VLG: General Video Recognition with Web Textual Knowledge

Anonymous authors
Paper under double-blind review

Abstract

Video recognition in an open world is quite challenging, as we need to handle different settings such as close-set, long-tail, few-shot and open-set. By leveraging semantic knowledge from noisy text descriptions crawled from the Internet, we focus on the general video recognition (GVR) problem of solving different recognition tasks within a unified framework. The contribution of this paper is twofold. First, we build a comprehensive video recognition benchmark of Kinetics-GVR, including four sub-task datasets to cover the mentioned settings. To facilitate the research of GVR, we propose to utilize external textual knowledge from the Internet and provide multi-source text descriptions for all action classes. Second, inspired by the flexibility of language representation, we present a unified visual-linguistic framework (VLG) to solve the problem of GVR by devising an effective two-stage training paradigm. Our VLG is first pre-trained on video and language datasets to learn a shared feature space, and then devises a flexible bi-modal attention head to collaborate high-level semantic concepts under different settings. Extensive results show that our VLG obtains the state-of-the-art performance under four settings. The superior performance demonstrates the effectiveness and generalization ability of our proposed VLG framework. We hope our work makes a step towards the general video recognition and could serve as a baseline for future research.

1 Introduction

Similar to image classification, the existing video recognition tasks are roughly grouped into four settings: close-set (Kay et al., 2017), long-tail (Zhang et al., 2021), few-shot (Zhu & Yang, 2018) and open-set (Acsintoae et al., 2021), to mimic the realistic scenarios in practice. With multiple video benchmarks (Kay et al., 2017; Soomro et al., 2012; Goyal et al., 2017), a number of works (Wang et al., 2016; Zhang et al., 2021; Zhu & Yang, 2018; 2020; Bao et al., 2021) have been developed to study video recognition in these diverse scenarios.

Though various video benchmarks and frameworks have been established in the last few years, there still remain two problems: 1) video datasets in different settings are normally collected from various data sources and naturally introduce domain bias. They are not suitable for studying general video representation. It is also inefficient for data organization and storage to use multiple benchmarks separately in different settings; 2) most works (Feichtenhofer et al., 2016; Carreira & Zisserman, 2017; Tran et al., 2018; Kumar Dwivedi et al., 2019; Shu et al., 2018) focus on addressing individual settings separately with different frameworks. These separate investigations would ignore the potential sharing of knowledge among different settings. These problems severely impede the advance in video recognition as well as its application in the real world. Accordingly, we aim to present a single video benchmark covering all these settings, and propose a simple framework to handle these different sub-problems under a unified perspective.

To address the first problem: we build a comprehensive video benchmark to dig into the **General Video Recognition** (GVR) problem, namely, covering video recognition under the following four settings. As shown in the left of Figure 1, this benchmark for GVR can cover a wide range of settings including close-set, long-tail, few-shot and open-set. Specifically, we curate a general video recognition benchmark **Kinetics-GVR** from the Kinetics-400 dataset (Carreira & Zisserman, 2017), with four sub-settings: Kinetics-Close, Kinetics-LT, Kinetics-Fewshot and Kinetics-Open, to mimic the video distribution of different scenarios in real-world applications. Our Kinetics-GVR aims to

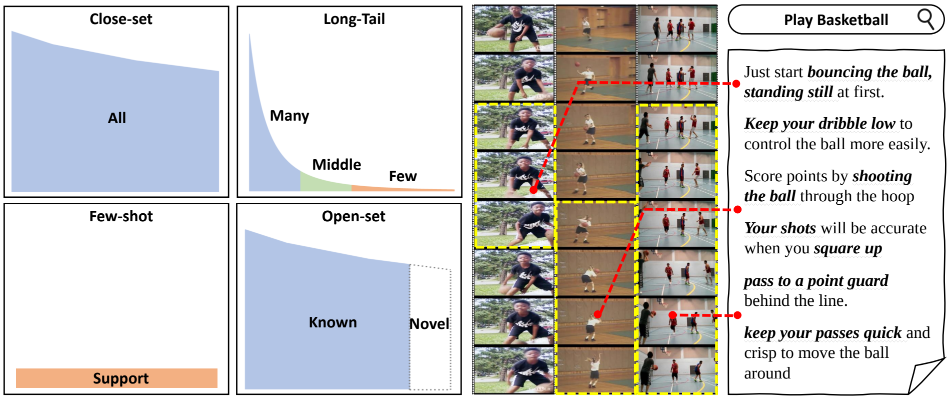

Close-set
All
Long-Tail
Many
Middle
Few
Few-shot
Support
Open-set
Known
Novel
Play Basketball
Just start *bouncing the ball, standing still* at first.
Keep your dribble low to control the ball more easily.
Score points by *shooting the ball* through the hoop
Your shots will be accurate when you *square up*
pass to a point guard behind the line.
keep your passes quick and crisp to move the ball around

Figure 1: **Video label distribution of different scenarios and different modalities.** As shown in the left, videos in GVR tasks have arbitrary distributions similar to natural data, such as close-set, long-tail, few-shot and open-set. Most works only focus on coping with one aspect of them, while our method can use a unified framework to address the GVR task by combining the advantages of video and text modalities. The right part of the figure provides intuitive explanations for the correspondence between the videos and text modalities.

provide a solid benchmark to verify the performance of video recognition models under different video and label distributions.

Since some works (Radford et al., 2021; Jia et al., 2021; Li et al., 2022; Yuan et al., 2021) have shown the efficacy of using natural language to supervise the visual representation learning, we intend to draw some extra knowledge (*i.e.*, web text information) into our benchmark to facilitate the development of GVR. The extra web knowledge is expected to provide new cues for GVR. However, obtaining the paired text data for each video is prohibitively expensive. As shown in the right of Figure 1, we observe that there are some connections between the video and text descriptions of its corresponding category. Specifically, the text descriptions for a specific video category exhibit some high-level semantic concepts to represent the static characteristics (*e.g.*, scene) in space and dynamics (*e.g.*, the steps to shooting) in time. In this sense, we hope that the text descriptions of video categories could also provide useful clues to learn a more general representation for GVR under different settings. As a result, we also provide abundant text descriptions per-category in our benchmark to facilitate the research of GVR by crawling from the Internet.

To address the second problem: we develop a unified framework to address general video recognition. Instead of dealing with each setting of video recognition with different frameworks, the unified framework would greatly reduce the work of hand-crafted design specific to each setting, and potentially increase its generalization ability due to the comprehensive consideration of all settings.

We find some recent visual-linguistic representation works, *e.g.* CLIP (Radford et al., 2021) and ALIGN (Jia et al., 2021), can learn transferable visual models from natural language supervision, and show the promising performance on image recognition under different settings. However, there is still a lack of work to bridge the gap between video and text for general recognition under different scenarios. Accordingly, we develop a video-language framework for general video recognition, termed as **VLG**. VLG could benefit from the visual-linguistic models pretrained on the large-scale image-text pairs (*e.g.*, CLIP (Radford et al., 2021)), and connect video and text through customized temporal modeling. Our VLG leverages the rich semantic information of web text descriptions to guide the spatio-temporal feature learning. Specifically, our method primarily contains four components: 1) The frame encoder to learn the visual representation for each frame. 2) The temporal module to model temporal features across frames for video domain adaption; 3) The language encoder to learn the textual representation for each sentence of category description. 4) The bi-modal attention head to perform general video recognition under different settings. As text descriptions are directly collected from the Internet, they may include some noisy information. Thus we design a two-stage procedure to train our VLG: **Stage I** is to perform video-language pretraining, adapting the encoders from the image domain to the video domain to learn a visual-linguistic representation. **Stage II** filters out noisy texts and train the bi-modal attention module to produce our final prediction. As demonstrated in experiments, our proposed VLG can effectively handle GVR under different settings of close-set, long-tail, few-shot, and open-set.

In summary, we make the following major contributions. 1) We formulate the task of general video recognition (GVR) and establish a comprehensive benchmark to fairly test the performance of video recognition models under different data distributions. The benchmark for general video recognition based on Kinetics400 (Kay et al., 2017) comprises close-set, long-tail, few-shot and open-set, which shows different data distribution in practice. 2) To facilitate the research of GVR, we elaborately collect abundant text descriptions for each category. These extra textual knowledge exhibits more rich and high-level semantic concepts to represent the characteristics both in time and space, and contributes to the development of GVR. 3) We develop a unified video-language framework for general video recognition (VLG), which leverages the extensive web textual knowledge to effectively handle GVR under our customized two-stage learning strategy 4) Extensive experiments demonstrate the effectiveness of our VLG on the Kinetics-GVR for general video recognition under four settings.

2 Related Work

Video Representation. Video recognition has made rapid progress from the early hand-craft descriptors (Klaser et al., 2008; Wang et al., 2013) to current deep networks. Deep neural networks can capture more general spatio-temporal representation from early two-stream networks, 3D-CNNs, and light-weight temporal modules to current transformer-based networks. Two-stream networks (Simonyan & Zisserman, 2014; Wang et al., 2016) used two inputs of RGB and optical flow to separately model appearance and motion information in videos with a late fusion. 3D-CNNs (Tran et al., 2015; Carreira & Zisserman, 2017) proposed 3D convolution and pooling to model space and time jointly. Light-weight temporal modules (Xie et al., 2018; Lin et al., 2019; Li et al., 2020; Liu et al., 2021c) were designed as powerful plugins to achieve the trade-off between efficacy and efficiency. Recently, several works (Bertasius et al., 2021; Arnab et al., 2021) try to employ and adapt strong vision transformers to encode the spatial and temporal features jointly. The aforementioned methods mostly focus on addressing the video recognition problem only using visual modality in a supervised way, while ignoring the potentiality of natural language.

Visual-Textual Learning. Visual-Textual Pretraining has made great progress on several downstream vision tasks. Li & Wang (2020) learned powerful video representation from a large-scale video-text pairs, with a contrastive learning method of CPD. Miech et al. (2020) proposed a new learning loss to address misalignments inherent in narrated videos. Akbari et al. (2021) proposed a framework for learning multimodal representations for unlabeled data. Recently, CLIP (Radford et al., 2021) and ALIGN (Jia et al., 2021) adopted simple noisy contrastive learning to obtain visual-linguistic representation from large-scale image-text web data. Some works also focus on a specific type of downstream tasks by adapting the pretrained image-text representation, *e.g.* video-text VQA (Kant et al., 2020; Singh et al., 2019), video-text retrieval (Dong et al., 2021; Liu et al., 2021a). Wang et al. (2021b); Ju et al. (2021) adopted prompt engineering to reformulate their tasks into the same format as the pretraining objectives. However, these methods cannot excavate the values of the noisy text descriptions data from the Internet, leading to an unsatisfactory performance on real-world applications. To mitigate these issues, Tian et al. (2021) proposed to adopt class-wise text descriptions for long-tailed image recognition, while our method seeks to learn video-language representations and further extends the framework to varied video recognition settings.

General Video Recognition (GVR). While GVR has not been defined in the existing literature, we briefly summarize these sub tasks of GVR: long-tailed classification, few-shot learning and open-set classification. Long-tailed classification has been extensively studied based on re-sampling data (Buda et al., 2018; Shen et al., 2016), re-weighting loss (Cao et al., 2019; Wang et al., 2017) and transferring strategy (Kang et al., 2019; Liu et al., 2019). Specifically, Zhang et al. (2021) proposed to dynamically sample frames for long-tailed video recognition. As for few-shot video classification, there has been a wide range of approaches, including key frame representation memory (Zhu & Yang, 2018), adversarial video-level feature generation (Kumar Dwivedi et al., 2019), and networks to utilize temporal information (Perrett et al., 2021). As for open set video recognition, Shu et al. (2018) proposed ODN to gradually append new classes to the classification head, and Bao et al. (2021) incorporated evidential learning for uncertainty-aware video recognition.

Compared with some current visual-linguistic approaches (Wang et al., 2021b; Ju et al., 2021) for tasks related to video recognition, our method can not only provide a comprehensive video-language representation to bridge the gap between videos and texts in different cases, but also effectively utilize noisy web text annotations in practical applications.

3 KINETICS-GVR

To simulate the real-world video recognition from different scenarios, we build a comprehensive video recognition benchmark of Kinetics-GVR, consisting of Kinetics-Close, Kinetics-LT, Kinetics-Fewshot, and Kinetics-Open. Our benchmark is curated from the Kinetics-400 (Kay et al., 2017). To obtain the text descriptions on labels, we crawl text entries from the Internet.

Kinetics-Close. We adopt the original Kinetics400 (Kay et al., 2017) for close-set setting, which contains activities in daily life and has around 300k trimmed videos covering 400 categories.

