
HiTeA-B (Ye et al., 2023)	✗	✓	21.7	37.4
mPLUG-2 (Xu et al., 2023)	✗	✓	43.8	55.3
Flamingo-3B (Alayrac et al., 2022)	✓	✗	11.0	27.5
Flamingo-9B (Alayrac et al., 2022)	✓	✗	13.7	30.2
VideoPrism-B w/ PaLM-2-1B	✓	✗	19.5	36.7
VideoPrism-B w/ PaLM-2-1B	✗	✓	23.1	43.2
VideoPrism-B w/ PaLM-2-1B	✓	✓	28.5	39.5
VideoPrism-B w/ PaLM-2-8B	✓	✗	24.8	42.7
VideoPrism-B w/ PaLM-2-8B	✗	✓	23.4	43.4
VideoPrism-B w/ PaLM-2-8B	✓	✓	**32.0**	**47.1**

fuse it with a pretrained LLM decoder in a further training stage for good performance on tasks that are generative in language such as video captioning and video QA. We provide details about model training and our evaluation protocols in what follows.

Implementation. We pass the features of our video encoder through a one-layer Perceiver Resampler (Alayrac et al., 2022) that outputs a fixed number of continuous tokens representing the input video. It is always set to be 256 in our experiments. These tokens are then prepended to the embedded text prompt and fed into a LLM decoder, *i.e.*, PaLM-2 (Anil et al., 2023). The resampled features are then added with the original query features via skip connection. Note that there are two differences from the original implementation (Alayrac et al., 2022). First, a separate LayerNorm is used for query and key features as we find it works better than the shared LayerNorm. Second, we do not concatenate the key features with the query features before the cross attention, since the feature dimensions from VideoPrism is different from the pretrained PaLM-2. Otherwise, the feature dimensions would need to be projected via a linear projection layer before the concatenation, and we find it leads to unstable training. We ablate with different number of Resampler layers (*i.e.*, 1, 3, and 6) and find that the one-layer Resampler works the best in our experiments.

Model training. We train this multimodal model on a combination of video-text captioning data from the pretraining stage, an aggregated Academic-Corpus, and VQAv2 (Goyal et al., 2017b) (an image QA dataset) using a standard autoregressive language modeling loss. Table 20 lists all the datasets in Academic-Corpus, which includes Ego4D (Grauman et al., 2022), EPIC-Kitchens (Dima et al., 2022), Spoken Moments In Time (Monfort et al., 2021), *etc.*, totalling 4.4M video clips.

Both the video encoder and the LLM are kept entirely frozen during training, only the one-layer Resampler is optimized. We train VideoPrism-B with PaLM-2-1B and PaLM-2-8B separately. We set batch size to be 256 for PaLM-2-1B and 64 for PaLM-2-8B and trained for 2×10^5 steps. We use Adam optimizer (Kingma & Ba, 2015) with weight decay 1×10^{-4} and the learning rate is set to be peaked at 5×10^{-4} with warmup steps 1×10^4 and then linearly decreased. Beta1 is set to be 0.9 and Beta2 is set to be 0.999. We do not set EMA decay, L2 regularizer weight decay, and gradient clipping in the training. Each frame is center-cropped to 346 before being randomly cropped to 288 during the training and center-cropped to 288 during the evaluation. We set the maximum decoding steps to be 32 since the datasets in this work have relatively short answers. We use greedy decoding for all our experiments.

Model evaluation. We report both open-vocabulary and closed-vocabulary evaluation results for MSRVTT-QA and MSVD-QA in Table 21. For the open-vocabulary configuration, we adopt the zero-shot approach of Flamingo (Alayrac et al., 2022) and use two-shot text-only prompts from the training set on each downstream dataset. The use of two-shot text-only prompts is to guide the output style of the answers. We use the following process to select the two-shot prompts for each dataset. We first choose the two most common answers from the training set of the dataset, and then for each of them, a question is randomly drawn from ones in the training set with the corresponding answer.[2] Compared to Flamingo-9B, VideoPrism-B with PaLM-2-8B

[2]The final text-only two-shot prompts we employed are *"question: who is talking to his family? answer: man."* and *"question: what is a woman doing? answer: talk."* on MSRVTT-QA, and *"question: who is using a wrench on a pipe fitting? answer: man."* and *"question: who breaks an egg into a bowl? answer: woman."* on MSVD-QA.

shows an absolute 11.1% and 12.5% gain on MSRVTT-QA and MSVD-QA, respectively.

