# VideoGLUE: Video General Understanding Evaluation of Foundation Models

**Anonymous authors**
**Paper under double-blind review**

## Abstract

We evaluate the video understanding capabilities of existing foundation models (FMs) using a carefully designed experiment protocol consisting of three hallmark tasks (action recognition, temporal localization, and spatiotemporal localization), eight datasets well received by the community, and four adaptation methods tailoring an FM for downstream tasks. Furthermore, we jointly profile FMs' efficacy and efficiency when adapting to general video understanding tasks using various cost measurements under different scenarios, namely training, inference, and storage. Our main findings are as follows. First, task-specialized models significantly outperform the six FMs studied in this work, in sharp contrast to what FMs have achieved in natural language and image understanding. Second, video-native FMs, whose pretraining data contains the video modality, are generally better than image-native FMs in classifying motion-rich videos, localizing actions in time, and understanding a video of more than one action. Third, the video-native FMs can perform well on video tasks under light adaptations to downstream tasks (e.g., freezing the FM backbones), while image-native FMs win in full end-to-end finetuning. The first two observations reveal the need and tremendous opportunities to conduct research on video-focused FMs, and the last confirms that both tasks and adaptation methods matter when it comes to the evaluation of FMs. We will release our code upon acceptance.

## 1 Introduction

Foundation models (FMs) are a term coined by Bommasani et al. (2021), referring to "any model that is trained on broad data that can be adapted (e.g., finetuned) to a wide range of downstream tasks." Some representative FMs include but are not limited to BERT (Devlin et al., 2018), GPT-3 (Brown et al., 2020), CLIP (Radford et al., 2021), and ALIGN (Jia et al., 2021). This work primarily investigates the video understanding capabilities of seven visual and multimodal FMs: CoCa (Yu et al., 2022), CLIP (Radford et al., 2021), FLAVA (Singh et al., 2022), DINO-v2 (Oquab et al., 2023), VideoMAE (Tong et al., 2022), VATT (Akbari et al., 2021), and InternVideo (Wang et al., 2022b). We select these models because they are amendable for the video understanding and make their checkpoints accessible to us.

It is nontrivial to evaluate FMs. In contrast to "specialist" models developed for a particular task, FMs are considered as "generalists" that learn shareable meta-knowledge across tasks so that one can quickly adapt them to achieve superior performance on various downstream tasks. Hence, *both the tasks and adaptation methods matter when it comes to the evaluation of FMs.* However, the community has not reached a consensus on these two aspects. FM developers select their own different sets of downstream tasks — interestingly, often covering no video or only appearance-rich video classification tasks (Buch et al., 2022; Lei et al., 2023). Moreover, they rely on distinct adaptation methods, making apples-to-apples comparisons challenging and causing mismatches with the FMs' actual use cases.

To this end, we propose to evaluate FMs' video understanding capabilities using a carefully designed experiment protocol, named VideoGLUE, consisting of three hallmark tasks (action recognition, temporal localization, and spatiotemporal localization), eight datasets well received by the research community, and four model adaptation methods tailoring a foundation model for downstream tasks. The tasks examine an FM from

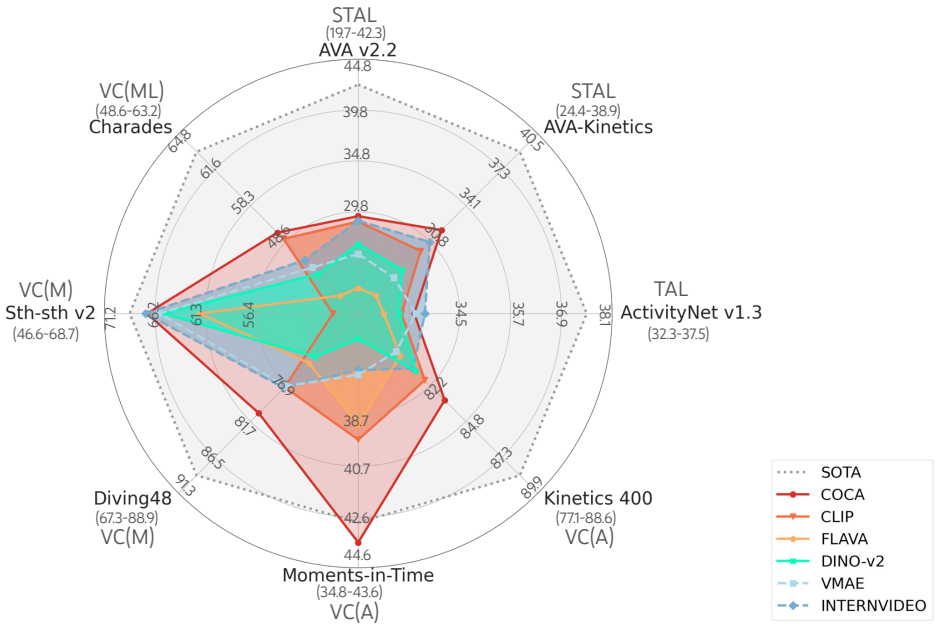


Figure 1: FMs with end-to-end tuning vs. state-of-the-art task-specialized models on video understanding. On TAL, we report frozen eval results following common practice. Unlike natural language and image understanding, video tasks are where FMs generally fall behind "specialists". VC(A), VC(M), and VC(ML) stand for appearance-focused, motion-focused, and multi-labeled video classification, respectively. STAL stands for spatiotemporal action localization, and TAL stands for temporal action localization. For each task, we include the (min-max) range shown in the figure.

various aspects needed for understanding video. The "all-around" adaptations represent the main use cases of FMs in the literature and, more importantly, allow us to thoroughly probe an FM's potential in video understanding.

Why do we specifically focus on videos? The main motivation is to promote video understanding in the evaluation of FMs. More concretely, we test the following conjectures through this work. First, FMs' high performance on existing evaluation suites does not necessarily indicate their potential in video since these suites either lack video-specific tasks or selectively choose video tasks whose appearance feature is more important than motion — InternVideo (Wang et al., 2022b) is an exception as discussed in the next paragraph. Second, many existing FMs probably cannot heed motion in video, given that they learn primarily from static images (Radford et al., 2021; Singh et al., 2022; Yu et al., 2022) or short video clips containing limited motion (Feichtenhofer et al., 2022; Wang et al., 2022b). Third, popular adaptation methods (e.g., finetuning all weights) cannot supplement FMs with all the cues needed to recognize motion-rich actions and localize entities temporally and/or spatiotemporally, as elaborated in Sections 4.1 and 4.2.

While our work is not the first to emphasize the evaluation of FMs, it is unique on multiple fronts. Unlike ELEVATER (Li et al., 2022a)'s target of evaluating language-augmented FMs, we consider all FMs adaptable to video understanding which does not necessarily involve language. Unlike Perception Test (Patraucean et al., 2024)'s coverage of a broad spectrum of perception tasks, we focus on video, allowing us to cover various aspects of this vertical domain. Interestingly, many of our datasets also appear in InternVideo (Wang et al., 2022b), a video-oriented FM. However, we promote model adaptation methods as an inherent part of the evaluation protocol — a consistent set of diverse adaptation methods is necessary to provide FMs ample opportunities to expose their video understanding capabilities. Moreover, unlike InternVideo's focus on their single FM, we evaluate FMs developed by different research groups in an uniform experiment protocol — the first of its kind for visual and multimodal FMs, to the best of our knowledge.

Our main findings are as follows. First, task-specialized models still significantly outperform the seven FMs studied in this work (see Figure 1), in sharp contrast to what FMs have achieved in natural language (OpenAI,

2022; Roberts et al., 2022) and image understanding (Radford et al., 2021; Yu et al., 2022; Chen et al., 2022). Hence, there is a need and tremendous opportunities to research video-focused FMs. Second, video-native FMs, whose pretraining data contains the video modality, are generally better than image-native FMs in classifying motion-rich videos, localizing actions in time, and understanding a video of more than one action. Third, the video-native FMs can perform well on video tasks under light adaptations to downstream tasks (e.g., freezing the FM backbones), while image-native FMs win in full end-to-end finetuning. This observation confirms that both tasks and adaptation methods matter when it comes to the evaluation of FMs.

## 2 Related work

**Foundation models.** One common type of FMs are Large Language Models (LLMs) trained to acquire generic, transferable, and diverse representations that can enable sample-efficient learning and knowledge transfer across a broad range of downstream tasks. FMs are often trained with simple self-supervised learning objectives such as predicting the next token in a sentence (e.g., GPT-3 (Brown et al., 2020), PaLM (Chowdhery et al., 2022)), or denoising the masked tokens (e.g., BERT (Devlin et al., 2018), UNILM (Dong et al., 2019), and BEiT (Bao et al., 2021)). An intriguing characteristic of FMs is their ability to gradually acquire new capabilities as the model grows and the training data size increases, despite being trained on simple learning objectives (Wei et al., 2022). For example, PaLM (Chowdhery et al., 2022; Anil et al., 2023), a massive LM with 540 billion parameters has started to show new capabilities in tasks such as explaining jokes, solving math, and performing common-sense reasoning when scaled to over 100B parameters.