Kinetics-LT. For the long-tailed case, we construct the Kinetics-LT dataset, which is a long-tailed version of Kinetics400 by sampling a subset following the Pareto distribution (Liu et al., 2019). Overall, it contains about 34.1K videos from 400 categories, with maximally 930 videos per class and minimally 5 videos per class. The test set of it is the same as the original version.

Kinetics-Fewshot. In the few-shot setting, we adopt the few-shot version of Kinetics (Zhu & Yang, 2018), which has been frequently used in previous works (Zhu & Yang, 2020; Bishay et al., 2019; Perrett et al., 2021). In this setup, 100 videos from 100 classes are selected, with 64/12/24 classes used for train/val/test.

Kinetics-Open. For the open-set case, we split the Kinetics400 into two parts, with 250 categories for training and the remaining 150 categories for evaluation. Videos in the training set and validation set are from different categories.

Text descriptions. The text descriptions are mainly crawled from Wikipedia (Wikipedia, 2022) and wikiHow (wikiHow, 2022). We first use the label as the keyword to search for the best matching entry. Then, we filter out some irrelated parts of the entries, such as "references", and "bibliography", *etc.*, to obtain the external text descriptions for each class. In addition, we also append 96 prompt sentences for each class as basic descriptions, which are generated by filling the pre-set templates, like '`a video of a {label}`', with label names.

For more details about these datasets, please refer to the Sec. A of the Appendices.

4 METHOD

We first introduce the architecture of our proposed framework in Sec. 4.1, and then discuss its training strategy in Sec. 4.2. Finally, we present how to adapt our framework for different tasks in Sec. 4.3.

4.1 OVERVIEW

To effectively connect the video and language such that language concepts can relate to visual representations for general video recognition, we adopt a transformer-based Siamese network architecture (Radford et al., 2021), consisting of a video encoder $\Phi_{\text{video}}(\cdot)$ and a language encoder $\Phi_{\text{text}}(\cdot)$, to provide the visual representation and linguistic representation respectively. Specially, the video encoder $\Phi_{\text{video}}(\cdot)$ is constructed with a frame encoder $\Phi_{\text{img}}(\cdot)$ followed by a temporal module $\Phi_{\text{temp}}(\cdot)$, which aggregates spatial features obtained from $\Phi_{\text{img}}(\cdot)$ over the temporal dimension.

As shown in the top of the Figure 2, we first randomly sample a batch of videos $\mathcal{V} = \{V_i\}_{i=1}^{N}$, and the corresponding text sentences $\mathcal{T} = \{T_i\}_{i=i}^{N}$, where V_i and T_i are of the same class, N denotes the batch size, and each video contains F frames $V = \{I_i\}_{i=1}^{F}$. For texts $\mathcal{T}$, they are fed to the language encoder $\Phi_{\text{text}}(\cdot)$ to yield text embeddings E^T, while for videos $\mathcal{V}$, they are fed to the video encoder $\Phi_{\text{video}}(\cdot)$ to yield video embeddings E^V, by extracting frame features with $\Phi_{\text{img}}(\cdot)$ and then aggregating features along the temporal dimension with $\Phi_{\text{temp}}(\cdot)$:

$$E_i^T = \Phi_{\text{text}}(T_i), \quad E_i^V = \Phi_{\text{video}}(V_i) = \Phi_{\text{temp}}(\{\Phi_{\text{img}}(I_1), ..., \Phi_{\text{img}}(I_F)\}). \tag{1}$$

After that, we use a bi-modal attention head to aggregate the visual and linguistic features and then obtain the final prediction, as shown in the bottom of Figure 2.

As raw text descriptions crawled from the Internet are noisy, it is necessary to obtain the salient sentences (namely, clean text descriptions) described in Sec. 4.2. The salient sentences reduce the impacts of noises for final prediction, which has been demonstrated in experiments in the Sec. D

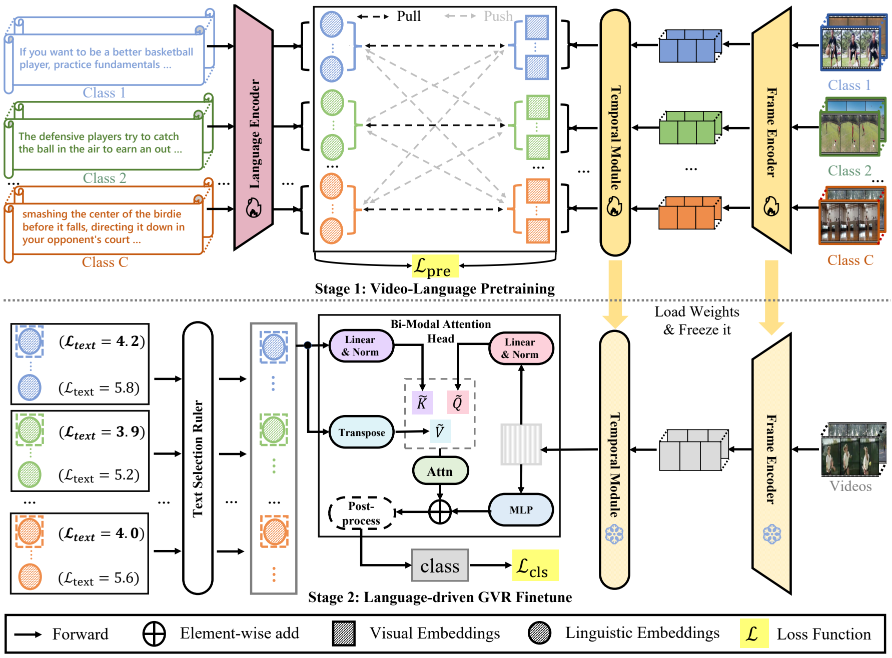

If you want to be a better basketball player, practice fundamentals ...
Class 1
The defensive players try to catch the ball in the air to earn an out ...
Class 2
smashing the center of the birdie before it falls, directing it down in your opponent's court ...
Class C
Language Encoder
Pull
Push
$\mathcal{L}_{\text{pre}}$
Stage 1: Video-Language Pretraining
Temporal Module
Frame Encoder
Class 1
Class 2
Class C
Load Weights & Freeze it
$(\mathcal{L}_{text} = 4.2)$
$(\mathcal{L}_{\text{text}} = 5.8)$
$(\mathcal{L}_{text} = 3.9)$
$(\mathcal{L}_{\text{text}} = 5.2)$
$(\mathcal{L}_{text} = 4.0)$
$(\mathcal{L}_{\text{text}} = 5.6)$
Text Selection Ruler
Bi-Modal Attention Head
Linear & Norm
Linear & Norm
$\widetilde{K}$
$\widetilde{Q}$
Transpose
$\widetilde{V}$
Attn
Post-process
MLP
class
$\mathcal{L}_{\text{cls}}$
Stage 2: Language-driven GVR Finetune
Temporal Module
Frame Encoder
Videos
Forward
Element-wise add
Visual Embeddings
Linguistic Embeddings
$\mathcal{L}$
Loss Function

Figure 2: **The pipeline of VLG.** The entire framework has two training stages. In the first stage, video-language pretraining (VLP) takes both the videos and text descriptions of each category as inputs, learning to link the two modalities through contrastive learning. In the second stage, embeddings of salient sentences, determined by the text selection ruler, are fed into the bi-modal attention head to make final predictions.

of the Appendix. The bi-modal attention head dynamically fuses the video embeddings and text embeddings of salient sentences based on the attention weights. Specially, given video embedding $E^V \in \mathbb{R}^D$ and salient text embeddings of a certain class $E^T \in \mathbb{R}^{M \times D}$, we first calculate the query $\widetilde{Q} \in \mathbb{R}^D$, key $\widetilde{K} \in \mathbb{R}^{M \times D}$ and value $\widetilde{V} \in \mathbb{R}^{M \times D}$ of the attention operation.

$$\widetilde{Q} = \text{Linear}(\text{LayerNorm}(E^V)), \quad \widetilde{K} = \text{Linear}(\text{LayerNorm}(E^T)), \quad \widetilde{V} = E^T, \tag{2}$$

where C is the class number, and M is the maximum number of sentences for each class, corresponding to the number of sampled salient sentences. Next, we adopt an attention operation to gather these M salient sentence embeddings for $\widetilde{G} \in \mathbb{R}^D$:

$$\widetilde{G} = \text{Softmax}(\frac{\widetilde{Q}\widetilde{K}^{\mathsf{T}}}{\sqrt{D}})\widetilde{V}. \tag{3}$$

Then, we perform broadcasting to gather the salient sentences embeddings over all the classes for $G \in \mathbb{R}^{C \times D}$, where C is the class number. The final classification probabilities are obtained based on the video embeddings E^V and enhanced text embeddings G:

$$P^V = \text{Softmax}(\text{MLP}(E^V)), \quad P^T = \text{Softmax}(\text{sim}(E^V, G)/\tau), \quad P = P^V + P^T, \tag{4}$$

where P is the classification probability of the video, consisting of two terms, respectively for classification probability based on video representation P^V, and classification probability based on language representation P^T. $\text{sim}(\cdot, \cdot)$ denotes cosine similarity and τ is a learned parameter.

4.2 Training

We train our framework in two stages, namely Video-Language Pretraining (*VLP*) and Language-driven GVR Finetune, and design specific loss functions, *i.e.* $\mathcal{L}_{\text{pre}}$ and $\mathcal{L}_{\text{cls}}$, respectively for pretraining and classification.

Stage I: Video-Language Pretraining. We jointly optimize the language encoder and video encoder together with the temporal module. The video features would be enclosed to their related category descriptions with higher similarity, and pulled away from irrelated sentences. Specially, we use two contrastive learning NCE losses respectively for video embeddings E^V and text embeddings E^T:

$$\mathcal{L}_{\text{text}} = -\frac{1}{|\mathcal{V}_i^+|} \sum_{V_j \in \mathcal{V}_i^+} \log \frac{\exp(\text{sim}(E_j^V, E_i^T)/\tau)}{\sum_{V_k \in \mathcal{V}} \exp(\text{sim}(E_k^V, E_i^T)/\tau)}, \tag{5}$$

$$\mathcal{L}_{\text{video}} = -\frac{1}{|\mathcal{T}_i^+|} \sum_{T_j \in \mathcal{T}_i^+} \log \frac{\exp(\text{sim}(E_j^T, E_i^V)/\tau)}{\sum_{T_k \in \mathcal{T}} \exp(\text{sim}(E_k^T, E_i^V)/\tau)}, \tag{6}$$

where $\mathcal{L}_{\text{video}}$ and $\mathcal{L}_{\text{text}}$ represent the video and language losses respectively. $\mathcal{V}_i^+$ indicates a subset of $\mathcal{V}$, where all videos are of the same category with the text T_i. Similarly, all texts in $\mathcal{T}_i^+$ share the same class with the video V_i.

To effectively promote our framework for learning to connect the cross-modal information with limited text corpus, we adopt CLIP (Radford et al., 2021) pretrained model as the teacher model to distill knowledge for better visual-linguistic representation. To aggregate the frame features along the temporal dimension, the teacher model replaces the temporal module with the average pooling and outputs the same dimensions of embeddings as the student model. Their visual-linguistic similarities are used as soft targets for training weights associated with the student networks by the following objective:

$$S_{\mathcal{V}} = \frac{\exp(\text{sim}(E_i^V, E_i^T)/\tau)}{\sum_{V_j \in \mathcal{V}} \exp(\text{sim}(E_j^V, E_i^T)/\tau)}, \quad S_{\mathcal{T}} = \frac{\exp(\text{sim}(E_i^T, E_i^V)/\tau)}{\sum_{T_j \in \mathcal{T}} \exp(\text{sim}(E_j^T, E_i^V)/\tau)}, \tag{7}$$

$$\mathcal{L}_{\text{dist}} = -S'_{\mathcal{V}} \cdot \log S_{\mathcal{V}} - S'_{\mathcal{T}} \cdot \log S_{\mathcal{T}}, \tag{8}$$

where S and S' are cosine similarity scores respectively produced by our model and the frozen CLIP model. With this pretraining stage, our framework can not only learn great video-language representation, but also reduce the risk of overfitting limited text corpus data. Therefore, we optimize the video encoder and language encoder via pretraining loss $\mathcal{L}_{\text{pre}}$, defined as a weighted sum of $\mathcal{L}_{\text{video}}$, $\mathcal{L}_{\text{text}}$ and $\mathcal{L}_{\text{dist}}$:

$$\mathcal{L}_{\text{pre}} = \alpha \cdot (\mathcal{L}_{\text{video}} + \mathcal{L}_{\text{text}}) + (1 - \alpha) \cdot \mathcal{L}_{\text{dist}}. \tag{9}$$

Here, α is used to balance $\mathcal{L}_{\text{video}}$, $\mathcal{L}_{\text{text}}$ and $\mathcal{L}_{\text{dist}}$, which is set to 0.5 in our experiments.