Additionally, for MSRVTT-QA and MSVD-QA, we experiment with the closed-vocabulary evaluation configuration, following Li et al. (2022); Yang et al. (2022). In this case, instead of directly outputting an answer via the language decoder, we score candidate answers using the log-likelihood of the decoder and choose the answer with the top score. The candidate answers are picked by taking the top-K most frequently appearing one-token answers from the training and validation sets of the dataset, where K is optimized over the validation set by ablating over the values $\{100, 250, 500, 1000, 2000\}$. For both MSRVTT-QA and MSVD-QA, we find $K = 250$ to work best. Any example where the groundtruth answer is not one of the candidate answers is automatically marked as incorrect. This method additionally steers the model towards answers that fit the exact style of the particular dataset and boosts performance further. In the closed-vocabulary evaluation configuration, VideoPrism-B with PaLM-2-8B outperforms FrozenBiLM-L by an absolute margin of 1.2% and 4.4% on MSRVTT-QA and MSVD-QA, respectively.

Recently, a number of works (Maaz et al., 2023; Lin et al., 2023a; Li et al., 2023f) have begun evaluating captioning and VideoQA tasks using an LLM-in-the-loop protocol, where an LLM such as ChatGPT[3] is used to compare predictions to ground-truth answers along a number of different dimensions (*e.g.*, correctness of information, temporal understanding, consistency). This can help mitigate the issue of metrics like exact-match and BLEU score being overly reliant on superficial token matching. We leave it to future work to compare against these models using these new protocols.

H. CV for Science

We evaluate the CV for Science datasets using frozen features with the same feature extraction setup (MAP probing) as the VideoGLUE tasks in Section E.1. The datasets are: Fly vs. Fly (Eyjolfsdottir et al., 2014) for fly video classification, CalMS21 (Sun et al., 2021a) for mouse video classification from top view, CRIM13 (Burgos-Artizzu et al., 2012) for mouse video classification with top and side views, ChimpACT (Ma et al., 2023) for chimp spatiotemporal action localization, and KABR (Kholiavchenko et al., 2024) for video classification with Kenyan animals. The domain expert models reported in the main paper are trained on the training split of each dataset, and reported originally in task programming (Sun et al., 2021b) for Fly vs. Fly, CalMS21 1D ConvNet with extra unlabelled data (Sun et al., 2021a) for CalMS21, KABR X3D (Kholiavchenko et al., 2024) for KABR, and ChimpACT SlowFast (Ma et al., 2023) for ChimpACT. For each dataset, we use the train and test splits defined by existing work, with the same metrics (mAP for all works, except KABR, which uses macro-accuracy averaged across classes). For Fly vs. Fly, we use the data split defined in Sun et al. (2021b), which includes all behaviors with more than 1000 frames of annotations in the training set. We note that following previous work (Sun et al., 2021a;b), in datasets where there are background classes, the metric is only averaged across behaviors-of-interest (not including background classes).

We extract all frames from the video at the original FPS of each dataset. We use 16 frames as input in Fly vs. Fly, CalMS21, and CRIM13, 64 frames for ChimpACT, and 16 frames with a stride of 5 for KABR, following baselines. Note that for ChimpACT, the benchmark uses groundtruth bounding boxes during training and testing, which we follow.

The training setup and implementation details are similar to the VideoGLUE frozen-backbone setting (MAP probing) in Appendix E.1. We use the AdamW (Loshchilov & Hutter, 2019) optimizer and cosine learning rate decay in the CV for science experiments. For data augmentation, we use the same ones as other video classification datasets in VideoGLUE (*e.g.*, Charades, Diving48, and MiT) for our video classification datasets. For ChimpACT (spatiotemporal action localization), we use the AVA data augmentation. We use a learning rate of 5×10^{-5} for video classification and spatiotemporal action localization, except for KABR, where the base-scale model uses 5×10^{-6} and large-scale model uses 1×10^{-6}. Following the baseline, KABR is also trained with the EQL loss (Tan et al., 2020) to account for class imbalance. Finally, all models are trained with 0.5 dropout rate.