In addition to self-supervised transformers, FMs in computer vision also encompass transformers specifically trained to align image-text paired data. These FMs use learning objectives include contrastive learning (e.g., CLIP (Radford et al., 2021)), denoising masked tokens (e.g., BEiT-3 (Wang et al., 2022a)), predicting the next token in a single modality (e.g., DALL-E (Ramesh et al., 2021)) or in the interleaved image-text sequence (e.g., Flamingo, KOSMOS-1 (Huang et al., 2023)). Recent FMs are also trained on a mixture of these objectives (e.g., CoCa (Yu et al., 2022), FLAVA (Singh et al., 2022), MAE (He et al., 2022)). For example, MAE combines autoencoder reconstruction objective jointly with the denoising objective (He et al., 2022) that was extended to video (Feichtenhofer et al., 2022; Tong et al., 2022). Our criteria of choosing foundation models to study are primarily based on the definition of FMs, and their amendability on video understanding and accessibility of checkpoints. We leave some models (e.g. detection, segmentation models) out of the scope of this work, because of their current lack of generalization on video understanding tasks. Finally we choose seven representative FMs, i.e., CoCa (Yu et al., 2022), CLIP (Radford et al., 2021), FLAVA (Singh et al., 2022), DINO-v2 (Oquab et al., 2023), VideoMAE (Tong et al., 2022), VATT (Akbari et al., 2021), and InternVideo (Wang et al., 2022b).

**Evaluation of foundation models.** As the mission of FMs is to enable sample-efficient knowledge transfer, the design of downstream tasks is critical to evaluate the capabilities and limitations of these models. The evaluation of FMs is pioneered by the NLP researchers. For example, GLUE (Wang et al., 2018a) and SuperGLUE (Wang et al., 2019) introduced a suite of tools for evaluating language understanding tasks. The authors utilized established public benchmarks and provided tools for evaluating, probing, and benchmarking pretrained FMs, allowing for a comparison to human baselines. ELEVATER (Li et al., 2022a) introduced this concept to vision FMs along with a toolkit for evaluating vision-language tasks, including knowledge augmentation, hyperparameter tuning, and three adaptation techniques. In parallel, there have been attempts to establish a diagnostic benchmark for perceptual understanding of the world. For instance, Perception Test (Patraucean et al., 2024) crowd-sourced 11K videos in which about 100 users performed scripted activities. This benchmark (Patraucean et al., 2024) comprises videos filmed by only about 100 participants, which may not provide the same level of domain coverage and diversity as the other FM evaluation works mentioned earlier.

**Evaluation of video foundation models.** While some vision-language FMs have incorporated video tasks, their evaluation typically follows that of static images and neglects the unique aspects of video spatial-temporal modeling and reasoning. To our knowledge, no previous work has been solely dedicated to evaluating video FMs. The closest work to ours are InternVideo (Wang et al., 2022b) and VideoMAE (Tong et al.,

Table 1: Foundation models studied in this work (MxM stands for Masked Image/Language/Video Modeling).

| Foundation Model | Modality | Pretraining Data | Pretraining Objective |
|---|---|---|---|
| CoCa | Image + Text | JFT3B + ALIGN | Contrastive + Captioning |
| CLIP | Image + Text | WebImageText | Contrastive |
| FLAVA | Image + Text | PMD | Contrastive + MIM + MLM |
| DINO-v2 | Image | LVD-142M | MIM + DINO |
| VideoMAE | Video | K400 | MVM |
| InternVideo | Video | UnlabeledHybrid | MVM + Contrastive |
| VATT | Video + Audio + Text | HT100M | Contrastive |

2022), which introduce new FMs and show their superiority over several dozen video datasets. There are two key differences to the prior works. First, our evaluation is video-centric using the tasks that require motion understanding or long-term temporal reasoning. Second, instead of promoting new video FMs, our work proposes no new models and is solely dedicated to evaluating current and future video FMs in an impartial reproducible experimental setup. Concretely, our goal is to provide tools for probing and benchmarking FMs on motion tasks in various setting include using the parameter-efficient adapter.

## 3 Tasks and adaptation methods both matter when evaluating foundation models

This section describes our video general understanding evaluation (VideoGLUE) benchmark. We first introduce the visual and multimodal FMs evaluated in this work. Then we discuss the video-focused downstream tasks and methods to adapt an FM to the tasks. The former concretizes the video understanding capabilities we want to evaluate from an FM, while the latter provides various paths for an FM to showcase the corresponding capabilities.

### 3.1 Foundation models for video understanding

We are interested in examining which FMs are good at solving video tasks, what makes them better than others in the video domain, and how to best adapt them to video understanding. Table 1 shows the seven FMs we gained access to via public repositories or personal communications. Thanks to the powerfulness and scalability of transformer architecture, most developed FMs converge to adopt the vision transformer architecture. Thus for all evaluated FMs, we intentionally choose the ViT-B variant to bring fair comparison into our benchmarks. We also notice in previous literature, models may be evaluated with different number of frames and resolutions, resulting in unfair comparison (Yan et al., 2022; Feichtenhofer et al., 2021). In VideoGLUE, we control the number of tokens observed by the ViT to be consistent for different FMs on the same task. For detailed setup on each dataset, please refer to supplementary materials D.

### 3.2 Video understanding tasks

Like objects' role in image understanding, actions are the core of video understanding, leading us to select tasks and datasets that *recognize* and *localize* actions in time and space. Table 2 provides a quick summary. Next, we explain the rationale behind the particular choices of datasets and postpone the datasets' details to the supplementary materials B. In table 3 we show the publishing year, venues and citations to demonstrate the popularity and community acceptance of datasets of our choices.

#### 3.2.1 Recognizing actions

**General actions.** We first include the action recognition datasets of Kinetics400 (K400) (Kay et al., 2017), Moments-in-Time (MiT) (Monfort et al., 2019), and Charades (Sigurdsson et al., 2016), considering their popularity that they are being complementary to each other. Data domain coverage is an important factor when designing benchmarks for FMs, as nowadays FMs are typically trained on massive data sources.

Table 2: Summary of statistics, video properties, and data sources of each dataset. Tasks involved are spatiotemporal action localization (STAL), temporal action localization (TAL), and video classification (VC). Column "Num. videos" contains video examples in train/evaluation splits, respectively.

| Task | Dataset | Num. videos | Avg. length | Data source | Note |
|---|---|---|---|---|---|
| VC | Kinetics400 | 235,693 / 19,165 | 10 seconds | Web | holistic, appearance |
| | Moments-in-Time | 791,246 / 33,898 | 3 seconds | Web | holistic, appearance |
| | Sth-sth v2 | 168,913 / 24,777 | 2-6 seconds | Crowd-source | holistic, motion |
| | Diving48 | 15,027 / 1,970 | 5 seconds | Web | holistic, motion |
| | Charades | 7,811 / 1,814 | 30 seconds | Crowd-source | multi-label, long-clip |
| STAL | AVA v2.2 | 210,634 / 57,371 | 15 mins | Movie | spatiotemporal, instance |
| | AVA-Kinetics | 354,201 / 91,919 | 10 seconds | Web | spatiotemporal, instance |
| TAL | ActivityNet v1.3 | 10,002 / 4,926 | 5-10 mins | Web | temporal |

Table 3: Summary of datasets publishing year, venue and citations (as of July 2024).

| Task | Dataset | Year | Venue | Citation |
|---|---|---|---|---|
| VC | Kinetics400 (Kay et al., 2017) | 2017 | arXiv | 4284 |
| | Moments-in-Time (Monfort et al., 2019) | 2018 | TPAMI | 614 |
| | Sth-sth v2 (Goyal et al., 2017) | 2017 | ICCV | 1450 |
| | Diving48 (Li et al., 2018) | 2018 | ECCV | 338 |
| | Charades (Sigurdsson et al., 2016) | 2016 | ECCV | 1352 |
| STAL | AVA (Gu et al., 2018) | 2018 | CVPR | 1119 |
| | AVA-Kinetics (Li et al., 2020) | 2020 | arXiv | 138 |
| TAL | ActivityNet v1.3 (Fabian Caba Heilbron & Niebles, 2015) | 2015 | CVPR | 2713 |

Regarding data sources, K400 videos are from Youtube, MiT draws videos from different Web venues, while Charades contains scripted indoor videos. Internet often returns entertaining and atypical videos, while Charades contains typical everyday videos (Sigurdsson et al., 2016). Regarding action labels, the datasets differ in granularities and real-life scenarios, a verb defines an action in MiT, K400 groups actions by verb-subject pairs, and Charades actions are about indoor activities. Regarding the average length, K400 and MiT videos are between 3 and 10 seconds, each with one action label, while Charades videos are about 30 seconds, each with multiple actions.

**Fine-grained motion-focused actions.** We also include Something-something-v2 (SSv2) (Goyal et al., 2017) and Diving48 (D48) (Li et al., 2018) as another two action recognition datasets, whose actions are fine-grained and motion-focused. SSv2 contains 174 human hand gestures as action labels, such as putting something into something, turning something upside down, and covering something with something. The videos are mostly focusing on hand-object interactions. D48 is all about competitive diving recordings collected from Web sources. Notably, in this dataset the foreground objects' motion is a more significant discriminative cue than their appearance.