Stage II: Language-driven GVR Finetune. In order to take advantage of the valid semantic information and video-language feature, the second stage aims to select the salient sentences by filtering out the noisy texts, and then finetune the bi-modal attention head with the ground truth label.

To filter out noisy texts, we design a training-free text selection ruler (*TSR*) after obtaining the text embeddings, to sample the most discriminative sentences for each category. Specially, we randomly choose λ videos of each class to construct a video batch V'. Then, we calculate $\mathcal{L}_{\text{text}}$ between each sentence and video batch V'. Finally, we select M sentences with the smallest $\mathcal{L}_{\text{text}}$ for the following classification. Note that the TSR only needs to perform once at stage II.

To finetune the bi-modal attention head, we adopt two Cross Entropy losses $\mathcal{L}_{\text{CE}}$ for P^V and P^T (see Eq. 4) respectively:

$$\mathcal{L}_{\text{cls}} = \mathcal{L}_{\text{CE}}(P^V, \mathbf{y}) + \mathcal{L}_{\text{CE}}(P^T, \mathbf{y}), \tag{10}$$

where $\mathbf{y}$ is the ground truth label.

4.3 VLG for General Video Recognition

In most cases, given a query video and pre-selected text embeddings of salient sentences, we first feed the query video into the video encoder to obtain video embeddings. Then, the final result is predicted with the video embeddings and text embeddings of salient sentences through a video-language attention head. We follow this procedure in both the close-set setting and long-tailed setting. For the few-shot setting, we use base videos to pretrain the encoders during the first stage. Then, we use support videos to select salient sentences when combining the linguistic features, or directly use the video embeddings from VLP for linear probe testing. For the open-set setting, we follow the common procedure to train the framework, and insert a post-process step, which can be instantiated as the off-the-shelf open-set procedures (*e.g.*, OpenMax (Bendale & Boult, 2016), Softmax with threshold, *etc.*), to recognize the novel videos during inference.

Table 1: **Results on Kinetics-Close.** By introducing the class-wise text descriptions, our model achieves superior performance to the existing approaches. “IN” denotes ImageNet and "K400" denotes Kinetics400. "-" indicates the numbers are not available for us. "CLIP*" denotes that the model is initialized with the weights pretrained on 400M image-text pairs provided in CLIP (Radford et al., 2021). The total GFLOPs are calculated by the number of views and GFLOPs (per-view).

Method	Pretrain	Frame	Views	Top-1	Top-5	GFLOPs (per-view)	Param (M)
SlowFast, R101+NL (Feichtenhofer et al., 2019)	None	16	10 × 3	79.8	93.9	234.0	59.9
MViT-B, 64×3 (Fan et al., 2021)		64	3 × 3	**81.2**	95.1	455.0	36.6
TSM, ResNeXt101 (Lin et al., 2019)	IN-1K	8	10 × 3	76.3	-	-	-
TANet, R152 (Liu et al., 2021c)		16	4 × 3	79.3	94.1	242.0	-
TDN, R101 (Wang et al., 2021a)		24	10 × 3	**79.4**	94.4	198.0	88.0
ViViT-L/16x2 (Arnab et al., 2021)	IN-21K	32	4 × 3	80.6	94.7	-	-
TimeSformer-L (Bertasius et al., 2021)		8	1 × 3	80.7	94.7	2380.0	121.4
ViViT-L/16x2 (320) (Arnab et al., 2021)		32	4 × 3	81.3	94.7	3992.0	310.8
Swin-L (384) (Liu et al., 2021b)		32	10 × 5	84.9	96.7	2107.0	200.0
MViTv2-L (312) (Li et al., 2021b)		40	5 × 3	**86.1**	97.0	2828.0	217.6
ViViT-H/16x2 (Arnab et al., 2021)	JFT	32	4 × 3	84.8	95.8	-	-
TokenLearner 16at18 (L/10) (Ryoo et al., 2021)		-	4 × 3	**85.4**	96.3	4076.0	450.0
CLIP-Raw, R50 (Radford et al., 2021)	CLIP*	8	1 × 1	46.2	60.8	52.1	102.0
CLIP-Raw, ViT-B/16 (Radford et al., 2021)		8	1 × 1	55.0	67.5	144.0	150.0
CLIP-Close, R50 (Radford et al., 2021)		8	1 × 1	68.1	87.7	49.7	115.0
CLIP-Close, ViT-B/16 (Radford et al., 2021)		8	1 × 1	78.9	93.5	141.0	106.0
ActionCLIP (ViT-B/16) (Wang et al., 2021b)		16	10 × 3	**82.6**	96.2	563.1	141.7
VLG, R50	CLIP*	8	1 × 1	72.3	90.8	76.7	148.0
VLG, ViT-B/16		8	1 × 1	81.8	95.3	148.0	121.0
VLG, ViT-B/16		16	1 × 1	82.4	95.8	282.3	121.0
VLG, ViT-B/16		16	4 × 3	82.9	96.1	282.3	121.0
VLG, ViT-L/14		8	1 × 1	85.5	96.3	650.3	371.0
VLG, ViT-L/14		8	4 × 3	**86.4**	**97.0**	650.3	371.0

5 EXPERIMENTS

We first introduce the evaluation metrics for different settings in Sec. 5.1, before presenting state-of-the-art results over all these four benchmarks: Kinetics-Close, Kinetics-LT, Kinetics-Fewshot, and Kinetics-Open, respectively in Sec. 5.2, Sec. 5.3, Sec. 5.4, and Sec. 5.5. We then present some representative visualization in Sec. 5.6. More details, such as **Experimental Settings** and **Ablation Studies**, *etc.*, are provided respectively in Sec. C and Sec. D of the Appendix.

5.1 EVALUATION METRICS.

We evaluate the performance of our framework under all of these four benchmarks. Besides the top-1 classification accuracy over all classes, for the *long-tailed* setting, we also report the accuracy of three disjoint subsets: *many-shot classes* (more than 100 training videos in each class), *medium-shot classes* (20∼100 training videos in each class), *few-shot classes* (less than 20 training videos in each class). For the *open-set* setting, we use *F-measure* score as a balance between precision and recall.

5.2 EXPERIMENTS ON KINETICS-CLOSE

In Table 1, we compare our proposed methods with prior methods on Kinetice-Close, *i.e.* Kinetics400. There are mainly traditional CNN-based methods, Transformer-based methods and CLIP-based methods. It can be seen that Transformer-based methods and CLIP-based methods achieve better performance than traditional methods. Particularly, our models achieve higher accuracy than other competitors. For example, our method achieves 82.9% top-1 accuracy with ViT-B/16 frame encoder, which exceeds ActionCLIP, a CLIP-based method, with fewer video views. For a fair comparison, we further test our 16 frame ViT-B/16 on the val list of ActionCLIP, and our VLG achieves a higher accuracy performance of 83.5%. Moreover, when using ViT-L/14 as our visual backbone, our VLG can further achieve a higher accuracy of 86.4%, with lower resolution (224px) and fewer computational costs than MViTv2-L (312).

To further demonstrate the superiority of the proposed VLG, we propose CLIP-Raw and CLIP-Close as our baselines on Kinetics-Close to make fair comparisons. CLIP-Raw directly adopts the original CLIP weights and model with only prompt sentences to validate the accuracy performance, while CLIP-Close removing language encoder consists of the frame encoder loading CLIP pretrained weights, temporal module and a linear classifier layer and is finetuned on Kinetics-Close for 100 epochs. One can observe that our method also gets absolute accuracy gain against the baselines with

Table 2: **Results on Kinetics-LT.** Traditional Long-tailed methods use the same visual backbone. CLIP* denotes that the model is initialized by the CLIP (Radford et al., 2021) weights. We report the overall accuracy and the accuracy of three disjoint subsets.

Method	Pretrain	Backbone	Accuracy (%)			
			Overall	Many	Medium	Few
TSN (Wang et al., 2016)	ImageNet	ResNet-50	47.2	59.3	49.4	23.6
TSM (Lin et al., 2019)			46.0	66.3	46.1	17.3
SlowOnly (Feichtenhofer et al., 2019)			44.8	67.7	44.1	14.4
NCM (Kang et al., 2019)	CLIP*	ResNet-50	41.8	53.0	42.3	24.6
cRT (Kang et al., 2019)			43.7	58.9	43.8	22.3
τ-normalized (Kang et al., 2019)			43.9	63.8	43.1	18.5
LWS (Kang et al., 2019)			45.1	58.6	44.8	27.1
SSD-LT (Li et al., 2021a)			48.3	59.6	49.1	30.0
PaCo (Cui et al., 2021)			50.1	60.1	50.3	35.8
CLIP-Raw (Radford et al., 2021)	CLIP*	ResNet-50	46.2	48.3	44.8	46.7
CLIP-LT (Radford et al., 2021)			53.4	70.3	53.3	31.1
VLG			60.8	71.7	60.4	47.2
CLIP-Raw (Radford et al., 2021)	CLIP*	ViT-B/16	55.0	57.1	53.7	55.5
CLIP-LT (Radford et al., 2021)			63.8	79.7	63.8	42.8
VLG			**70.7**	**81.9**	**69.7**	**58.3**

ResNet-50 (72.3% vs. 68.1% vs. 46.2%) and ViT-B/16 (81.8% vs. 78.9% vs. 55.0%) backbones. The results are desirable since our framework can take the advantages of the semantic information in the text descriptions.

5.3 Experiments on Kinetics-LT

In Table 2, we can see that our VLG models are superior to conventional vision-based methods with the same video encoders. Since there are few long-tailed methods specific to videos, we re-implement and report the performance of some representative image long-tailed methods on Kinetics400-LT, such as τ-normalized, cRT, NCM, LWS (Kang et al., 2019), PaCo (Cui et al., 2021), and SSD-LT (Li et al., 2021a), which are all initialized with CLIP pretrained weights. We also add an additional temporal pooling without introducing any new parameters to aggregate features along the temporal dimension for them. In addition, we also build CLIP-LT and CLIP-Raw as our simple baseline based on CLIP to corroborate our method. CLIP-LT is built the same as CLIP-Close.

It can be seen that our proposed method is superior to prior visual-based methods with the same backbone. For example, using the same ResNet-50 backbone, the overall accuracy of VLG reaches 60.8%, which outperforms SSD-LT by 12.5 points (60.8% vs. 48.3%), and 10.7% better than PaCo (60.8% vs. 50.1%). Moreover, when compared to CLIP baseline models, the performance of our method is also promising, which is 7.4% better than the CLIP-LT, and 14.6% better than the CLIP-Raw (60.8% vs. 53.4% vs. 46.2%). When using ViT-B/16 as the backbone, the overall accuracy of VLG can further boost up to 70.7%.

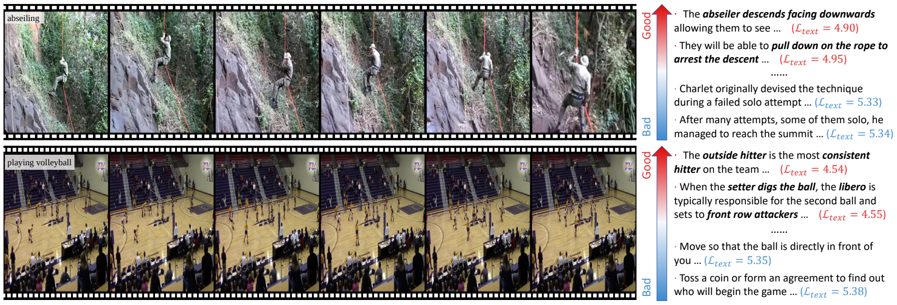
5.4 Experiments on Kinetics-Fewshot

Following Ju et al. (2021), we conduct two few-shot settings: **5-shot-5-way** and **5-shot-C-way**.

5-shot-5-way. For a fair comparison, this setting adopts the publicly accessible few-shot splits. During training, we simply use the base split for our first pretraining stage, without meta-learning paradigms. During the evaluation, we report average results over 200 trials with random sampling on the test split. Table 3 presents the average top-1 accuracy, and our method clearly achieves significant performance. Following CLIP (Radford et al., 2021), we directly adopt the linear probe to test the visual representation output from the video encoder, which obtains 84.6% top-1 accuracy and is higher than the traditional few-shot learning methods. When combining the linguistic features, the performance can further boost up to 94.0%. We also use the same network settings with textual information following Ju et al. (2021), and achieve better performance.