I. Ablation studies

I.1. Data

We study how to combine datasets with different caption qualities, quantities, and distributions when training a video-text contrastive model. In Table 22, three different combination methods are considered: (1) simply mixing different datasets (denoted with "+" and "✗" for AGD); (2) training with one dataset and then continue training with another dataset (denoted with "→" and "✗" for AGD); (3) combining different datasets with AGD (denoted with "+" and "✓" for AGD). We choose two representative datasets, namely InternVid (Wang et al., 2023e) and YTT180M (Zellers et al., 2021) and report Recall@1 (R@1) for zero-shot video-to-text (ZSV2T) and text-to-video (ZST2V) retrieval for this study. We notice that simply mixing InternVid and YTT180M results in a large performance drop on VATEX

[3] https://chat.openai.com

Table 22. **Data ablation for the first-stage model.** We report both zero-shot text-to-video (ZST2V) and video-to-text (ZSV2T) retrieval results of VideoPrism-B for each configuration under the evaluation metric of Recall@1.

Data	AGD	MSRVTT		VATEX	
		ZST2V	ZSV2T	ZST2V	ZSV2T
InternVid	✗	28.9	55.8	40.8	57.3
YTT180M	✗	19.6	47.5	26.3	49.7
InternVid + YT180M	✗	29.5	56.7	29.9	41.0
InternVid → YT180M	✗	21.7	54.7	29.8	54.8
YTT180M → InternVid	✗	29.3	56.4	39.9	57.5
InternVid + YTT180M	✓	29.8	57.4	39.4	58.2
Full pretraining corpus	✓	**34.3**	**64.4**	**52.0**	**69.7**

when compared with only using InternVid. On the other hand, training on one dataset then continue training on the other dataset is highly affected by the order of datasets. For instance, compared with only using InternVid, the performance of InternVid → YTT180M drops by a large margin on both MSRVTT and VATEX, while YTT180M → InternVid improves on three out of four metrics. Hence, this approach is not scalable with the number of datasets. Alternatively, AGD consistently improves the performance on MSRVTT and ZSV2T of VATEX compared with YTT180M → InternVid and InternVid with only a slightly drop in ZST2V of VATEX. As a result, AGD is chosen for combining different training datasets when we train the video-text contrastive model.

We further report the performance of AGD with all our pretraining corpus in the last row of Table 22. We observe a large improvement across all metrics, demonstrating that AGD scales well with the number of datasets.

To better understand how our models perform when only public datasets are included for pretraining, we conduct experiments under two setups where only (1) InternVid (7M) and (2) all public datasets (150M, including InternVid, YT-Temporal-180M, and WTS-70M) are used, respectively. As shown in Table 23, we find our first-stage model has already achieved overall favorable results, while our second-stage model yields better results, especially on SSv2. As with the case with any foundation model, the pretraining data is one of the factors to improve performance, but it is not the only factor. The proposed two-stage pretraining strategy, token-wise distillation with token shuffling, and global distillation all contribute to the performance improvements.

I.2. Model design

Factorized encoder. We favored the factorized attention over joint attention in the encoder because it balances the cost (*e.g.*, memory, efficiency) and performance well. The factorized attention is especially appealing given that contrastive learning (our Stage 1) demands a large batch size.

Table 23. **Model performance when only public datasets are used for pretraining.** Results using MAP probing with frozen backbone for VideoPrism-B are reported. We report Top-1 accuracy on K400 and SSv2.

Methods	Corpus size	K400	SSv2
UMT-B (Li et al., 2023b)	25M	77.1	47.7
VideoMAE-B (Tong et al., 2022)	1M	65.1	53.9
VideoPrism-B (Stage 1)	7M	81.0	47.6
VideoPrism-B (Stage 1)	150M	81.7	50.0
VideoPrism-B (Stage 2)	7M	81.5	60.3
VideoPrism-B (Stage 2)	150M	**82.7**	**60.5**

Table 24. **Joint *vs.* factorized attention.** Results of the first-stage VideoPrism-B model are reported. We report Top-1 accuracy on K400 and SSv2 using MAP probing with frozen backbone.

Encoder architectures	MSRVTT		K400	SSv2
	ZST2V	ZSV2T	VC	VC
Joint attention	**34.2**	64.0	**84.5**	52.6
Factorized attention	**34.2**	**64.6**	82.9	**55.5**

We note that the original ViViT paper (Arnab et al., 2021) also recommended the factorized-attention architecture over the other variants. In Table 24, we can see that the two attention schemes lead to similar performance on text-video retrieval tasks, while the joint attention gives rise to slightly better accuracy on K400 and yet worse on SSv2. The overall performances of these two models are similar, reinforcing that the factorized attention is probably a better choice given its smaller memory footprint.