#### 3.2.2 Localizing actions

The videos in action recognition are trimmed, but actions could occur anywhere in a video in the wild. Hence, temporal and spatiotemporal action localization is also crucial to video understanding. Accordingly, we choose three datasets for the experiments: the action localization track of ActivityNet v1.3 (ANet) (Fabian Caba Heilbron & Niebles, 2015), Atomic Visual Actions (AVA) (Gu et al., 2018), and AVA-Kinetics (AVA-K) (Li et al., 2020). The last two require a model to localize and recognize actions in both time and space, and their underlying videos are movies and general YouTube videos, respectively.

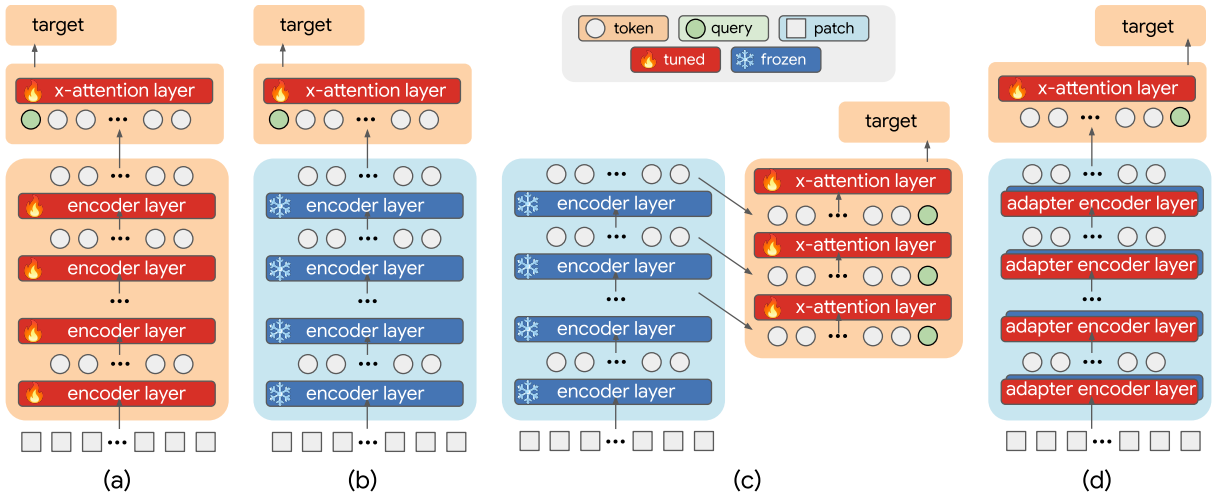


Figure 2: We study four adaptation methods to apply a foundation model (FM) to video understanding downstream tasks: (a) end-to-end finetuning, (b) frozen backbone evaluation, (c) frozen features with multi-layer attention pooler (MLAP), and (d) a low-rank adapter.

### 3.3 Adaptation methods

In this section, we detail the task-specific neural architecture design and adaptation methods when applying FMs to downstream tasks.

#### 3.3.1 Modifying foundation model architectures for downstream tasks

Given an $\textsc{fm}(\cdot)$, we can apply $\textsc{fm}(\cdot)$ to a video clip $C$ to extract a set of $k$ feature maps $\{F\}^k = \textsc{fm}(C), F \in \mathbb{R}^{n \times h \times w \times c}$, where $k$ is the number of endpoint layers from an FM, and $n, h, w, c$ are respectively a feature map's length, height, width, and number of channels.

For video classification tasks, we cast a feature map $F$ as $n \times h \times w$ tokens and aggregate them into a global representation using a learnable query token $\tau$ and lightweight cross-attention layers (Dosovitskiy et al., 2020). For spatiotemporal action localization, following the standard practice (Feichtenhofer et al., 2019; Tong et al., 2022), we first detect humans on key-frames using a human detector (Ren et al., 2015), producing a set of human bounding boxes $B$. We then apply the RoI pooling operation (Jaderberg et al., 2015) that takes both the feature map $F$ and box coordinates $B$ as inputs and outputs one feature vector per box as the query token, $\tau = \textsc{RoIPool}(F, B)$, followed by the same cross-attention layers as in video classification. For both groups of tasks, we stack a linear classifier on top of the task token's last-layer encoding for final classification:

$$p = \textsc{LinearClassifier}(\textsc{CrossAttention}(\tau, F)). \tag{1}$$

For temporal action localization, we first perform feature extraction in a sliding window manner, resulting in a sequence of globally average pooled features $\{\textsc{AvgPool}(F_1), \cdots, \textsc{AvgPool}(F_t)\}$ for each video. Following a popular choice of prior works (Alwassel et al., 2021; Ju et al., 2022; Liu et al., 2022), we employ G-TAD (Xu et al., 2020) as our task head for predicting the action category and its start and end timestamps.

#### 3.3.2 Adapting modified foundation model to downstream tasks

Adapting the modified FMs to a downstream task is to tune their weights. Then, we immediately have two basic adaptation strategies: 1) full finetuning to update all weights in the original FM plus the task head and 2) freezing FM weights and only updating newly added weights. The choice of the adaptation methods depends on specific application scenarios such as computation and memory constraints. We argue that an ideal FM should perform well across various adaptation methods to support the breadth of use cases.

**End-to-end finetuning.** End-to-end finetuning is the most common FM evaluation method for videos (Akbari et al., 2021; Feichtenhofer et al., 2022; Tong et al., 2022; Wang et al., 2022b), but it requires the deployment of a separate and possibly expensive FM for each downstream task. When finetuning all weights in the modified FMs, we limit cross-attention to a single transformer layer with 12 heads and hidden size 768. We vary learning rates and weight decays for each experiment to ensure every FM is configured to its best setup. Figure 2(a) illustrates this end-to-end finetuning.

**Freezing foundation model weights.** Linear probing and cross-attention based pooling over frozen FM features are routinely used to test the strength of the FM representation (Tong et al., 2022; Yu et al., 2022; Singh et al., 2022; He et al., 2022; Lin et al., 2022). In practice, adapting task-specific heads with a frozen FM allows us to deploy the same FM for multiple tasks. If we use light-weight heads over the FM features, then a single FM inference can serve multiple tasks efficiently in terms of both compute and memory. To this end, we examine two variations with a frozen FM, one with a single cross-attention layer and the other with multiple layers. The first results in exactly the same model architectures as in end-to-end finetuning (Figure 2(b)), and the second allows us to leverage an FM's hierarchical features beyond its last endpoint layer (Figure 2(c)). First, the frozen features are extracted from the last $k$ layers, $F_{N-k+1}$, $F_{N-k+2}$, ..., $F_N$. Then, attention pooling is applied between a learnable token $\tau$ and the features $F_{N-k+1}$ using multi-head cross-attention (MHCA). The output of this layer serves as the query token for the next round of attention pooling with the features $F_{N-k+2}$. This process is repeated for $k$ rounds:

$$\begin{aligned} \tau_{N-k+1} &= \text{MLP}(\text{MHCA}(\tau, F_{N-k+1})) \\ \tau_{N-k+2} &= \text{MLP}(\text{MHCA}(\tau_{N-k+1}, F_{N-k+2})) \\ &\dots \\ \tau_N &= \text{MLP}(\text{MHCA}(\tau_{N-1}, F_N)) \end{aligned} \tag{2}$$

where $k = 4$ in our experiments, and the final classifier is $p = \text{LinearClassifier}(\tau_N)$.

**Freezing foundation model weights with low-rank adaptation**. Finally, we explore a frozen FM beyond the last $k$ layers using a low-rank adapter (Hu et al., 2021), which is a bottleneck architecture that projects a feature tensor into a low-dimensional space and then up-samples to the original space. The bottleneck space's dimension is 64 in our experiments. Inserting a few adapter layers with trainable weights $\{w\}$ into the pretrained FM while keeping all FM's weights frozen, the feature adapter is more parameter-efficient than end-to-end finetuning the whole network while achieving better performance than simply adding a task head to the frozen FM. Essentially, the adapter leads to a new FM with some trainable weights $\{w\}$: $\tilde{F} = \widetilde{\text{FM}}(C, \{w\})$, such that the output feature maps remain the same in shape as the original FM's output (Figure 2(d)). Hence, different pooling schemes and task heads aforementioned could be applied to the extracted feature map $\tilde{F}$. For simplicity, we still choose the single-layer cross-attention as the default task head due to its computation efficiency and performance.

The low-rank adaptation allows a single FM for multiple tasks, in contrast to the per-task models in end-to-end finetuning. However, it incurs a per-task forward pass at inference time, being less efficient than the task-specific heads over frozen features.

## 4 Experiments

### 4.1 End-to-end finetuning

Table 4 shows the end-to-end finetuning results of six FMs on eight datasets. We split the FMs into two groups based on their input modalities at the time of pretraining: CoCa, CLIP, FLAVA and DINO-v2 are image-native FMs, and VideoMAE, VATT, and InternVideo are video-native. The datasets span video classification (VC) and spatiotemporal action localization (STAL). Note that it is infeasible to end-to-end fine-tune or lora fine-tune the vision encoder on TAL task, because the videos in ANet are typically long (up to 30 mins). We follow the common practice of pre-computing visual features in a sliding window manner offline, and training a temporal detection network on top of the visual features (Wang et al., 2021; Zhang et al., 2022). We will report TAL results in the next section. We draw the following observations from Table 4.