Table 3: **Results on Kinetics-Fewshot.** Here, $\mathcal{C}_{ALL}$ denotes the model is tested on all categories of the corresponding dataset. In Kinetics-fewshot, $\mathcal{C}_{ALL} = 400$. "VLG-L" denotes our method with linear probe testing.

Method	Backbone	K-shot	N-way	Top-1
CMN (Zhu & Yang, 2018)	ResNet-50	5	5	78.9
TARN (Bishay et al., 2019)		5	5	78.5
ARN (Zhang et al., 2020)		5	5	82.4
VLG-L		5	5	84.6
VLG		5	5	**94.0**
E-Prompt (Ju et al., 2021)	ViT-B/16	5	5	96.4
VLG		5	5	**96.9**
E-Prompt (Ju et al., 2021)	ViT-B/16	5	$\mathcal{C}_{ALL}$	58.5
VLG		5	$\mathcal{C}_{ALL}$	**62.8**



· The ***abseiler descends facing downwards*** allowing them to see … ($\mathcal{L}_{text} = 4.90$)

· They will be able to ***pull down on the rope to arrest the descent*** … ($\mathcal{L}_{text} = 4.95$)

……

· Charlet originally devised the technique during a failed solo attempt … ($\mathcal{L}_{text} = 5.33$)

· After many attempts, some of them solo, he managed to reach the summit … ($\mathcal{L}_{text} = 5.34$)

· The ***outside hitter*** is the most ***consistent hitter*** on the team … ($\mathcal{L}_{text} = 4.54$)

· When the ***setter digs the ball***, the ***libero*** is typically responsible for the second ball and sets to ***front row attackers*** … ($\mathcal{L}_{text} = 4.55$)

……

· Move so that the ball is directly in front of you … ($\mathcal{L}_{text} = 5.35$)

· Toss a coin or form an agreement to find out who will begin the game … ($\mathcal{L}_{text} = 5.38$)

Figure 3: **Visualization of some text descriptions with corresponding $\mathcal{L}_{\textbf{text}}$.** The values of $\mathcal{L}_{\text{text}}$ reflect the saliency of these sentences, indicating the effectiveness of our TSR.

5-shot-C-way. We further investigate a more challenging experiment setting, which samples 5 videos from the training set for each class as the base split, and then directly evaluates the model on the standard Kinetics400 testing split. For statistical stability, we report the average results over 10 trials to ensure the reliability of results. It can be seen that our model still obtains a superior performance (62.8% vs. 58.5%), which is also higher than Ju et al. (2021).

5.5 Experiments on Kinetics-Open

Openset video recognition aims to not only accurately classify known categories which have appeared in training, but also recognize unknown categories which are not seen in training. Without any other modifications to our framework, we only adopt softmax with *thresholds* and *OepnMax* (Bendale & Boult, 2016) as a post-process on the prediction logits to obtain the classification results, as described in Sec. 4.3. In addition, we also re-implement the OLTR with CLIP initialization as a comparison. As shown in Table 4, we outperform OLTR (Liu et al., 2019) among all different threshold numbers, indicating the significance of our video-language representation.

Table 4: **Results on Kinetics-Open.** OLTR and VLG are both initialized by CLIP weight. With the same backbone, VLG outperforms OLTR among all thresholds.

Method	Post-process	F-measure				
		thr=0.1	thr=0.2	thr=0.3	thr=0.5	thr=0.7
OLTR, R50 (Liu et al., 2019)	Threshold	0.490	0.504	0.513	0.502	0.458
VLG, R50	Threshold	0.610	0.639	**0.654**	**0.610**	**0.469**
VLG, R50	OpenMax	**0.616**	**0.641**	0.651	0.614	0.465
VLG, ViT-B/16	Threshold	0.657	0.672	0.694	**0.721**	0.697
VLG, ViT-B/16	OpenMax	0.694	0.698	**0.703**	0.699	0.633

5.6 More Results and Visualizations

As shown in Figure 3, we present some sentences sampled or filtered out by our TSR. We observe that our method can learn specific concepts or steps for each class, such as the "pull down on the rope" for "abseiling" and "libero" for "playing volleyball". The salient sentences commonly contain these words of specific concepts in the category. More examples are provided in the Appendix.

More results about ablation studies on the effectiveness of our video-language pretraining, CLIP pretrained weights, bi-modal attention head, temporal module, distillation loss and text selection ruler, *etc.*, can be found in the Sec. D of the appendix. In addition, we also provide additional visualization in Sec. F to illustrate the class-level performance improvement on long-tailed videos, examples of text descriptions, and more samples to show the relationship between videos and texts.

6 Conclusions

In this paper, we have studied the general video recognition (GVR) under four different settings. The GVR task enables us to examine the generalization ability of a video recognition model in real-world applications. To facilitate the research of GVR, we build comprehensive video benchmarks of Kinetics-GVR containing text descriptions for all action classes. Then, we propose a unified visual-linguistic framework (VLG) to accomplish the task of GVR. In particular, we present an effective two-stage training strategy to effectively adapt the image-text representation to video domain for GVR. Extensive results demonstrate that our VLG obtains the state-of-the-art performance under all settings on the Kinetics-GVR benchmark. We hope that the datasets and framework will help the future research in GVR.

References

Andra Acsintoae, Andrei Florescu, Mariana-Iuliana Georgescu, Tudor Mare, Paul Sumedrea, Radu Tudor Ionescu, Fahad Shahbaz Khan, and Mubarak Shah. Ubnormal: New benchmark for supervised open-set video anomaly detection. *arXiv preprint arXiv:2111.08644*, 2021.

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.

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

Wentao Bao, Qi Yu, and Yu Kong. Evidential deep learning for open set action recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13349–13358, 2021.

Abhijit Bendale and Terrance E Boult. Towards open set deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1563–1572, 2016.

Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding. *arXiv preprint arXiv:2102.05095*, 2(3):4, 2021.

Mina Bishay, Georgios Zoumpourlis, and Ioannis Patras. Tarn: Temporal attentive relation network for few-shot and zero-shot action recognition. *arXiv preprint arXiv:1907.09021*, 2019.

Mateusz Buda, Atsuto Maki, and Maciej A Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106:249–259, 2018.

Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. *Advances in neural information processing systems*, 32, 2019.

Kaidi Cao, Jingwei Ji, Zhangjie Cao, Chien-Yi Chang, and Juan Carlos Niebles. Few-shot video classification via temporal alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10618–10627, 2020.

Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6299–6308, 2017.

Jiequan Cui, Zhisheng Zhong, Shu Liu, Bei Yu, and Jiaya Jia. Parametric contrastive learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 715–724, 2021.

Jianfeng Dong, Xirong Li, Chaoxi Xu, Xun Yang, Gang Yang, Xun Wang, and Meng Wang. Dual encoding for video retrieval by text. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

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, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1933–1941, 2016.

Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6202–6211, 2019.

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.

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

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.

Boyuan Jiang, MengMeng Wang, Weihao Gan, Wei Wu, and Junjie Yan. Stm: Spatiotemporal and motion encoding for action recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2000–2009, 2019.

Chen Ju, Tengda Han, Kunhao Zheng, Ya Zhang, and Weidi Xie. Prompting visual-language models for efficient video understanding. *arXiv preprint arXiv:2112.04478*, 2021.

Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. *arXiv preprint arXiv:1910.09217*, 2019.

Yash Kant, Dhruv Batra, Peter Anderson, Alexander Schwing, Devi Parikh, Jiasen Lu, and Harsh Agrawal. Spatially aware multimodal transformers for textvqa. In *European Conference on Computer Vision*, pp. 715–732. Springer, 2020.

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.

Alexander Klaser, Marcin Marszałek, and Cordelia Schmid. A spatio-temporal descriptor based on 3d-gradients. In *BMVC 2008-19th British Machine Vision Conference*, pp. 275–1. British Machine Vision Association, 2008.

Sai Kumar Dwivedi, Vikram Gupta, Rahul Mitra, Shuaib Ahmed, and Arjun Jain. Protogan: Towards few shot learning for action recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pp. 0–0, 2019.

Tianhao Li and Limin Wang. Learning spatiotemporal features via video and text pair discrimination. *CoRR*, abs/2001.05691, 2020.

Tianhao Li, Limin Wang, and Gangshan Wu. Self supervision to distillation for long-tailed visual recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 630–639, 2021a.

Yan Li, Bin Ji, Xintian Shi, Jianguo Zhang, Bin Kang, and Limin Wang. Tea: Temporal excitation and aggregation for action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 909–918, 2020.

Yanghao Li, Chao-Yuan Wu, Haoqi Fan, Karttikeya Mangalam, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. Improved multiscale vision transformers for classification and detection. *arXiv preprint arXiv:2112.01526*, 2021b.

Zejun Li, Zhihao Fan, Huaixiao Tou, and Zhongyu Wei. Mvp: Multi-stage vision-language pre-training via multi-level semantic alignment. *arXiv preprint arXiv:2201.12596*, 2022.

Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7083–7093, 2019.

Jintao Lin, Haodong Duan, Kai Chen, Dahua Lin, and Limin Wang. Ocsampler: Compressing videos to one clip with single-step sampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13894–13903, 2022.

Yang Liu, Qingchao Chen, and Samuel Albanie. Adaptive cross-modal prototypes for cross-domain visual-language retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14954–14964, 2021a.

Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. *arXiv preprint arXiv:2106.13230*, 2021b.

Zhaoyang Liu, Limin Wang, Wayne Wu, Chen Qian, and Tong Lu. TAM: temporal adaptive module for video recognition. In *ICCV*, pp. 13688–13698. IEEE, 2021c.

Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-scale long-tailed recognition in an open world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2537–2546, 2019.

Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

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

Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9879–9889, 2020.

Toby Perrett, Alessandro Masullo, Tilo Burghardt, Majid Mirmehdi, and Dima Damen. Temporal-relational crosstransformers for few-shot action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 475–484, 2021.

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.

William J Reed. The pareto, zipf and other power laws. *Economics letters*, 74(1):15–19, 2001.

Michael S Ryoo, AJ Piergiovanni, Anurag Arnab, Mostafa Dehghani, and Anelia Angelova. Token-learner: What can 8 learned tokens do for images and videos? *arXiv preprint arXiv:2106.11297*, 2021.

Gilad Sharir, Asaf Noy, and Lihi Zelnik-Manor. An image is worth 16x16 words, what is a video worth? *arXiv preprint arXiv:2103.13915*, 2021.

Li Shen, Zhouchen Lin, and Qingming Huang. Relay backpropagation for effective learning of deep convolutional neural networks. In *European conference on computer vision*, pp. 467–482. Springer, 2016.

Yu Shu, Yemin Shi, Yaowei Wang, Yixiong Zou, Qingsheng Yuan, and Yonghong Tian. Odn: Opening the deep network for open-set action recognition. In *2018 IEEE international conference on multimedia and expo (ICME)*, pp. 1–6. IEEE, 2018.

Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. *Advances in neural information processing systems*, 27, 2014.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8317–8326, 2019.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

Jonathan Stroud, David Ross, Chen Sun, Jia Deng, and Rahul Sukthankar. D3d: Distilled 3d networks for video action recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 625–634, 2020.

Changyao Tian, Wenhai Wang, Xizhou Zhu, Xiaogang Wang, Jifeng Dai, and Yu Qiao. Vl-ltr: Learning class-wise visual-linguistic representation for long-tailed visual recognition. *arXiv preprint arXiv:2111.13579*, 2021.

Du Tran, Lubomir D. Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pp. 4489–4497. IEEE Computer Society, 2015. doi: 10.1109/ICCV.2015.510. URL `https://doi.org/10.1109/ICCV.2015.510`.

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

Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. Dense trajectories and motion boundary descriptors for action recognition. *International journal of computer vision*, 103(1): 60–79, 2013.

Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *European conference on computer vision*, pp. 20–36. Springer, 2016.

Limin Wang, Zhan Tong, Bin Ji, and Gangshan Wu. Tdn: Temporal difference networks for efficient action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1895–1904, June 2021a.

Mengmeng Wang, Jiazheng Xing, and Yong Liu. Actionclip: A new paradigm for video action recognition. *arXiv preprint arXiv:2109.08472*, 2021b.

Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. Learning to model the tail. *Advances in Neural Information Processing Systems*, 30, 2017.

wikiHow. wikhow, the most trusted how-to site on the internet. `https://www.wikihow.com/Main-Page`, 2022. [Online; accessed 19-May-2022].

Wikipedia. Wikipedia, the free encyclopedia. `https://www.wikipedia.org/`, 2022. [Online; accessed 19-May-2022].

Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 305–321, 2018.

Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021.