Initialization. To understand how the end results vary with respect to the initialization from different image models, we replace CoCa (Yu et al., 2022) with CLIP (Radford et al., 2021) as the spatial encoder in our model. We then experiment with two variants: (1) freezing CLIP while training the temporal layers and (2) unfreezing all weights in training. Note that we use CLIP-B/14 and all the public datasets in our pretraining corpus for model training in this experiment. Results are shown in Table 25. We can see that the results using CLIP for initialization are still comparable with those with the CoCa initialization, and unfreezing the CLIP weights benefits all evaluation benchmarks in the table.

Masking method. We then study the impact of masking method and masking ratio on the second-stage model using VideoPrism-B as an example. In Figure 6, we compare the performance of the second-stage model under tube masking (Tong et al., 2022) and BEVT masking (Wang et al., 2022b) with various masking ratios on different video focused tasks. We notice that BEVT masking outperforms tube masking in most cases. When comes to the masking ratio, BEVT masking with masking ratio 0.65 and 0.75

Table 25. **Effects of model initialization.** Results of the first-stage VideoPrism-B model are reported. We report Top-1 accuracy on K400 and SSv2 using MAP probing with frozen backbone.

Configurations	MSRVTT		K400	SSv2
	ZST2V	ZSV2T	VC	VC
CLIP (Radford et al., 2021) (frozen)	27.6	52.4	73.1	39.8
CLIP (Radford et al., 2021)	27.4	53.3	79.6	**51.1**
CoCa (Yu et al., 2022)	**29.9**	**57.3**	**81.7**	50.0

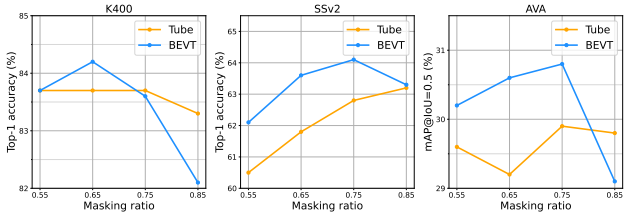


Figure 6. **Ablation study for second-stage masking strategy and masking ratio.** Results using MAP probing with frozen backbone for VideoPrism-B are reported.

have similar performance and outperform the other masking ratios. As a result, if not otherwise specified, all the second-stage models are trained with the BEVT masking with 0.65 masking ratio.

Token shuffling and global distillation. We then study the performance of token shuffling and global distillation which are the two new techniques introduced in our masking distillation method. We show the results of the second-stage model (VideoPrism-B) without token shuffling or global distillation and compare the results with the full second-stage model on video classification (K400 and SSv2) and spatiotemporal action location (AVA) tasks in Table 26. From this table, we notice that both token shuffling and global distillation help improving the performance of the second-stage model by a large margin. Especially, token shuffling improves the performance of the second-stage model on motion-focused video classification dataset SSv2 by 1.8%. We believe that token shuffling introduces a harder learning objective to the second-stage model that is akin to Jigsaw puzzle (Noroozi & Favaro, 2016), forcing the second-stage model to better understand the motion in the video. Global distillation, on the other hand, can further boost the performance of the second-stage model on appearance-based video tasks (0.8% on K400 and 1.6% on AVA) and plays an important role in training a good second-stage model.

The second-stage training. Figure 7 compares the second-stage model with the first-stage model on different video datasets across a variety of tasks using VideoPrism-B. Specifically, for video-only tasks such as video classification (K400 [VC] and SSv2 [VC]) and spatiotemporal action

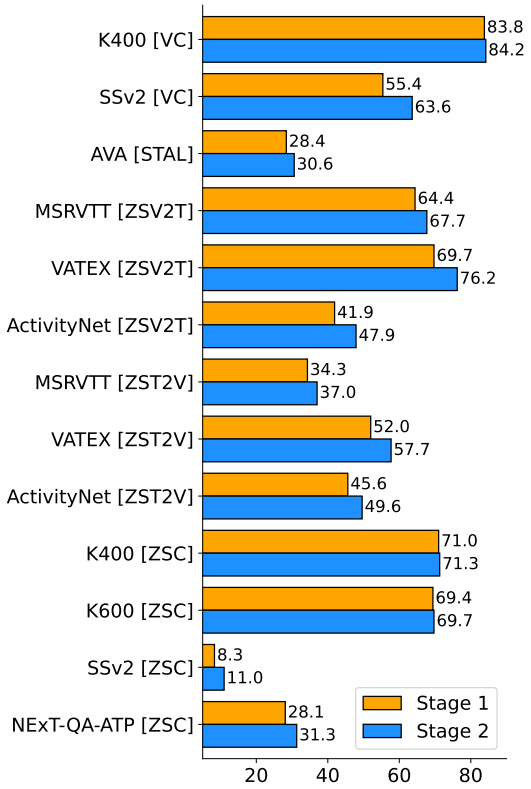


Figure 7. **Comparison between the first-stage and second-stage models of VideoPrism-B on video understanding tasks.** For video-only tasks, results are from using MAP probing with a frozen backbone.