Table 4: Evaluating FMs when adapted to video understanding tasks using end-to-end finetuning. We report the Top-1 accuracy on K400, MiT, SSv2 and D48, MAP on Charades and ANet, and mAP@IOU0.5 on AVA and AVA-K.

| Model | VC (A) | | VC (M) | | VC (ML) | TAL | STAL | | AVG |
|---|---|---|---|---|---|---|---|---|---|
| | K400 | MiT | SSv2 | D48 | Charades | ANet | AVA | AVA-K | |
| CoCa | **82.6** | **43.6** | 66.8 | **79.6** | 55.0 | – | **27.7** | **31.0** | 55.2 |
| CLIP | 81.0 | 39.0 | 46.6 | 75.7 | 54.3 | – | 27.1 | 28.9 | 52.8 |
| FLAVA | 79.1 | 38.3 | 61.1 | 72.0 | 48.6 | – | 22.0 | 25.6 | 49.4 |
| DINO-v2 | 80.3 | 38.7 | 60.9 | 70.2 | 52.4 | – | 24.6 | 27.0 | 50.5 |
| VideoMAE | 78.7 | 36.1 | 65.5 | 75.5 | 51.4 | – | 23.5 | 26.2 | 51.0 |
| InternVideo | 80.1 | 35.9 | **67.0** | 75.8 | 52.2 | – | 27.2 | 29.8 | 52.5 |
| VATT | 77.1 | 34.8 | 65.1 | 77.6 | **55.7** | – | 27.0 | 28.4 | 52.7 |
| Task-specialized | 88.6<br>TubeViT | 42.7<br>UniformerV2 | 68.7<br>MViT | 88.9<br>AIM | 63.2<br>MoViNet | 37.5<br>PRN | 42.3<br>RAFT | 38.9<br>RAFT | – |

*FMs underperform task-specialized models on video tasks in general.* Table 4's last row collects the state-of-the-art results on the eight datasets, each obtained by a task-specialized model with comparable architecture or size to ours in the prior work. Specifically, those task-specialized models are RAFT (Rajasegaran et al., 2023), PRN (Wang et al., 2021), TubeViT (Piergiovanni et al., 2023), UniformerV2 (Li et al., 2022b), AIM (Yang et al., 2023), MViT (Fan et al., 2021) and MoViNet (Kondratyuk et al., 2021) respectively. All seven FMs underperform the task-specialized models on all video tasks except on Moments-in-time at the comparable model scale, indicating the lack of strong video-focused FMs. This observation is in sharp contrast to what FMs have achieved on natural language (OpenAI, 2022; Anil et al., 2023) and image understanding (Chen et al., 2022).

*CoCa performs the best among image-native FMs on the video tasks.* It actually gives rise to the highest accuracy on all datasets, with slightly inferior performance on SSv2 and Charades. This shows strong generalization capability of the CoCa model, regardless it is an image-based model with image-only pre-training data. However, in the latter session, we will reveal that under light-weight and parameter efficient adaptation scenarios, the same model may perform inferior on many video understanding tasks, especially on SSv2 (Tables 5, 6, and 7), Charades(Tables 5, 6, and 7) and ANet(Tables 5, and 6), which requires complex motion or multiple actions understanding per video. These are in contrast that CoCa achieves the best general performance in end-to-end fine-tuning (Table 4), highlighting the importance of considering adaptation methods on FMs benchmarking

### 4.2 Freezing foundation models

End-to-end finetuning is infeasible for some application scenarios due to FMs' rapidly growth in size and the consequent demands in computational resources. In the following, we evaluate frozen FMs with various adaptation methods. Tables 5, 6, and 7 are the results of adaptation with a single cross-attention layer, multiple cross-attention layers, and a low-rank adapter, respectively.

*General speaking, DINO-v2 performs the best in frozen feature pooler eval (Tables 5), CLIP performs the best with multi-head attention pooler among image-native frozen FMs (Tables 6), but CoCa catches up thanks to the low-rank adapter (Table 7).* It is worth noting that this ranking of image-native frozen FMs differs from the ranking of image-native FMs in end-to-end finetuning. It seems that DINO-v2 and CLIP's frozen features are more amendable to the video tasks than CoCa, but CoCa as a whole adapts better to video under both finetuning and the adapter. Hence, it is crucial to consider adaptation methods as an organic part of the evaluation of FMs to supply them various paths to demonstrate their capabilities.

*Video-native FMs are better than image-native FMs in understanding motion-rich SSv2 and D48, Charades that contain multiple actions per video, and ANet for temporal action localization.* This observation is about the same as the one under end-to-end finetuning. The image-native FMs is mainly superior on appearance-rich

Table 5: Evaluating FMs when adapted to video understanding using frozen features. Only weights in the task heads are updated using the downstream tasks' training sets.

| | VC (A) | | VC (M) | | VC (ML) | TAL | STAL | | |
|---|---|---|---|---|---|---|---|---|---|
| Model | K400 | MiT | SSv2 | D48 | Charades | ANet | AVA | AVA-K | AVG |
| CoCa | 73.1 | 32.0 | 41.5 | 34.1 | 8.8 | 33.0 | 23.3 | 24.7 | 31.2 |
| CLIP | **75.2** | **32.6** | 41.0 | 44.1 | 11.2 | 32.7 | 21.1 | 25.9 | 32.8 |
| FLAVA | 71.3 | 29.7 | 40.6 | 45.9 | 12.6 | 32.2 | 18.8 | 21.5 | 31.7 |
| DINO-v2 | 72.3 | 29.0 | 40.0 | 40.4 | 25.8 | 32.6 | **24.6** | **26.9** | 35.0 |
| VideoMAE | 65.1 | 23.0 | 53.9 | **59.5** | 11.3 | 33.0 | 16.0 | 19.9 | 32.6 |
| InternVideo | 69.3 | 26.3 | **58.2** | 55.6 | 13.0 | 33.3 | 13.4 | 15.7 | 33.1 |
| VATT | 75.1 | 32.1 | 57.8 | 49.7 | **33.3** | **35.3** | 20.3 | 22.2 | 39.1 |

Table 6: Evaluating FMs when adapted to video understanding using multi-layer attention pooler (MLAP), which takes multiple frozen features from an FM as inputs and map them hierarchically for the final task prediction. Only the multi-layer attention pooling layers are updated using the downstream tasks' training sets.

| | VC (A) | | VC (M) | | VC (ML) | TAL | STAL | | |
|---|---|---|---|---|---|---|---|---|---|
| Model | K400 | MiT | SSv2 | D48 | Charades | ANet | AVA | AVA-K | AVG |
| CoCa | 74.2 | 37.2 | 45.9 | 48.4 | 19.6 | 33.3 | 24.4 | 27.0 | 36.3 |
| CLIP | **77.1** | **39.0** | 50.1 | 55.8 | 41.5 | 33.9 | **27.7** | **29.6** | 43.3 |
| FLAVA | 71.5 | 34.5 | 43.1 | 58.5 | 38.2 | 32.4 | 21.3 | 23.2 | 39.3 |
| DINO-v2 | 75.4 | 36.0 | 46.3 | 51.9 | 47.8 | 33.6 | 25.4 | 27.0 | 42.5 |
| VideoMAE | 71.7 | 32.2 | 57.4 | 69.6 | 35.9 | 33.4 | 19.6 | 22.1 | 40.9 |
| InternVideo | 73.7 | 34.7 | **60.3** | **71.9** | 40.5 | 33.6 | 15.9 | 17.7 | 42.2 |
| VATT | 75.1 | 35.6 | 58.7 | 60.1 | **58.2** | **35.0** | 22.9 | 24.1 | 46.3 |

Table 7: The low-rank adapter results of FMs for video understanding. We only update the weights of the adapter and task head while keeping the original FMs' weights frozen.

| | VC (A) | | VC (M) | | VC (ML) | TAL | STAL | | |
|---|---|---|---|---|---|---|---|---|---|
| Model | K400 | MiT | SSv2 | D48 | Charades | ANet | AVA | AVA-K | AVG |
| CoCa | **80.9** | **41.4** | 56.1 | 67.1 | 45.8 | – | **26.6** | **28.7** | 49.0 |
| CLIP | 80.2 | 39.7 | 56.0 | **77.2** | 44.2 | – | 24.5 | 28.0 | 49.3 |
| FLAVA | 74.7 | 34.1 | 52.1 | 68.4 | 40.8 | – | 17.9 | 23.8 | 44.1 |
| DINO-v2 | 77.7 | 36.1 | 59.0 | 76.6 | 38.0 | – | 22.5 | 27.9 | 47.0 |
| VideoMAE | 73.6 | 30.6 | 61.4 | 76.0 | 43.0 | – | 16.6 | 23.3 | 45.9 |
| InternVideo | 75.5 | 31.3 | **63.9** | 73.6 | 46.2 | – | 19.2 | 25.5 | 47.7 |
| VATT | 75.0 | 36.5 | 63.5 | 68.9 | **53.5** | – | 22.3 | 25.8 | 49.9 |

video datasets, where high-quality spatial perceptual features are the key. We conjecture that the vast image data empowering image-native FMs is more diverse in appearance than videos used to pretrain video-native FMs.