Hongguang Zhang, Li Zhang, Xiaojuan Qi, Hongdong Li, Philip HS Torr, and Piotr Koniusz. Few-shot action recognition with permutation-invariant attention. In *European Conference on Computer Vision*, pp. 525–542. Springer, 2020.

Xing Zhang, Zuxuan Wu, Zejia Weng, Huazhu Fu, Jingjing Chen, Yu-Gang Jiang, and Larry S Davis. Videolt: Large-scale long-tailed video recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7960–7969, 2021.

Linchao Zhu and Yi Yang. Compound memory networks for few-shot video classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 751–766, 2018.

Linchao Zhu and Yi Yang. Label independent memory for semi-supervised few-shot video classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(1):273–285, 2020.

Linchao Zhu, Du Tran, Laura Sevilla-Lara, Yi Yang, Matt Feiszli, and Heng Wang. Faster recurrent networks for efficient video classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 13098–13105, 2020.

Appendix

In this supplementary material, we first provide some details on video splits and text descriptions of our proposed benchmarks in Sec. A. Then, we summarize the notations used in the paper in Sec. B, and implementation details in Sec. C. We also provide results of ablation studies and additional experiments respectively in Sec. D and Sec. E. Finally, we provide more visualization and discuss the limitation of our method, respectively in Sec. F and Sec. G. We also present the ethic statement and reproducibility statement of our method in Sec. H and Sec. I respectively.

A Benchmark Details

A.1 Kinetics-Close

We directly adopt the original Kinetics400 (Kay et al., 2017) for close-set setting, which contains activities in daily life and has around 300k trimmed videos covering 400 categories. Because of the expirations of some YouTube links, some original videos are missing over time. Our copy includes 240436 training videos and 19796 validation videos.

A.2 Kinetics-LT

For the long-tailed case, we construct the Kinetics-LT dataset, which is a long-tailed version of Kinetics400 by sampling a subset following the Pareto distribution (Reed, 2001) similar to ImageNet-LT (Liu et al., 2019), with 930~5 videos per class from the 400 classes of Kinetics400 dataset. Videos are randomly selected based on the distribution values of each class, and the 400 classes are randomly split into 109 many-shot classes, 209 medium-shot classes, and 82 few-shot classes. We randomly select 20 training videos per class from the original training set as the validation set. The original validation set of Kinetics400 is used as the testing set in this paper. The dataset specifications are shown in Figure 4.

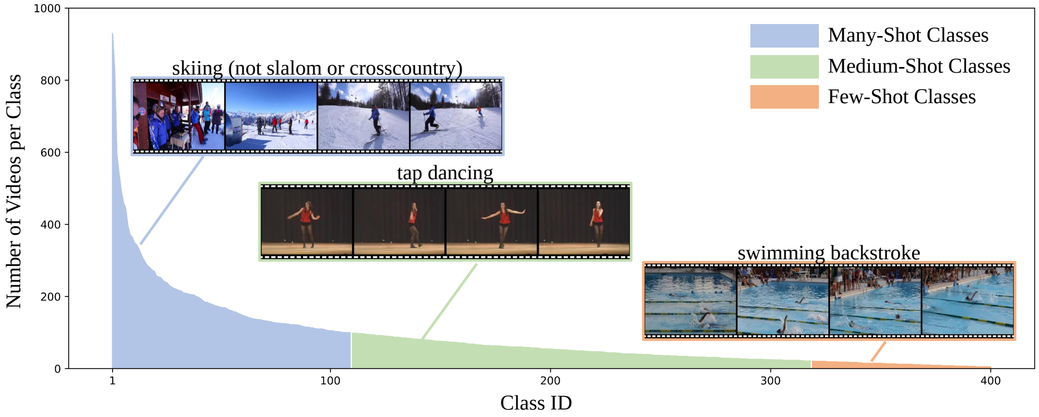


Figure 4: **The dataset statistics of Kinetics-LT.**

A.3 Kinetics-Fewshot

For the few-shot case, we conduct two kinds of few-shot settings, *i.e.*, **5-shot-5-way** and **5-shot-C-way**.

For the **5-shot-5-way setting**, we adopt the few-shot version of Kinetics (Zhu & Yang, 2018; 2020), which has been frequently used to evaluate few-shot video recognition in previous works (Zhu & Yang, 2018; 2020; Bishay et al., 2019; Zhang et al., 2020; Cao et al., 2020; Perrett et al., 2021). In this setup, 100 videos from 100 classes are selected, with 64, 12 and 24 classes used for train/val/test. We conduct 200 trials with random samplings, to ensure the statistical significance.

Specifically, the **train action categories** are sampled from: *air drumming, arm wrestling, beatboxing, biking through snow, blowing glass, blowing out candles, bowling, breakdancing, bungee jumping, catching or throwing baseball, cheerleading, cleaning floor, contact juggling, cooking chicken, country line dancing, curling hair, deadlifting, doing nails, dribbling basketball, driving tractor, drop kicking, dying hair, eating burger, feeding birds, giving or receiving award, hopscotch, jetskiing, jumping into pool, laughing, making snowman, massaging back, mowing lawn, opening bottle, playing accordion, playing badminton, playing basketball, playing didgeridoo, playing ice hockey, playing keyboard, playing ukulele, playing xylophone, presenting weather forecast, punching bag, pushing cart, reading book, riding unicycle, shaking head, sharpening pencil, shaving head, shot put, shuffling cards, slacklining, sled dog racing, snowboarding, somersaulting, squat, surfing crowd, trapezing, using computer, washing dishes, washing hands, water skiing, waxing legs, weaving basket*.

The **val action categories** are sampled from: *baking cookies, crossing river, dunking basketball, feeding fish, flying kite, high kick, javelin throw, playing trombone, scuba diving, skateboarding, ski jumping, trimming or shaving beard*.

The **test action categories** are sampled from: *blasting sand, busking, cutting watermelon, dancing ballet, dancing charleston, dancing macarena, diving cliff, filling eyebrows, folding paper, hula hooping, hurling (sport), ice skating, paragliding, playing drums, playing monopoly, playing trumpet, pushing car, riding elephant, shearing sheep, side kick, stretching arm, tap dancing, throwing axe, unboxing*.

For the **5-shot-C-way setting**, we follow (Ju et al., 2021) to sample 5 videos from all categories to construct the training dataset, and measure the performance on the standard validation set, *i.e.* all videos from all categories in the validation set of Kinetics400. For statistical significance, we also conduct 10 random sampling rounds to choose training videos.

A.4 Kinetics-Open

For the open-set case, we split the Kinetics400 into two parts, with 250 categories for training and the remaining 150 categories for evaluation. Videos in the training set and validation set are from different categories.

Specifically, the **train action categories** are sampled from: *air drumming, answering questions, applying cream, archery, arm wrestling, arranging flowers, assembling computer, baby waking up, balloon blowing, bandaging, barbequing, bartending, bee keeping, belly dancing, bending back, bending metal, biking through snow, blowing glass, blowing nose, blowing out candles, bookbinding, bouncing on trampoline, breading or breadcrumbing, breakdancing, brush painting, brushing teeth, bungee jumping, carrying baby, cartwheeling, carving pumpkin, catching or throwing frisbee, celebrating, changing oil, checking tires, cheerleading, chopping wood, clean and jerk, cleaning floor, cleaning gutters, cleaning shoes, cleaning toilet, cleaning windows, climbing a rope, climbing ladder, climbing tree, cooking egg, cooking sausages, counting money, cracking neck, crossing river, crying, cutting nails, cutting watermelon, dancing charleston, decorating the christmas tree, digging, disc golfing, diving cliff, doing laundry, doing nails, drinking, drinking beer, drinking shots, driving car, driving tractor, drumming fingers, dunking basketball, dying hair, eating cake, eating carrots, eating chips, eating doughnuts, eating hotdog, eating spaghetti, egg hunting, exercising with an exercise ball, faceplanting, feeding fish, filling eyebrows, flipping pancake, folding napkins, folding paper, front raises, frying vegetables, garbage collecting, gargling, getting a haircut, giving or receiving award, golf chipping, golf driving, grinding meat, grooming dog, grooming horse, headbanging, headbutting, high kick, hitting baseball, hockey stop, holding snake, hopscotch, hoverboarding, hugging, hula hooping, hurdling, hurling (sport), ice climbing, ice skating, javelin throw, jetskiing, jogging, juggling fire, juggling soccer ball, jumpstyle dancing, kicking soccer ball, kissing, krumping, laying bricks, making bed, making pizza, making snowman, making sushi, making tea, massaging feet, massaging person's head, mopping floor, motorcycling, moving furniture, opening present, parasailing, passing American football (in game), peeling apples, petting animal (not cat), petting cat, picking fruit, planting trees, plastering, playing accordion, playing badminton, playing bagpipes, playing basketball, playing bass guitar, playing cello, playing chess, playing clarinet, playing cymbals, playing didgeridoo, playing drums, playing flute, playing guitar, playing harmonica, playing harp, playing ice hockey, playing keyboard, playing monopoly, playing organ, playing paintball, playing piano, playing squash or racquetball, playing tennis, playing trombone, playing* *ukulele, playing violin, playing xylophone, pole vault, presenting weather forecast, pull ups, punching person (boxing), push up, pushing car, pushing wheelchair, reading book, riding camel, riding mountain bike, riding mule, riding scooter, riding unicycle, rock climbing, roller skating, running on treadmill, sailing, salsa dancing, sanding floor, scuba diving, setting table, shaking head, shaving legs, shearing sheep, shining shoes, shooting basketball, shooting goal (soccer), shoveling snow, shredding paper, sign language interpreting, singing, ski jumping, skiing crosscountry, skiing slalom, skipping rope, skydiving, slacklining, snatch weight lifting, sniffing, snowboarding, somersaulting, spinning poi, spraying, springboard diving, squat, stomping grapes, stretching leg, surfing crowd, swimming breast stroke, swimming butterfly stroke, swing dancing, swinging legs, swinging on something, tango dancing, tap dancing, tapping guitar, tasting beer, testifying, throwing discus, tobogganing, tossing coin, training dog, trapezing, trimming or shaving beard, triple jump, unboxing, using computer, using remote controller (not gaming), using segway, vault, waiting in line, walking the dog, washing feet, washing hair, water skiing, watering plants, waxing eyebrows, waxing legs, weaving basket, welding, whistling, windsurfing, wrapping present, wrestling, writing, yawning, zumba*.

The **validation action categories** are sampled from: *abseiling, applauding, auctioning, baking cookies, beatboxing, bench pressing, blasting sand, blowing leaves, bobsledding, bowling, braiding hair, brushing hair, building cabinet, building shed, busking, canoeing or kayaking, capoeira, catching fish, catching or throwing baseball, catching or throwing softball, changing wheel, clapping, clay pottery making, cleaning pool, contact juggling, cooking chicken, cooking on campfire, country line dancing, crawling baby, curling hair, cutting pineapple, dancing ballet, dancing gangnam style, dancing macarena, deadlifting, dining, dodgeball, doing aerobics, drawing, dribbling basketball, drop kicking, eating burger, eating ice cream, eating watermelon, exercising arm, extinguishing fire, feeding birds, feeding goats, finger snapping, fixing hair, flying kite, folding clothes, getting a tattoo, golf putting, gymnastics tumbling, hammer throw, high jump, ice fishing, ironing, juggling balls, jumping into pool, kicking field goal, kitesurfing, knitting, laughing, long jump, lunge, making a cake, making a sandwich, making jewelry, marching, massaging back, massaging legs, milking cow, mowing lawn, news anchoring, opening bottle, paragliding, parkour, passing American football (not in game), peeling potatoes, playing cards, playing controller, playing cricket, playing kickball, playing poker, playing recorder, playing saxophone, playing trumpet, playing volleyball, pumping fist, pumping gas, punching bag, pushing cart, reading newspaper, recording music, riding a bike, riding elephant, riding mechanical bull, riding or walking with horse, ripping paper, robot dancing, rock scissors paper, scrambling eggs, shaking hands, sharpening knives, sharpening pencil, shaving head, shot put, shuffling cards, side kick, situp, skateboarding, skiing (not slalom or crosscountry), slapping, sled dog racing, smoking, smoking hookah, sneezing, snorkeling, snowkiting, snowmobiling, spray painting, sticking tongue out, stretching arm, strumming guitar, surfing water, sweeping floor, swimming backstroke, sword fighting, tai chi, taking a shower, tapping pen, tasting food, texting, throwing axe, throwing ball, tickling, tossing salad, trimming trees, tying bow tie, tying knot (not on a tie), tying tie, unloading truck, washing dishes, washing hands, water sliding, waxing back, waxing chest, yoga*.