localization (AVA [STAL]), we apply the frozen feature and only update the weights in the task head. We report top-1 accuracy for video classification and mAP@IoU=0.5 for spatiotemporal action localization, respectively. For zero-shot video-to-text retrieval (MSRVTT [ZSV2T], VATEX [ZSV2T], and ActivityNet [ZSV2T]), zero-shot text-to-video retrieval (MSRVTT [ZST2V], VATEX [ZST2V], and ActivityNet [ZST2V]), and zero-short video classification (K400 [ZSC], K600 [ZSC], SSv2 [ZSC], and NExT-QA-ATP [ZSC]) tasks, we apply LiT (Zhai et al., 2022b) to learn a text encoder paired up with the second-stage model as described in Section 3.2 and report Recall@1 (R@1). We notice that the second-stage training significantly improves the performance of the video encoder compared with the first-stage model across all video tasks on different datasets, strongly demonstrating the effectiveness of the proposed two-stage training.

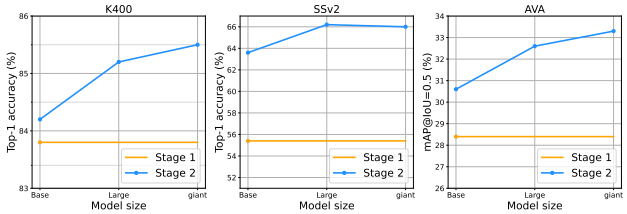
I.3. Scaling properties

In Figure 8a, we study the scaling behavior of our models by keeping the data fixed. We find that both our first-stage model and second-stage model scale well with the model

Table 26. **Ablation study for the second-stage model training strategy.** Results using MAP probing with frozen backbone for VideoPrism-B are reported. We report Top-1 accuracy on K400 and SSv2, and mean average precision (mAP) on AVA.

Models	K400	SSv2	AVA
Full configuration	**84.2**	63.6	**30.6**
w/o token shuffling	83.6 (↓0.6)	61.8 (↓1.8)	29.4 (↓1.2)
w/o global distillation	83.4 (↓0.8)	**64.2** (↑0.6)	29.0 (↓1.6)

(a) Scaling student and teacher model size.



(b) Scaling student model size with a fixed *Base* teacher model.

Figure 8. **Preliminary studies on model scaling.** Results using MAP probing with frozen backbone are reported. We report Top-1 accuracy on K400 and SSv2, and mean average precision (mAP) on AVA.

size. Interestingly, the second-stage model shows consistent improvements over the first-stage model of around 8% on SSv2 and 2.2% on AVA, across the model sizes. In Figure 8b, we scale the second-stage model by fixing the first-stage model to be of Base size. For *Large* and *giant* second-stage models, as they are incompatible with the first-stage model of Base size, we initialize them with the corresponding image model of CoCa (Yu et al., 2022). We observe that even with a fixed first-stage model, our second-stage models still show a reasonable scaling capability.

In Table 22, we demonstrate strong data scaling capability of the first-stage model where the model trained on the pretraining corpus outperforms the one trained on InternVid (Wang et al., 2023e) by 5.4% on MSRVTT ZST2V retrieval and 11.2% on VATEX ZST2V retrieval. This motivates us to also study the data scaling ability of our second-stage model. An interesting aspect of our second-stage training is that it works with video-only data without annotations. This equips us to economically increase the corpus size during the second-stage training. To test the benefit of data scaling, we mine additional 280M video clips without annotations from YouTube and add them to our second-stage training. We use model with *Large* size as an example and train the first-stage model using pretraining corpus. We then compare the second-stage model trained only on the pretraining corpus with that trained with both the pretraining corpus and the additional clips in Table 27. We can see that our second-stage model scales well with data. Note that prior work on masked modeling either does not demonstrate good data-scaling properties or shows marginal improvements with data scaling (Feichtenhofer et al., 2022; Tong et al., 2022).

Table 27. **Studies on data scaling for the second-stage model.** We report results of the *Large* model using MAP probing with frozen backbone and only the full pretraining corpus is used to train the first-stage model.

Data	# of clips	K400	SSv2	AVA
Full pretraining corpus	618M	85.8	66.4	33.1
+ additional video-only	898M	86.1	66.7	33.7