*Given frozen FMs, the low-rank adapter outperforms cross-attention layers, and multiple layers of cross-attention is better than a single cross-attention layer.* Many works (Caron et al., 2021; He et al., 2022) have shown features from different layers of a vision transformer have different attention maps. Hence, it is potentially beneficial to have an adaptation method to leverage multiple layers of a frozen FM. Table 6 reports the results with four cross-attention layers, whose average score per model (across different columns) is higher than that with a single cross-attention layer (Table 5) by 18% to 40%. The low-rank adapter (Table 7) further improves upon the cross-attention results partially because it explores all layers of a frozen FM.

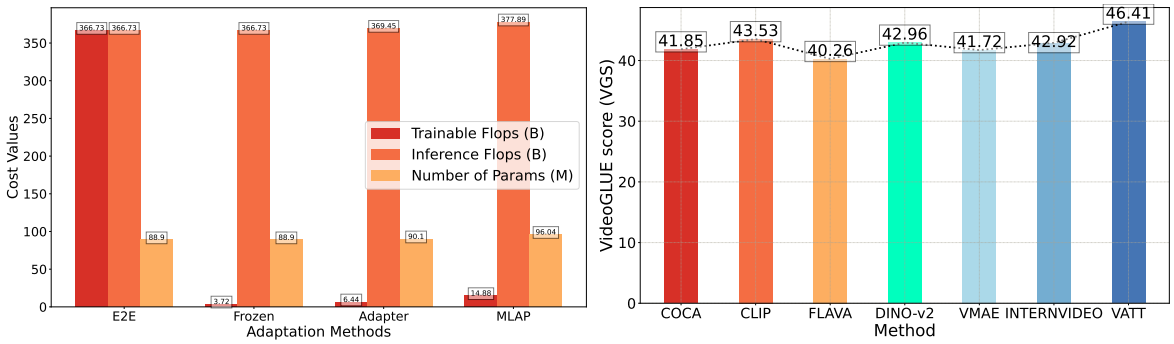


Figure 3: On the left, we measures FMs' training, inference and storage cost in trainable FLOPs, inference FLOPs and number of parameters respectively. On the right, we report VideoGLUE Score, which considers a FM's video understanding capability amortized by the developmental costs and adaptation methods.

*On average, image-native FMs outperform video-native FMs under end-to-end finetuning and the adapter, but it becomes the inverse in the other two adaptation methods.* The adapter experiment paired with end-to-end finetuning experiment reveal the fact that existing image-based FMs could be more easily adapted to video tasks when we could adjust the feature space of FMs, possibly caused by the large-scale higher quality image(-text) pretraining datasets. On the other hand, frozen feature experiments discussed above present us the inverse picture where video-based FM performs better. The seemingly paradox encourages more future research on bridging the gap on video-based pretraining with high-quality data, more effective modeling and better design on video benchmarks.

### 4.3 Profiling foundation models for video understanding

In this section, we consolidate our studies of the FMs with different adaptation methods and video tasks, focusing on their overall efficacy and efficiency. Specifically, we use trainable FLOPs, inference FLOPs, and the number of parameters to approximately represent the training, inference, and storage costs of an FM. Since all FMs in our evaluation are ViT-B and we align the same number of input tokens for each task. The models have almost the same cost in one adaptation method. The left of Figure 3 shows the cost values for each adaptation method. Note that an FM with LoRA adaptor tuning could have high inference cost despite lower training/adaptation costs than end-to-end fine-tuning. While the figure provides a holistic view of an FM from multiple dimensions, one might be interested in a ranking among the FMs in terms of their video understanding capabilities. To this end, we summarize the multi-dimensional comparisons across different datasets, adaptation methods, and costs using a simplified scalar measure, termed VideoGLUE Score (VGS), to probe an FM's general video understanding capability.

We use the cost values to normalize an adapted FM's average score $s$ over all tasks. Formally, denoting by $\mathcal{S}_i$ an FM's average score over our video tasks under the $i$-th adaptation method and by $C_i^k$ the corresponding cost value under the $k$-th developmental scenario, we calculate the FM's $VGS^k$ by

$$VGS^k = \sum_{i=1}^{N} w_i^k \mathcal{S}_i, \text{ where } w_i^k = \frac{\mathcal{A}_i^k}{\sum_{j=1}^{N} \mathcal{A}_j^k} \text{ and } \mathcal{A}_i^k = \frac{1}{\log_{10} C_i^k}, \tag{3}$$

where $N = 4$ is the number of adaptation methods, and $w_i \in [0, 1]$ weighs score $\mathcal{S}_i$ according to the cost $C_i^k$. The final VGS is the arithmetic average on $\{\text{VGS}^k\}$, where $k = 1, 2, 3$ corresponding to training, inference, and storage, respectively.

On the right panel of Figure 3, we plot each FM's VideoGLUE Score. We notice the average VGS for video-native and image-native FMs on our video understanding tasks are 43.68 vs. 42.15 respectively. Zooming in to the individual FMs, we find that VATT, a video-native FM, is at the first place with VGS 46.41, followed by the image-native CLIP with VGS 43.53. This suggests that in-domain pretraining yields overall the best

adaptation capability to video tasks, and image-native FMs could also achieve competitive results on many but not all video understanding tasks.

## 5 Limitations

VideoGLUE serves as a comprehensive benchmark for studying and probing various video understanding capabilities of foundation models. The current task portfolio includes various unimodal action understanding tasks. We believe the scope of this work could be further extended as there are many other important video tasks not covered here, e.g. object or point-level tracking, long-term memory and forecasting. Moreover, our benchmark could be strengthened by adding multimodal tasks like video captioning and question answering, given the rise of general Vision Language Models (VLM). We chose three representative FM adaptation methods and used them to provide as uniform experiment protocols for different FMs as possible. However, some of our observations could be flipped with the evolution of FMs development and adaptation methods, which are an active research area. We proposed a scalar score, VideoGLUE Score (VGS), to capture the efficacy and efficiency of an FM on video understanding. However, VGS might be dominated by one or a few datasets — when it becomes a serious issue, we should probably improve the score and/or retire the datasets from future versions of VideoGLUE. Indeed, VGS is not a perfect score that covers all aspects of FMs in a comprehensive manner. For example, it does not account for an FM's model size, model architecture, etc. We hope future research will lead to new metrics to complement VGS and a more comprehensive evaluation of FMs for visual tasks.

## 6 Conclusion

In this report, we study four image-based and three video-based foundation models and their adaptation capability on general video understanding tasks. Experiments are conducted on three hallmark video tasks, eight diverse datasets with four distinct adaption methods. Our study shows existing image-based FMs performs well on some appearance-rich video datasets, while video-based FMs tend to achieve better on motional and temporal reasoning. Four studied adaption methods curve different landscape, revealing the critical role of considering adaption methods as an organic part of evaluating FMs. Finally, we propose one single metric VGS to represent the video task adaptation efficiency of FMs. We hope our research provides useful resources for evaluating and analyzing video foundation models, and address the current gap in foundation model evaluation within the video domain.

# Supplementary Materials

We first discuss the limitations, ethcial concerns and broader impact of this work (Section **??**). We detail the datasets (Section B), models (Section C), and training setups (Section D) in the supplementary materials to improve this work's reproducibility. Besides, Section E includes more experimental studies to strengthen the main text.

## A Ethical concern and broader impact

**Ethical concern.** We evaluate FMs on three video tasks, eight datasets in total. We select the tasks and datasets based on their popularity and representativeness. Although carefully designed, our benchmark inevitably inherited some ethical concerns from those datasets. For instance, many of the datasets are curated by crawling videos from the Internet, which do not proportionately represent the experiences of the global population and can potentially lead to biased evaluations of FMs. Moreover, the video datasets involve human daily activities, leading to privacy concerns about the human actors in the videos. How to evaluate FMs for video understanding in a fair and privacy-preserving manner could be an important direction for future research.

**Broader impact.** Our research reveals the need and tremendous opportunities to research video-first FMs by improving pretraining video data and methodologies. Our studies on different adaptation methods on versatile tasks confirms that both tasks and adaptation methods matter when it comes to the evaluation of FMs, shedding light on the already vibrant area of FM adaptations. Finally, we hope our research could inspire research on foundation models development and video understanding in general, along with their applications in the real world.

## B Video understanding datasets

### B.1 Appearance-focused action recognition

Video classification is a task of classifying videos into pre-defined labels, with the major focus on human actions.

Kinetics400 (Kay et al., 2017) (K400) is a large-scale, high-quality video dataset widely used as a standard video classification benchmark. It contains more than 250k video clips with annotations of 400 human daily actions. The actions are human focused and cover a broad range of classes including human-human interactions and human-object interactions. Although the video clips span 10 seconds on average, many studies (Sevilla-Lara et al., 2021; Wang et al., 2018b) have pointed out the task could be easily solved on the Kinetics datasets by inferring from the static objects appeared or background environment — motion information is less important than the visual appearance. Hence, we categorize Kinetics400 as an appearance-focused action classification dataset.

Moments-in-Time (Monfort et al., 2019) (MiT) is a large-scale video event classification dataset, with one million human annotated short video clips (around 3 seconds each). The temporal span corresponds to the averaged duration of human working memory and is a temporal envelope holding meaningful actions between people, objects, and phenomena. Videos in MiT are annotated with 339 most used verbs in the English vocabulary.