A.5 Text Description

The text descriptions are mainly crawled from Wikipedia (Wikipedia, 2022) and wikiHow (wikiHow, 2022). Following Tian et al. (2021), we first use the label name as the keyword to search for the best matching entry. Then, we filter out some irrelated parts of the entries, such as "references", "external links", and "bibliography", *etc.*, to obtain the external text descriptions for each class. In addition, we also append 96 prompt sentences for each class as basic descriptions, which are generated by filling the pre-set templates, like '`a video of a {label}`', with label names.

In Figure 7, we display a part of text descriptions collected for our benchmarks. We see that it is inevitable to include some noisy text descriptions, since these texts are all crawled from the Internet without fine-grained cleaning. In addition, we also report the detailed statistics of the collected text descriptions in Table 5. It can be seen that the text quantity of different classes varies significantly.

B Notations

we summarize the notations used in the paper in Table 6.

Table 5: **Detailed statistics of the text descriptions.** where N_{min}, N_{max}, N_{mean}, and N_{Med} denote for minimum, maximum, mean, and median number of sentences of classes respectively. M_{min}, M_{max}, M_{mean}, and M_{Med} denote for minimum, maximum, mean, and median number of words of classes respectively. L_{Avg} denotes the average number of tokens per sentence.

Datasets	N_{min}	N_{max}	N_{mean}	N_{Med}	M_{min}	M_{max}	M_{mean}	M_{Med}	L_{Avg}
Kinetics400	7	634	143	99	252	20340	4011	2605	28

Table 6: **Summary of notations used in the paper.**

Notation	Meaning
Φ_{video}	Video encoder
Φ_{text}	Language encoder
$\Phi_{\text{img}}(\cdot)$	Frame encoder
$\Phi_{\text{temp}}(\cdot)$	Temporal module
$\mathcal{V} = \{V_i\}_{i=1}^{N}$	A batch of N video samples
$\mathcal{T} = \{T_i\}_{i=1}^{N}$	A batch of N text samples
$V = \{I_i\}_{i=1}^{F}$	A video of F frames
E_i^T	Embeddings of text T_i
E_i^V	Embeddings of video V_i
$\text{sim}(\cdot, \cdot)$	Similarity function
S	Cosine similarity scores produced by our model
S'	Cosine similarity scores produced by the frozen CLIP model
M	Number of sampled sentences per class
$\mathcal{L}_{\text{text}}$	Contrastive learning NCE losses for texts
$\mathcal{L}_{\text{video}}$	Contrastive learning NCE losses for videos
$\mathcal{L}_{\text{dist}}$	Distillation loss
$\mathcal{L}_{\text{pred}}$	Video-Language pretraining loss
$\mathcal{L}_{\text{CE}}$	CrossEntropy loss
$\mathcal{L}_{\text{cls}}$	Language-driven GVR finetune loss
$\mathbf{y}$	Ground truth label

C Implementation Details

Data Pre-processing. If not specified, we use the segment-based input frame sampling strategy (Wang et al., 2016) with 8 frames. During training, we follow Wang et al. (2021b) to process all frames to 224×224 input resolution. During inference, we resize all frames to 256×256 and center-crop them to 224×224.

Network Architectures. If not specified, the video encoder adopts the pre-trained CLIP (Radford et al., 2021) visual encoder (ViT-B/16 (Sharir et al., 2021)) as our frame encoder. For the temporal module, we use a smaller version of the transformer with 6-layers and 8-head self-attention as default. To indicate the temporal order, we also add learnable temporal positional encoding onto the frame features as input. The language encoder also follows that of CLIP (Radford et al., 2021), which is a 12-layer transformer, and the maximum length of text tokens is set to 77 (including `[SOS]` and `[EOS]` tokens). We initialize the frame encoder and language encoder with pretrained weights of CLIP (Radford et al., 2021) during the first stage.

Training Hyper-parameters. In our implementation, we always train the models using an AdamW (Loshchilov & Hutter, 2017) optimizer with the cosine schedule (Loshchilov & Hutter, 2016), a weight decay of 5×10^{-2}, and a momentum of 0.9 for 50 epochs. During the first stage, the size of the mini-batch is set to 16, and α is set to 0.5. The initial learning rate is set to 1×10^{-5} for frame encoder and language encoder, and set to 1×10^{-3} for the temporal module. During the second stage, the size of the mini-batch is set to 128. Both encoders are kept frozen, and the only trainable part is the bi-modal attention head. The learning rate of which is set to 1×10^{-3}. The number of selected sentences per class M is set to 64, and λ is set to 50. We conduct all experiments on 8 V100 GPUs.

Table 7: **Ablation studies on Kinetics-Close.** "Head" denotes the classification head used in stage II, "Bi-M" denotes the bi-modal attention head, "TSR" denotes the proposed text selection ruler, "RAND" denotes random selection strategy, and "BASIC" denotes only using basic prompted sentences.

#	Pretrain	CLIP Weights	Fine-tuning		Top-1
			Head	Ruler	
1	✓	✓	Bi-M	TSR	**81.8**
2	-	✓	Bi-M	TSR	76.0
3	✓	-	Bi-M	TSR	32.6
4	✓	✓	FC	TSR	79.5
5	✓	✓	KNN	TSR	79.9
6	✓	✓	Bi-M	RAND	80.0
7	✓	✓	Bi-M	BASIC	78.9

Table 8: **Ablation studies on the number of layers in the Temporal Module.** We evaluate accuracy on Kinetics-Close and Kinetics-Fewshot with different numbers of layers in temporal module.

(a) **Ablation studies of Temporal Module with different numbers of layers on Kinetics-Close.**

Number of layers	0	1	2	4	6	8
Top-1 Acc	79.8	80.9	81.2	81.4	**81.8**	80.6

(b) **Ablation studies of Temporal Module with different numbers of layers on Kinetics-Fewshot.**

Number of layers	0	1	2	4	6	8
Top-1 Acc	**96.9**	96.5	95.6	95.8	96.1	95.2

D Ablation Study

In order to provide a deep analysis of our proposed method, we also conduct ablation studies on the Kinetics-Close dataset. In these experiments, we use ViT-B/16 as the default backbone. All other settings remain the same as Sec. C unless specifically mentioned.

Video-Language Pre-training. To examine the effectiveness of our video-language pretraining (VLP) framework, we remove it by directly performing the finetuning process on the pretrained weights of CLIP (Radford et al., 2021). As reported in the #1 and #2 of Table 7, the model with VLP outperforms the one without VLP by 5.8 points on the top-1 accuracy. Such a gap might be attributed to the difficulties in learning temporal information and semantic inconsistency between videos and text representation, which can be alleviated by our VLP.

CLIP Pre-trained Weights. To analyze the influence of CLIP pre-trained weights, we train our method with randomly initialized weights. Comparing the #1 and #3 of Table 7, we can see that initializing with CLIP pre-trained weights benefits our approach. This phenomenon is caused by the limited text corpus for pre-training. There are only 400 class descriptions (about 95K sentences) for Kinetics400, and it is easy to overfit a video to a specific set of sentences without a pre-trained linguistic encoder.

Bi-modal Attention Head. We investigate the effectiveness of bi-modal attention head by comparing it with other recognition heads, including FC (only video-based), and KNN (video-language based). As reported in #1, #4 and #5 of Table 7, the proposed head performs better than FC and KNN by 2.3% and 1.9% points respectively. It is notable that, as another bi-modal head, KNN also works better than FC. These results indicate the superiority of bi-modal attention head and the power of video-language representation.

Salient Sentences. We study the significance of the sampled salient sentences by replacing them with those sampled by "Random" and "Basic" strategies. For "Random", we randomly select M sentences from text descriptions. For "Basic", we only use the basic prompt sentences as the salient sentences. As shown in Table 7, the model with TSR (See the #1 of Table 7) outperforms the model with other strategies on the Top-1 accuracy. It indicates the effectiveness of our TSR to filter out some noisy sentences.

Table 9: **Effectiveness of Distillation Loss.** We evaluate the performance on Kinetics-LT to investigate the effectiveness of distillation loss.

Distill Kind	Dataset	Backbone	$1 - \alpha$	Top-1	Many	Medium	Few
-	Kinetics-LT	ResNet-50	0	57.5	70.8	57.2	40.8
Logits	Kinetics-LT	ResNet-50	0.1	58.4	71.9	58.2	40.7
			0.5	**60.8**	71.7	**60.4**	**47.2**
			0.9	54.7	64.4	54.4	42.5
Feature	Kinetics-LT	ResNet-50	0.1	58.1	71.4	58.1	40.7
			0.5	59.7	72.2	59.6	43.0
			0.9	58.2	71.5	57.6	41.7

Table 10: **Ablation studies of text descriptions.** "NO TEXT" denotes using no text descriptions for training, same as the CLIP-Close and CLIP -LT. "BASIC" denotes only using basic prompted sentences for training, and "FULL" denotes using both basic prompted sentences and crawled text descriptions for training.

(a) **Ablation studies of text descriptions on Kinetics-Close.**

Method	Backbone	Operation	Top-1	Top-5
CLIP-Close	ViT-B/16	NO TEXT	78.9	93.5
VLG		BASIC	78.9	94.8
VLG		FULL	**81.8**	**95.3**

(b) **Ablation studies of text descriptions on Kinetics-LT.**

Method	Backbone	Operation	Overall	Many	Medium	Few
CLIP-LT	ResNet-50	NO TEXT	53.4	70.3	53.3	31.3
VLG		BASIC	57.8	71.4	56.9	35.8
VLG		FULL	**60.8**	**71.7**	**60.4**	**47.2**

TEMPORAL MODULE. We investigate the effectiveness of the temporal module by using different numbers of layers in the temporal module. As shown in Table 8, the model achieves the highest recognition accuracy on Kinetics-Close with 6 transformer layers in the temporal module. An interesting phenomenon is that increasing the number of layers in the temporal module leads to a significant rise in accuracy performance at the beginning, but the accuracy falls when the temporal module has more than 6 layers. It may be attributed to the overfitting caused by using the transformer with too many layers. In the few-shot setting, the model achieves the highest recognition accuracy without the additional temporal module, since there are few videos to feed the data-hungry Transformer layers in the few-shot case.

DISTILLATION LOSS. To better investigate the effectiveness of distillation loss on reducing the risk of overfitting caused by limited text corpus. We conduct the ablation study on Kinetics-LT, which has fewer videos to avoid the influence of excessive visual information, and use ResNet-50 as the backbone. As shown in Table 9, our method with distillation loss achieves higher performance in medium-shot, few-shot and overall cases, compared to the one without distribution loss. It indicates that the distillation loss helps the model learn better video-language representation with limited data.

To further study the influence of distillation in the pre-training stage, we try to use the pre-trained CLIP model as the teacher model to distill the video and language encoder of our model at the feature level, in addition to the logits distillation. As shown in Table 9, both feature distillation and logits distillation with α of 0.5 can improve the performance in many-shot, medium-shot, few-shot and overall cases. And our method achieves the highest performance on Kinetics-LT when using logits distillation with the loss weight α of 0.5.

TEXT DESCRIPTIONS. We study the significance of our collected text descriptions by replacing them with "Using no sentences" operation and "Only using basic prompted sentences" operation, both in Kinetics-Close and Kinetics-LT. As shown in Table 10, using the extra class-wise text description crawled from Wiki and Wikihow can significantly improve the performance in both Kinetics-Close

Table 11: **Ablation studies of loss terms.** We investigate the effectiveness of the two terms in Eq. 4 by adopting three operations: "only P^V", "only P^T", and "both P^V and P^T".

(a) **Ablation studies of loss terms on Kinetics-Close.**

Method	Backbone	Operation	Top-1	Top-5
VLG	ViT-B/16	Only P^V	79.5	94.7
		Only P^T	80.7	95.1
		P^V and P^T	**81.8**	**95.3**

(b) **Ablation studies of loss terms on Kinetics-LT.**

Method	Backbone	Operation	Overall	Many	Medium	Few
VLG	ResNet-50	Only P^V	56.9	**73.2**	57.1	34.7
		Only P^T	59.8	67.2	60.2	**48.8**
		P^V and P^T	**60.8**	71.7	**60.4**	47.2

Table 12: **Ablation studies of different splitting strategies.** "RAND" denotes using the original splits in our Kinetics-LT, which are randomly chosen. "GOOGLE" denotes using the splits sorted by the number of entries in Google search.

Method	Backbone	Operation	Overall	Many	Medium	Few
VLG	ResNet-50	RAND	60.8	71.7	60.4	47.2
		GOOGLE	61.2	71.4	62.7	44.2

and Kinetics-LT. Specifically, in the few-shot classes of Kinetics-LT, one can observe that VLG with both basic prompted sentences and crawled text description gets absolute accuracy gain against VLG with only basic prompts and VLG without text descriptions (47.2% vs. 35.8% vs. 31.3%). It indicates the validity of text descriptions from the Internet, and effectiveness of leveraging abundant semantic knowledge to make up for the lack of video data.

Loss Terms in Stage II. In Eq. 4, the first term P^V is based on the video-only embedding E^V, and the second term P^T is based on the enhanced text embedding G. The first term adopts the MLP to obtain the classification probability, and the second term calculates the cosine similarity between the video-only embedding E^V and the enhanced text embedding G. To study the effectiveness of these two terms, we add experiments by adopting the first term, the second term, or both in the close set and long-tailed set.

It can be seen in Table 11 that in the close set, the model with both of the two terms performs better than the others, indicating the power of video-language representation when given abundant training data. In the long-tailed case, the model with only P^V performs well in the "Many" case but performs poorly in the "Few" case, while the model with only P^T performs well in the "Few" case but performs poorly in the "Many" case. By contrast, the model with both of the two terms serves as the trade-off without sacrificing too much performance for all cases, and further improves the overall accuracy for the long-tailed datasets. Therefore, we hold that both the two terms are necessary.

E Additional Experiments

Re-splitting the Classes. To demonstrate the rationality of label splitting in our Kinetics-LT, we adopt another strategy to re-split the classes according to their number of entries in Google search. As shown in the Table 12, there are no apparent changes in the recognition results, indicating the rationality of our label splitting in the long-tailed case.

Experiments on UCF-101 To further demonstrate the transferability of VLG, we also conduct experiments on UCF-101. We compare our proposed methods with prior methods on UCF-101, and it can be seen in Table 13 that our model can achieve significant performance on this dataset.

Table 13: **Results on UCF-101.** We report the average accuracy over three splits on UCF-101, CLIP* denotes that the model is initialized by the CLIP (Radford et al., 2021) weights. K400 and K600 are used to denote Kinetics400 and Kinetics600 respectively.

Method	Pretrain	Backbone	Frame	Views	Top-1	Top-5
TSN (Wang et al., 2016)	ImageNet+K400	ResNet-50	16	-	91.1	-
STM (Jiang et al., 2019)		ResNet-50	16	-	96.2	-
S3D-G (Xie et al., 2018)		-	-	-	96.8	-
FASTER32 (Zhu et al., 2020)		-	32	8×1	96.9	-
STAM-32 (Sharir et al., 2021)		ViT-B/16	32	-	97.0	-
R(2+1)D (Tran et al., 2018)		-	-	-	97.3	-
D3D (Stroud et al., 2020)	ImageNet+K600	ResNet-101-NL	50	-	97.1	-
VLG	CLIP*+K400	ViT-B/16	8	1×1	96.2	99.6
			16	1×1	96.3	99.7
			16	4×3	96.5	99.7
VLG	CLIP*+K400	ViT-L/14	8	1×1	97.3	99.8
			16	1×1	97.6	99.9
			16	4×3	**97.7**	**99.9**

F VISUALIZATION

F.1 VISUALIZATION OF PERFORMANCE

We use a radar chart to summarize the results across all regimes in Figure 5. The shape and area of the radar chat can serve as the total result to quantify the effectiveness and generalization ability of our method. We compare our method with current state-of-the-art methods in the radar chat, indicating the superiority of VLG over all settings.

F.2 CLASS-LEVEL PERFORMANCE IMPROVEMENT

In Figure 6, we visualize the class-level performance improvement on Kinetics-LT, which is measured by the absolute accuracy gains of our method against the baseline, both of which use ViT-B/16 as the visual backbone. We observe that there are more gains in the few-shot classes, indicating the introduced text descriptions can help mitigate the long-tailed problem.

F.3 VISUALIZATION OF TEXT CORPUS

In this section, we provide some visualization of the collected text corpus in Figure 7. It can be seen that these texts contain not only some noisy information within them, but also some static characteristics, dynamic evolution, and logical definition of the corresponding categories.

F.4 MORE EXAMPLES OF SALIENT SENTENCE

To intuitively demonstrate the effectiveness of our text selection ruler (TSR), we provide more sentences reserved or dropped by our TSR of different categories in Figure 8. We observe that our method can sample useful texts or filter out the useless ones.

G LIMITATION

Although our VLG achieves superior performance on multiple general video recognition settings, it still needs a two-stage training paradigm and cannot be end-to-end trained. To tackle this, we can apply reinforcement learning with reward functions (Lin et al., 2022; Meng et al., 2020) or gumbel-softmax tricks (Jang et al., 2016) to further improve the non-differentiable text selection parts. In addition, it might be difficult to crawl suitable descriptions of labels from Wiki or WikiHow for subtle actions, like "Put the glass on top of the table". Probably, it needs to participle phrases and crawl definitions from some dictionary websites as supplementary to improve the text descriptions.

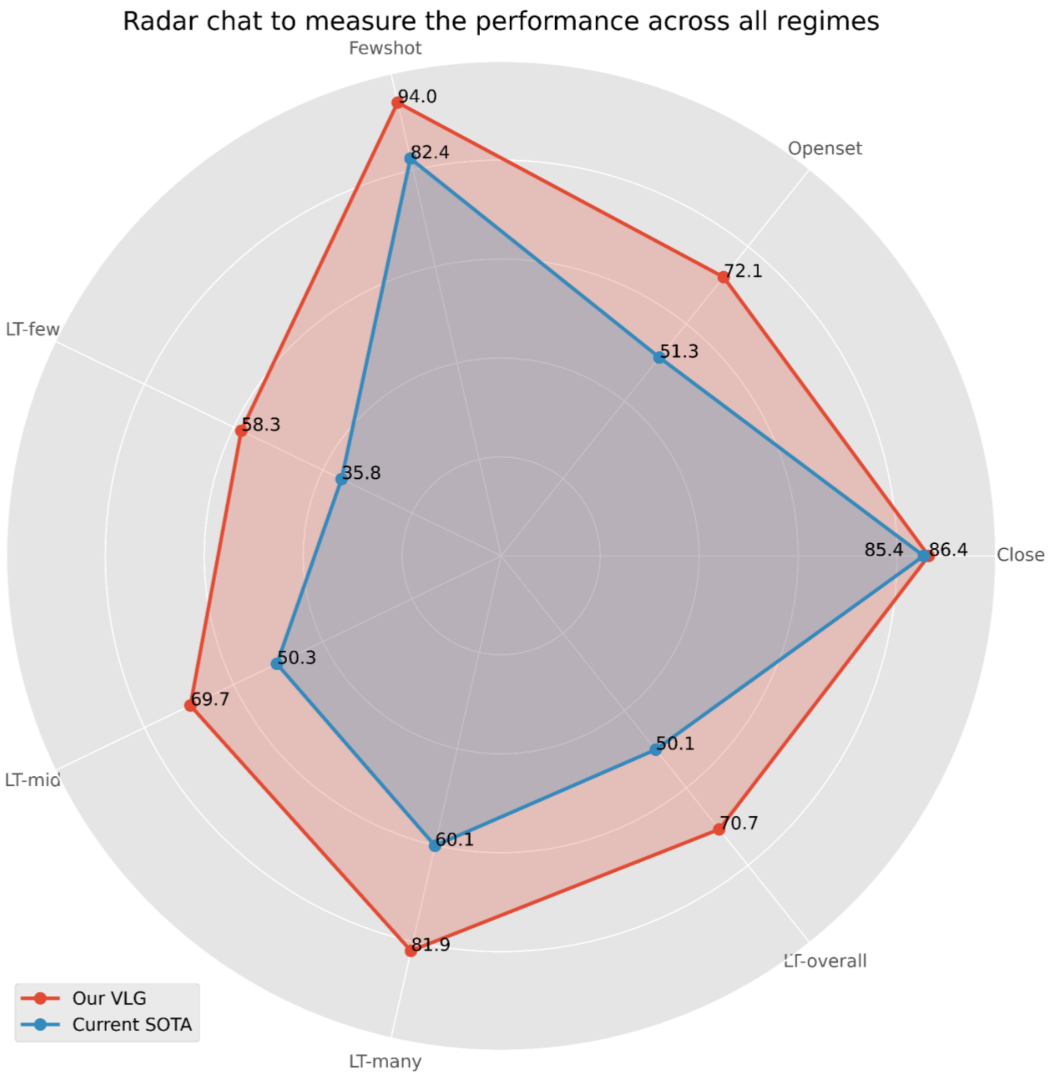


Figure 5: **Radar chat to measure the performance across all regimes.** It can be seen that our method outperforms current state-of-the-art methods for all settings.

H Ethic Statement

We use open datasets in our experiments following their licensing requirements. Our models may be subject to biases and other possible undesired mistakes, depending on how they are trained in reality. We didn't focus on potential negative impacts, because this work was not mainly designed for applications with potential negative impacts. As a recognition framework, it may be used for any related applications, just similar to many other general methods. But with proper usage, the proposed method could be beneficial to society.

I Reproducibility Statement

We report the necessary details to reproduce the experimental results in Sec. C, including network architectures, training hyperparameters, and dataset processing steps. We will also release our code in the future.

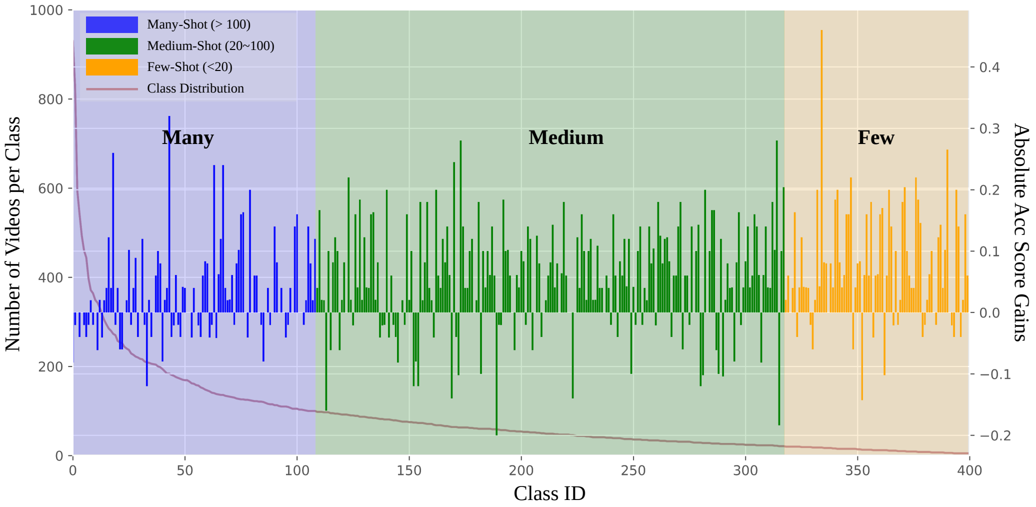
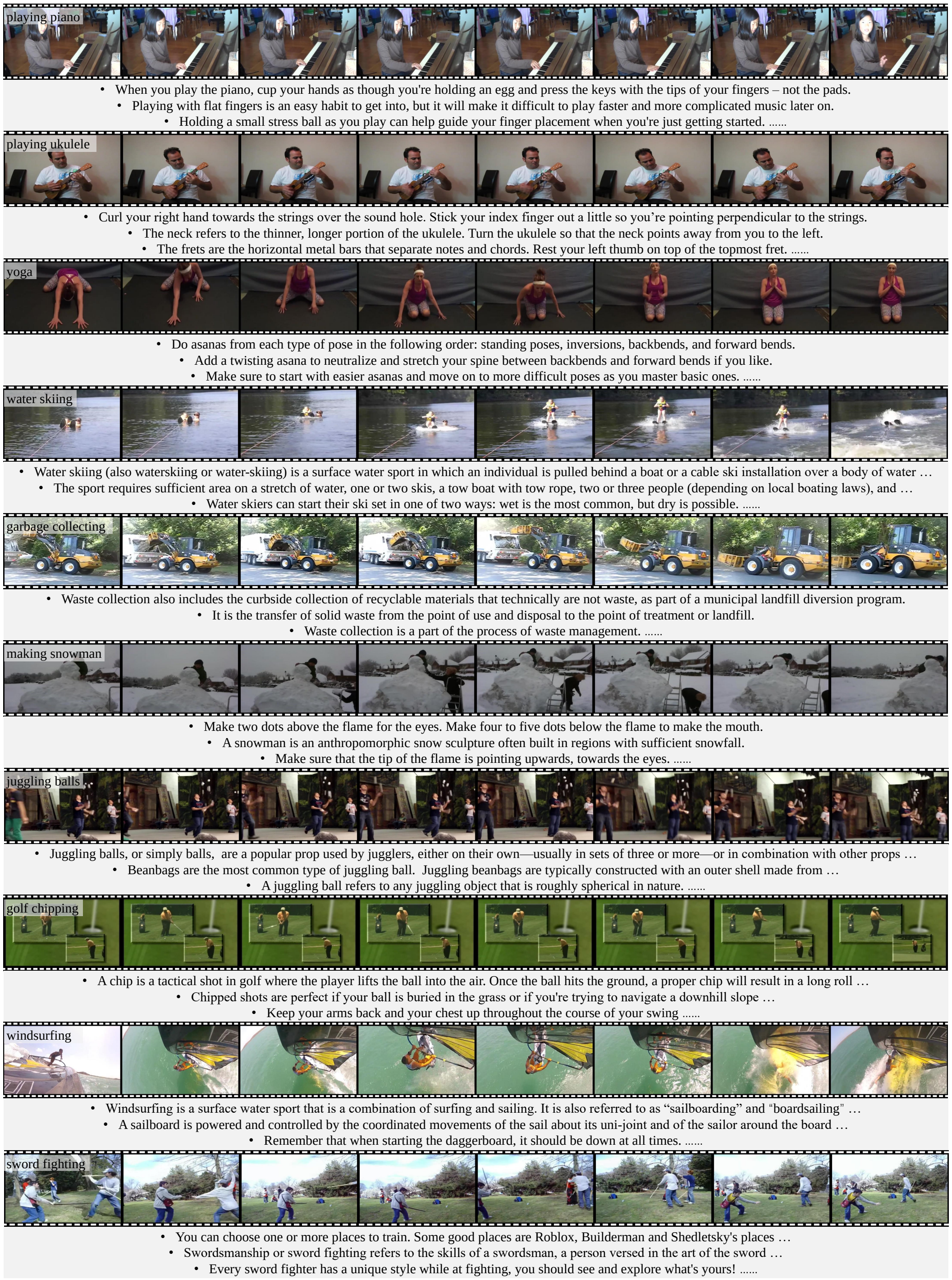