### B.2 Motion-focused action recognition

Videos contain much more commonsense knowledge than still images do, such as an object's motion patterns and the causal consequences of an action, just to name a few. However, appearance-based benchmarks do not evaluate a model's understanding of such commonsense knowledge, complex scenes, and situations. In observance of this, some video datasets have been proposed and studied in recent years with the focus on motions and common-sensing reasoning that are prosperous in video data.

Something-something v2 (Goyal et al., 2017) (SSv2) is a collection of around 200k videos of human performing pre-defined, basic actions with everyday objects. There are 174 unique labels in total depicting atomic hand manipulations, like putting something into something, turning something upside down or covering something with something. This dataset benchmarks a model's fine-grained understanding capability of object motions and scene changes by making the label space atomic-action-focused and background-invariant.

Diving48 (Li et al., 2018) (D48) is introduced to evaluate a model's dynamic reasoning capability. The video clips in this dataset are obtained by segmenting online videos of major diving competitions. In total, there are around 18k videos annotated with 48 classes. Because of its standardization, the diving scenario is purposefully chosen to avoid the scene, object, and person biases.

### B.3 Multi-label daily action classification

Most of current action classification datasets involve video clips with a clean snapshot of a single action. In contrast, humans perform daily complex activities step-by-step, simultaneously, or in an interleaving manner. Towards more comprehensive human daily activity reasoning, Charades (Sigurdsson et al., 2016) is introduced. Different from web-collected datasets whose contents are more structured, Charades is collected by crowd-sourcing from hundreds of actors recording their videos in their own homes, acting out casual everyday activities. Charades brings in more diversity into the video classification task due to its close-to-daily-life setting. Its videos are 30 seconds long on average and have multi-label annotations testing models' understanding of complex daily activities with multiple steps. Charades provides 110k videos with 157 action classes for training and evaluation.

### B.4 Temporal action localization

Natural long videos contain scene changes and semantic shifts, while most of the existing video benchmarks formulate problems to focus on trimmed video clips. Such a gap introduces evaluation bias as clip-level benchmarks could not reflect a model's temporal feature discriminativeness, which is of key importance to solve long-form video understanding tasks. To comprehend the study on foundation models' video capabilities, we include the temporal action localization (TAL) task in our evaluation. The task of TAL is to predict not only the action labels but also each action instance's temporal boundary in untrimmed videos. We adopt ActivityNet v1.3 (Fabian Caba Heilbron & Niebles, 2015) as the dataset for the TAL task, which contains $10,002$ untrimmed videos in training and $4,985$ in validation. The video length in this dataset is between 5-10 minutes. In total, there are 200 types of activities annotated.

### B.5 Spatiotemporal action localization

Spatiotemporal Action Localization (STAL) is a person-centric task that asks a system to localize actors and predict their atomic actions (Barker & Wright, 1955; Gu et al., 2018) in a transitory duration.

In AVA (Gu et al., 2018), 15 minutes long movie clips are densely annotated at 1Hz. In the key frames, every person is localized using a bounding box and labels corresponding to actions being performed by the actor. The label vocabulary consists of 80 different atomic visual actions. There are 430 different movies in total.

AVA-Kinetics (Li et al., 2020) follows the same labeling protocol as AVA, while its data source comes from the Kinetics700 (Kay et al., 2017) video pool. The dataset contains over 230k clips annotated with the 80 AVA action classes for each of the humans in key frames.

## C Model details

### C.1 Task head architectures

In Figure 4, we plot the task heads used in our video classification and spatiotemporal action localization experiments, namely, the simple pooler head and multi-layer attention pooling head. For temporal localization, please refer to (Xu et al., 2020) for the task head's detailed architecture.

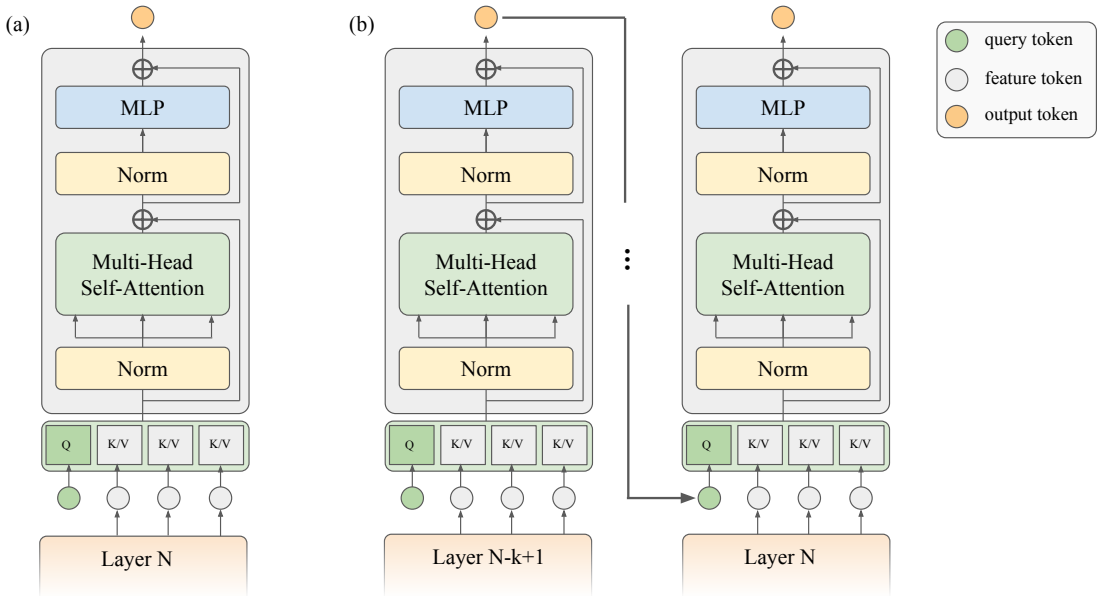


Figure 4: (a) Single-layer pooler head and (b) multi-layer attention pooling head for video classification and spatiotemporal action localization.

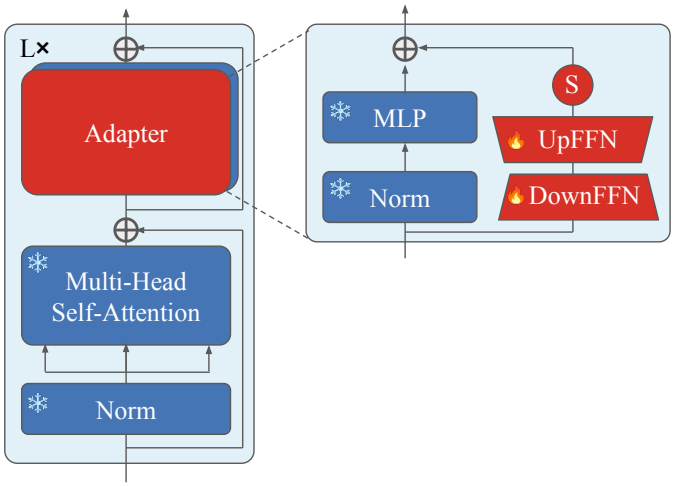


Figure 5: The adapter used in vision transformer. In the adapter layer, only the down-sample layer, up-sample layer, and the scaling factor are tunable. Between the down-sample layer and up-sample layer, an activation function is applied, which in our case is ReLU.

Figure 5 illustrates the encoder adapter layer's architecture. In the the adapter layer, only the down-sample layer, up-sample layer, and the scaling factor are tunable.

### C.2 Image-to-video adaptation

Adapting image backbones to video tasks requires us to fuse the image embeddings at some point in the network and also introduce additional temporal information.

We consider two choices, early-fusion and late-fusion, and ablate them in the frozen feature setting in Table 8. In both early-fusion and late-fusion, we first apply the projection layer on each frame independently to embed pixel patches into embedding tokens. We then average-pool the embedding tokens from nearby frames to

Table 8: Early vs. late fusion on image-native FMs. In this experiment, the frozen feature with a single-layer pooler head is used.

| | K400 | | SSv2 | |
|---|---|---|---|---|
| Method | Early | Late | Early | Late |
| CoCa | 72.7 | 61.4 | 41.5 | 33.3 |
| CLIP | 70.5 | 75.2 | 38.1 | 41.0 |
| FLAVA | 67.9 | 71.3 | 40.4 | 40.6 |

Table 9: Ablation study on the temporal positional embedding for image-to-video adaption. We choose FLAVA (Singh et al., 2022) with the frozen feature setting in this experiment.

| Temporal Positional | VC (A) | | VC (M) | | VC (ML) |
|---|---|---|---|---|---|
| Embedding | K400 | MiT | D48 | SSv2 | Charades |
| ✗ | 71.3 | 29.7 | 41.6 | 30.3 | 10.7 |
| ✓ | 71.3 | 29.7 | 45.9 | 40.6 | 12.6 |

reduce the sequence length to $n \times h \times w$. In the early-fusion setting, we pass all tokens *together* to the image backbone to extract video features. In late-fusion, we pass each set of $h \times w$ tokens *independently* to the image backbone. Empirically, we find that the FLAVA (Singh et al., 2022) and CLIP (Radford et al., 2021) models do better with late-fusion while CoCa (Yu et al., 2022) does better with early-fusion.

Furthermore, we ablate the importance of temporal information using the frozen-features from FLAVA (Singh et al., 2022). In Table 9, we find that adding temporal positional embedding to the input is essential for D48 (Li et al., 2018), SSv2 (Goyal et al., 2017), and Charades (Sigurdsson et al., 2016) while not necessary for K400 (Kay et al., 2017) and MiT (Monfort et al., 2019). This supports our grouping that K400 and MiT are appearance-focused datasets.

Based on these findings, we use late-fusion for FLAVA (Singh et al., 2022) and CLIP (Radford et al., 2021) and early-fusion for CoCa (Yu et al., 2022). We add learnable temporal positional embeddings for all the image-native FMs.

## D Task-specific hyperparameters

In the following, we provide experiment settings and hyperparamters we used in this study. In Table 10, we list the hyperparameters we applied in the video classification task. In Table 11, we present the hyperparameters we used on spatiotemporal action localization. In Table 12, we present the hyperparameters we used on temporal action localization task.

We performed a greedy search on the learning rate and weight decay in all our experiments while keeping most other hyperparameters (e.g., data augmentation magnitude, dropout rate, drop path rate, etc.) consistent across different models and datasets. Specifically, we start with learning rate 1e-4 and weight decay 1e-5 and uniformly sample learning rates and weight decay factors with a rate of 5 and 10, respectively, centered around the starting points. After the first round, we pick the best-identified learning rate and weight decay factor as the new starting point and conduct another round of sampling with a rate of 2. We repeat another two to three rounds of hyperparameter search (with a rate of 2) until the model's performance converges. This process is a trade-off between computation costs and thoroughly examining an FM's performance under each experiment setup. The search ranges for the learning rate and weight decay are [4e-5, 2.5e-3] and [1e-6, 1e-4], respectively. We found that the learning rate is the most crucial factor when adapting an FM to downstream video understanding tasks.

Table 10: Experimental configurations for video classification tasks. We let learning rate and weight decay to be tunable per model to allow some flexibility for task adaptations.

| **Config** | **Kinetics400** | **Sth-sth v2** | **MiT** | **Diving48** | **Charades** |
|---|---|---|---|---|---|
| batch size | 256 | 256 | 256 | 256 | 256 |
| training epochs | 150 | 50 | 50 | 100 | 50 |
| ViT sequence length | $8 \times 14 \times 14$ | $8 \times 14 \times 14$ | $8 \times 14 \times 14$ | $8 \times 14 \times 14$ | $8 \times 14 \times 14$ |
| **optimization** | | | | | |
| optimizer | AdamW | AdamW | AdamW | AdamW | AdamW |
| optimizer momentum | 0.9 | 0.9 | 0.9 | 0.9 | 0.9 |
| learning rate schedule | cosine decay | cosine decay | cosine decay | cosine decay | cosine decay |
| warmup ratio | 5% | 5% | 5% | 5% | 5% |
| **data augmentations** | | | | | |
| random horizontal flip | true | false | true | true | false |
| aspect ratio | (0.5, 2.0) | (0.5, 2.0) | (0.5, 2.0) | (0.5, 2.0) | (0.5, 2.0) |
| area ratio | (0.3, 1.0) | (0.3, 1.0) | (0.3, 1.0) | (0.3, 1.0) | (0.3, 1.0) |
| RandAug | (9, 0.5) | (9, 0.5) | - | - | - |
| MixUp | 0.8 | 0.8 | - | - | - |
| CutMix | 1.0 | 1.0 | - | - | - |
| **evaluation** | | | | | |
| multi-clips | 4 | 1 | 4 | 4 | 4 |
| multi-views | 3 | 3 | 3 | 3 | 3 |
| segment-based sample | false | true | false | false | false |

Table 11: Experimental configurations for spatiotemporal action localization.

| **Config** | **AVA v2.2** | **AVA-Kinetics** |
|---|---|---|
| batch size | 256 | 256 |
| training epochs | 50 | 50 |
| ViT sequence length | $8 \times 16 \times 16$ | $8 \times 16 \times 16$ |
| **optimization** | | |
| optimizer | AdamW | AdamW |
| optimizer momentum | 0.9 | 0.9 |
| layer decay | 0.75 | 0.75 |
| learning rate schedule | cosine decay | cosine decay |
| warmup ratio | 5% | 5% |
| **data augmentations** | | |
| random horizontal flip | true | true |
| random scale | (0.5, 2.0) | (0.5, 2.0) |
| random color augmentation | true | true |

# E More studies

## E.1 Large model adaptations

For the completeness of this report and reader's reference, in Table 13 we report experimental results under our settings with large FMs under the frozen backbone with one pooler head setup.

Table 12: Experimental configurations for temporal action localization.

| **Config** | **ActivityNet v1.3** |
|---|---|
| batch size | 32 |
| training epochs | 10 |
| **feature extraction** | |
| fps | 15 |
| per-clip length | 16 |
| clip stride | 16 |
| **optimization** | |
| optimizer | AdamW |
| optimizer momentum | 0.9 |
| learning rate schedule | cosine decay |

Table 13: Evaluating large-scale FMs when using frozen feature with a one-layer pooler head. We report the Top-1 accuracy on K400, MiT, D48, SSv2 and MAP on Charades.

| | VC (A) | | VC (M) | | VC (ML) |
|---|---|---|---|---|---|
| Model | K400 | MiT | D48 | SSv2 | Charades |
| InternVideo-L | 78.6 | 33.7 | 69.6 | 67.4 | 20.9 |
| VideoMAE-v2-B/DL | 86.7 | 38.9 | 61.4 | 57.7 | 33.2 |
| VideoMAE-v2-g | 59.7 | 20.7 | 42.5 | 44.2 | 12.7 |
| VideoMAE-v2-g/FT | 82.1 | 35.0 | 60.5 | 56.1 | 22.4 |
| InternVideo-v2-g | 85.0 | 43.0 | 53.1 | 61.6 | 40.9 |
| VideoPrism-g | 86.6 | 44.7 | 66.1 | 67.4 | 61.0 |

VideoMAE-v2-B/DL (Wang et al., 2023) denotes the ViT-B model distilled from ViT-g on the Kinetics710 datasets[1]. VideoMAE-v2-g (Wang et al., 2023) is the model that pretrained on UnlabeledHybrid dataset, while VideoMAE-v2-g/FT (Wang et al., 2023) conducts further finetuning using supervised training on Kinetics710. InternVideo-v2-g (Wang et al., 2024) and VideoPrism-g (Zhao et al., 2024) are two video foundation models with multi-stage pre-training on curated in-house web video data. For InternVideo-v2-g, we use their stage-2 checkpoint[2]. For videoPrism-g, we use their final checkpoint.

### E.2 Sample-efficient transfer learning

A strong FM should be able to adapt to downstream tasks with a few training samples. In this section, we test the adaption ability of FMs in a sample-efficient transfer learning setting. Particularly, we freeze backbones and train a pooler head to adapt the FMs on K400 and SSv2. For either dataset, we sample 1% and 10% data from the training set uniformly for training and evaluate on the full evaluation dataset.

We show our experimental results in Table 14. To better understand the data efficiency, we also show the relative Top-1 accuracy for each model (shown in the bracket), which is defined as the ratio between accuracy with fewer training examples and the accuracy achieved using all the training data. A higher relative Top-1 accuracy means the performance of the model is closer to its "full" capacity under the sample-efficient setting. We notice that the best performed model on each dataset in fully fine-tuned model also performs best in the few-shot setting. Especially, CLIP (Radford et al., 2021) achieves 46.2% and 83.6% relative Top-1 accuracy on K400 using only 1% and 10% of the training data, respectively. On SSv2, InternVideo (Wang et al., 2022b) achieves 33.6% and 70.6% relative Top-1 accuracy with only 1% and 10% of the training data.

[1] https://github.com/OpenGVLab/VideoMAEv2/blob/master/docs/MODEL_ZOO.md
[2] https://github.com/OpenGVLab/InternVideo/blob/main/InternVideo2/multi_modality/MODEL_ZOO.md

Table 14: Benchmark FMs adaptation on video understanding tasks under sample-efficient transfer learning. This table shows Top-1 classification accuracy and the relative accuracy (shown in the bracket). Results are achieved by using frozen features with pooler head.

| Method | K400 | | | SSv2 | | |
|---|---|---|---|---|---|---|
| | 1% | 10% | 100% | 1% | 10% | 100% |
| CoCa | 27.1(37.8%) | 48.9(67.0%) | 73.1 | 5.6(13.4%) | 20.9(50.4%) | 41.5 |
| CLIP | 36.9(46.2%) | 66.8(83.6%) | 79.0 | 8.7(19.3%) | 25.1(55.5%) | 45.3 |
| FLAVA | 14.4(20.2%) | 35.8(50.3%) | 71.3 | 7.2(17.7%) | 14.3(35.3%) | 40.6 |
| VideoMAE | 15.5(23.9%) | 32.0(49.2%) | 65.0 | 13.7(25.4%) | 30.3(56.2%) | 53.9 |
| InternVideo | 20.4(29.5%) | 50.2(72.4%) | 69.3 | 19.5(33.6%) | 41.1(70.7%) | 58.2 |
| VATT | 34.1(45.4%) | 63.7(84.8%) | 75.1 | 12.9(22.4%) | 37.6(65.0%) | 57.8 |

## References

Hassan Akbari, Liangzhe Yuan, Rui Qian, Wei-Hong Chuang, Shih-Fu Chang, Yin Cui, and Boqing Gong. Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text. *Advances in Neural Information Processing Systems*, 34:24206–24221, 2021.

Humam Alwassel, Silvio Giancola, and Bernard Ghanem. Tsp: Temporally-sensitive pretraining of video encoders for localization tasks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3173–3183, 2021.

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*, 2023.

Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*, 2021.

Roger G Barker and Herbert F Wright. Midwest and its children: The psychological ecology of an american town. *Marriage and family living*, 1955.

Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

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

Shyamal Buch, Cristobal Eyzaguirre, Adrien Gaidon, Jiajun Wu, Li Fei-Fei, and Juan Carlos Niebles. Revisiting the "Video" in Video-Language Understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9650–9660, 2021.

Xi Chen, Xiao Wang, Soravit Changpinyo, AJ Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, et al. Pali: A jointly-scaled multilingual language-image model. *arXiv preprint arXiv:2209.06794*, 2022.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022.

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

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. Unified language model pre-training for natural language understanding and generation. *Advances in neural information processing systems*, 32, 2019.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Bernard Ghanem Fabian Caba Heilbron, Victor Escorcia and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 961–970, 2015.

Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6824–6835, 2021.

Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, pp. 6202–6211, 2019.

Christoph Feichtenhofer, Haoqi Fan, Bo Xiong, Ross Girshick, and Kaiming He. A large-scale study on unsupervised spatiotemporal representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3299–3309, 2021.

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

Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense. In *Proceedings of the IEEE international conference on computer vision*, pp. 5842–5850, 2017.

Chunhui Gu, Chen Sun, David A Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, et al. Ava: A video dataset of spatio-temporally localized atomic visual actions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6047–6056, 2018.

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

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *ICLR*, 2021.

Shaohan Huang, Li Dong, Wenhui Wang, Yaru Hao, Saksham Singhal, Shuming Ma, Tengchao Lv, Lei Cui, Owais Khan Mohammed, Qiang Liu, et al. Language is not all you need: Aligning perception with language models. *arXiv preprint arXiv:2302.14045*, 2023.

Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. *Advances in neural information processing systems*, 28, 2015.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, pp. 4904–4916. PMLR, 2021.

Chen Ju, Tengda Han, Kunhao Zheng, Ya Zhang, and Weidi Xie. Prompting visual-language models for efficient video understanding. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXV*, pp. 105–124. Springer, 2022.

Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

Dan Kondratyuk, Liangzhe Yuan, Yandong Li, Li Zhang, Mingxing Tan, Matthew Brown, and Boqing Gong. Movinets: Mobile video networks for efficient video recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16020–16030, 2021.

Jie Lei, Tamara L. Berg, and Mohit Bansal. Revealing single frame bias for video-and-language learning. In *Proceedings if the 61st Annual Meeting of the Association for Computational Linguistics*, 2023.

Ang Li, Meghana Thotakuri, David A Ross, João Carreira, Alexander Vostrikov, and Andrew Zisserman. The ava-kinetics localized human actions video dataset. *arXiv preprint arXiv:2005.00214*, 2020.

Chunyuan Li, Haotian Liu, Liunian Harold Li, Pengchuan Zhang, Jyoti Aneja, Jianwei Yang, Ping Jin, Yong Jae Lee, Houdong Hu, Zicheng Liu, et al. Elevater: A benchmark and toolkit for evaluating language-augmented visual models. *arXiv preprint arXiv:2204.08790*, 2022a.

Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Limin Wang, and Yu Qiao. Uniformerv2: Spatiotemporal learning by arming image vits with video uniformer. *arXiv preprint arXiv:2211.09552*, 2022b.

Yingwei Li, Yi Li, and Nuno Vasconcelos. Resound: Towards action recognition without representation bias. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 513–528, 2018.

Ziyi Lin, Shijie Geng, Renrui Zhang, Peng Gao, Gerard de Melo, Xiaogang Wang, Jifeng Dai, Yu Qiao, and Hongsheng Li. Frozen clip models are efficient video learners. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXV*, pp. 388–404. Springer, 2022.

Xiaolong Liu, Song Bai, and Xiang Bai. An empirical study of end-to-end temporal action detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20010–20019, 2022.

Mathew Monfort, Alex Andonian, Bolei Zhou, Kandan Ramakrishnan, Sarah Adel Bargal, Tom Yan, Lisa Brown, Quanfu Fan, Dan Gutfruend, Carl Vondrick, et al. Moments in time dataset: one million videos for event understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–8, 2019. ISSN 0162-8828. doi: 10.1109/TPAMI.2019.2901464.

OpenAI. Gpt-4 technical report. *https://cdn.openai.com/papers/gpt-4.pdf*, 2022.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. In *TMLR*, 2023.

Viorica Patraucean, Lucas Smaira, Ankush Gupta, Adria Recasens, Larisa Markeeva, Dylan Banarse, Skanda Koppula, Mateusz Malinowski, Yi Yang, Carl Doersch, et al. Perception test: A diagnostic benchmark for multimodal video models. *Advances in Neural Information Processing Systems*, 36, 2024.

AJ Piergiovanni, Weicheng Kuo, and Anelia Angelova. Rethinking video vits: Sparse video tubes for joint image and video learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2214–2224, 2023.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763. PMLR, 2021.

Jathushan Rajasegaran, Georgios Pavlakos, Angjoo Kanazawa, Christoph Feichtenhofer, and Jitendra Malik. On the benefits of 3d pose and tracking for human action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 640–649, 2023.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pp. 8821–8831. PMLR, 2021.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

Adam Roberts, Hyung Won Chung, Anselm Levskaya, Gaurav Mishra, James Bradbury, Daniel Andor, Sharan Narang, Brian Lester, Colin Gaffney, Afroz Mohiuddin, et al. Scaling up models and data with `t5x` and `seqio`. *arXiv preprint arXiv:2203.17189*, 2022.

Laura Sevilla-Lara, Shengxin Zha, Zhicheng Yan, Vedanuj Goswami, Matt Feiszli, and Lorenzo Torresani. Only time can tell: Discovering temporal data for temporal modeling. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 535–544, 2021.

Gunnar A Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pp. 510–526. Springer, 2016.

Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. Flava: A foundational language and vision alignment model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15638–15650, 2022.

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

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*, 2018a.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*, 32, 2019.

Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks for action recognition in videos. *IEEE transactions on pattern analysis and machine intelligence*, 41(11): 2740–2755, 2018b.

Limin Wang, Bingkun Huang, Zhiyu Zhao, Zhan Tong, Yinan He, Yi Wang, Yali Wang, and Yu Qiao. Videomae v2: Scaling video masked autoencoders with dual masking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14549–14560, 2023.

Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, et al. Image as a foreign language: Beit pretraining for all vision and vision-language tasks. *arXiv preprint arXiv:2208.10442*, 2022a.

Xiang Wang, Zhiwu Qing, Ziyuan Huang, Yutong Feng, Shiwei Zhang, Jianwen Jiang, Mingqian Tang, Changxin Gao, and Nong Sang. Proposal relation network for temporal action detection. *arXiv preprint arXiv:2106.11812*, 2021.

Yi Wang, Kunchang Li, Yizhuo Li, Yinan He, Bingkun Huang, Zhiyu Zhao, Hongjie Zhang, Jilan Xu, Yi Liu, Zun Wang, et al. Internvideo: General video foundation models via generative and discriminative learning. *arXiv preprint arXiv:2212.03191*, 2022b.

Yi Wang, Kunchang Li, Xinhao Li, Jiashuo Yu, Yinan He, Guo Chen, Baoqi Pei, Rongkun Zheng, Jilan Xu, Zun Wang, et al. Internvideo2: Scaling video foundation models for multimodal video understanding. *arXiv preprint arXiv:2403.15377*, 2024.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*, 2022.

Mengmeng Xu, Chen Zhao, David S Rojas, Ali Thabet, and Bernard Ghanem. G-tad: Sub-graph localization for temporal action detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10156–10165, 2020.

Shen Yan, Tao Zhu, Zirui Wang, Yuan Cao, Mi Zhang, Soham Ghosh, Yonghui Wu, and Jiahui Yu. Videococa: Video-text modeling with zero-shot transfer from contrastive captioners. *arXiv preprint arXiv:2212.04979*, 2022.

Taojiannan Yang, Yi Zhu, Yusheng Xie, Aston Zhang, Chen Chen, and Mu Li. Aim: Adapting image models for efficient video action recognition. *arXiv preprint arXiv:2302.03024*, 2023.

Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022.

Chen-Lin Zhang, Jianxin Wu, and Yin Li. Actionformer: Localizing moments of actions with transformers. In *European Conference on Computer Vision*, pp. 492–510. Springer, 2022.

Long Zhao, Nitesh B Gundavarapu, Liangzhe Yuan, Hao Zhou, Shen Yan, Jennifer J Sun, Luke Friedman, Rui Qian, Tobias Weyand, Yue Zhao, et al. Videoprism: A foundational visual encoder for video understanding. *arXiv preprint arXiv:2402.13217*, 2024.