Figure 6: **Absolute accuracy score of our method over the baseline on Kinetics-LT.** Our method enjoys more performance gains in classes with fewer video samples.



- When you play the piano, cup your hands as though you're holding an egg and press the keys with the tips of your fingers – not the pads.
- Playing with flat fingers is an easy habit to get into, but it will make it difficult to play faster and more complicated music later on.
- Holding a small stress ball as you play can help guide your finger placement when you're just getting started. ……

- Curl your right hand towards the strings over the sound hole. Stick your index finger out a little so you're pointing perpendicular to the strings.
- The neck refers to the thinner, longer portion of the ukulele. Turn the ukulele so that the neck points away from you to the left.
- The frets are the horizontal metal bars that separate notes and chords. Rest your left thumb on top of the topmost fret. ……

- Do asanas from each type of pose in the following order: standing poses, inversions, backbends, and forward bends.
- Add a twisting asana to neutralize and stretch your spine between backbends and forward bends if you like.
- Make sure to start with easier asanas and move on to more difficult poses as you master basic ones. ……

- Water skiing (also waterskiing or water-skiing) is a surface water sport in which an individual is pulled behind a boat or a cable ski installation over a body of water …
- The sport requires sufficient area on a stretch of water, one or two skis, a tow boat with tow rope, two or three people (depending on local boating laws), and …
- Water skiers can start their ski set in one of two ways: wet is the most common, but dry is possible. ……

- Waste collection also includes the curbside collection of recyclable materials that technically are not waste, as part of a municipal landfill diversion program.
- It is the transfer of solid waste from the point of use and disposal to the point of treatment or landfill.
- Waste collection is a part of the process of waste management. ……

- Make two dots above the flame for the eyes. Make four to five dots below the flame to make the mouth.
- A snowman is an anthropomorphic snow sculpture often built in regions with sufficient snowfall.
- Make sure that the tip of the flame is pointing upwards, towards the eyes. ……

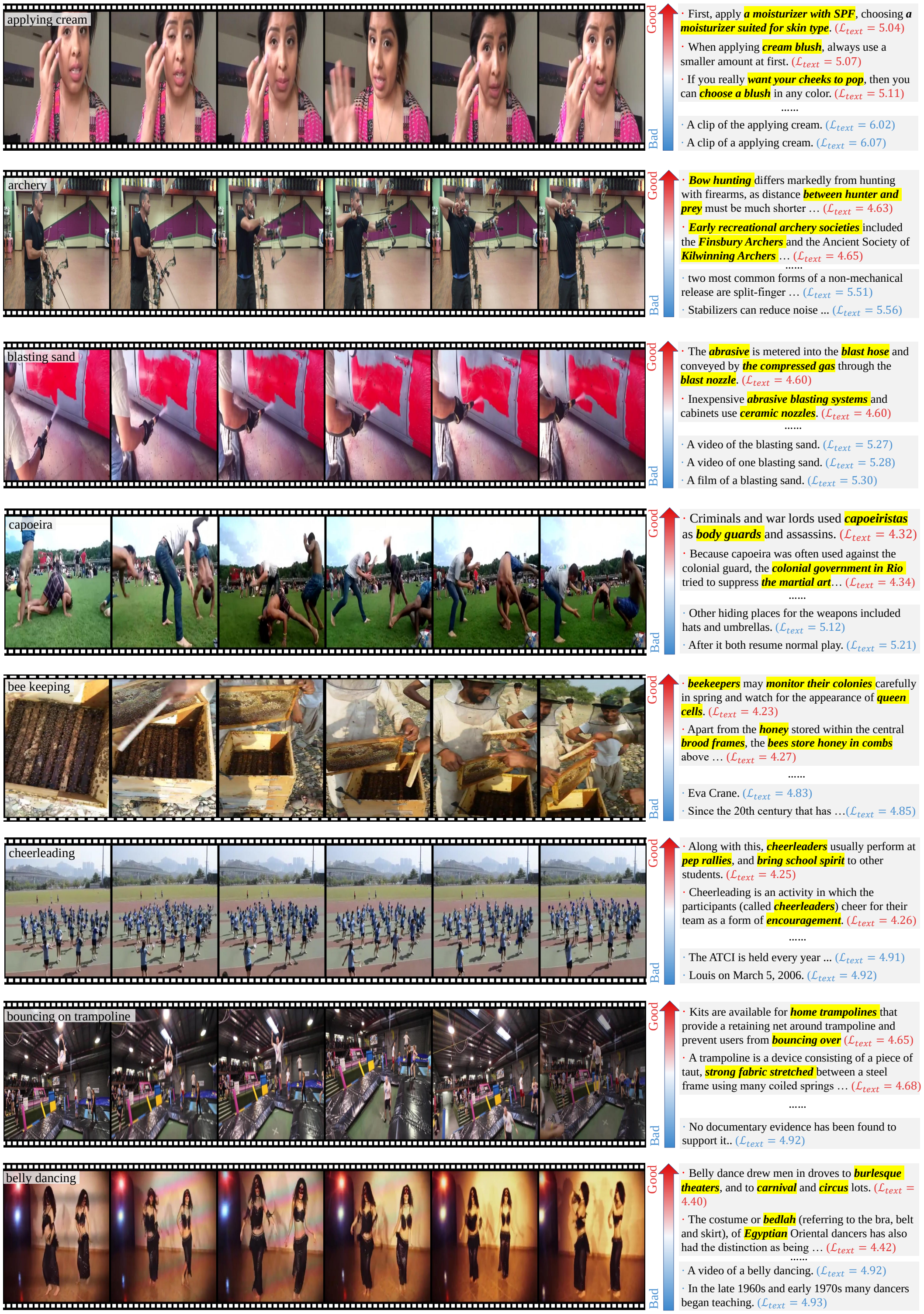
- Juggling balls, or simply balls, are a popular prop used by jugglers, either on their own—usually in sets of three or more—or in combination with other props …
- Beanbags are the most common type of juggling ball. Juggling beanbags are typically constructed with an outer shell made from …
- A juggling ball refers to any juggling object that is roughly spherical in nature. ……

- A chip is a tactical shot in golf where the player lifts the ball into the air. Once the ball hits the ground, a proper chip will result in a long roll …
- Chipped shots are perfect if your ball is buried in the grass or if you're trying to navigate a downhill slope …
- Keep your arms back and your chest up throughout the course of your swing ……

- Windsurfing is a surface water sport that is a combination of surfing and sailing. It is also referred to as "sailboarding" and "boardsailing" …
- A sailboard is powered and controlled by the coordinated movements of the sail about its uni-joint and of the sailor around the board …
- Remember that when starting the daggerboard, it should be down at all times. ……

- You can choose one or more places to train. Some good places are Roblox, Builderman and Shedletsky's places …
- Swordsmanship or sword fighting refers to the skills of a swordsman, a person versed in the art of the sword …
- Every sword fighter has a unique style while at fighting, you should see and explore what's yours! ……

Figure 7: **Examples of text descriptions crawled from Wikipedia and wikiHow for Kinetics400.** Both useful and redundant information can be found in these text corpus.



- First, apply ***a moisturizer with SPF***, choosing ***a moisturizer suited for skin type***. $(\mathcal{L}_{text} = 5.04)$
- When applying ***cream blush***, always use a smaller amount at first. $(\mathcal{L}_{text} = 5.07)$
- If you really ***want your cheeks to pop***, then you can ***choose a blush*** in any color. $(\mathcal{L}_{text} = 5.11)$

......

- A clip of the applying cream. $(\mathcal{L}_{text} = 6.02)$
- A clip of a applying cream. $(\mathcal{L}_{text} = 6.07)$

- ***Bow hunting*** differs markedly from hunting with firearms, as distance ***between hunter and prey*** must be much shorter … $(\mathcal{L}_{text} = 4.63)$
- ***Early recreational archery societies*** included the ***Finsbury Archers*** and the Ancient Society of ***Kilwinning Archers*** … $(\mathcal{L}_{text} = 4.65)$

......

- two most common forms of a non-mechanical release are split-finger … $(\mathcal{L}_{text} = 5.51)$
- Stabilizers can reduce noise ... $(\mathcal{L}_{text} = 5.56)$

- The ***abrasive*** is metered into the ***blast hose*** and conveyed by ***the compressed gas*** through the ***blast nozzle***. $(\mathcal{L}_{text} = 4.60)$
- Inexpensive ***abrasive blasting systems*** and cabinets use ***ceramic nozzles***. $(\mathcal{L}_{text} = 4.60)$

......

- A video of the blasting sand. $(\mathcal{L}_{text} = 5.27)$
- A video of one blasting sand. $(\mathcal{L}_{text} = 5.28)$
- A film of a blasting sand. $(\mathcal{L}_{text} = 5.30)$

- Criminals and war lords used ***capoeiristas*** as ***body guards*** and assassins. $(\mathcal{L}_{text} = 4.32)$
- Because capoeira was often used against the colonial guard, the ***colonial government in Rio*** tried to suppress ***the martial art***… $(\mathcal{L}_{text} = 4.34)$

......

- Other hiding places for the weapons included hats and umbrellas. $(\mathcal{L}_{text} = 5.12)$
- After it both resume normal play. $(\mathcal{L}_{text} = 5.21)$

- ***beekeepers*** may ***monitor their colonies*** carefully in spring and watch for the appearance of ***queen cells***. $(\mathcal{L}_{text} = 4.23)$
- Apart from the ***honey*** stored within the central ***brood frames***, the ***bees store honey in combs*** above … $(\mathcal{L}_{text} = 4.27)$

......

- Eva Crane. $(\mathcal{L}_{text} = 4.83)$
- Since the 20th century that has …$(\mathcal{L}_{text} = 4.85)$

- Along with this, ***cheerleaders*** usually perform at ***pep rallies***, and ***bring school spirit*** to other students. $(\mathcal{L}_{text} = 4.25)$
- Cheerleading is an activity in which the participants (called ***cheerleaders***) cheer for their team as a form of ***encouragement***. $(\mathcal{L}_{text} = 4.26)$

......

- The ATCI is held every year ... $(\mathcal{L}_{text} = 4.91)$
- Louis on March 5, 2006. $(\mathcal{L}_{text} = 4.92)$

- Kits are available for ***home trampolines*** that provide a retaining net around trampoline and prevent users from ***bouncing over*** $(\mathcal{L}_{text} = 4.65)$
- A trampoline is a device consisting of a piece of taut, ***strong fabric stretched*** between a steel frame using many coiled springs … $(\mathcal{L}_{text} = 4.68)$

......

- No documentary evidence has been found to support it.. $(\mathcal{L}_{text} = 4.92)$

- Belly dance drew men in droves to ***burlesque theaters***, and to ***carnival*** and ***circus*** lots. $(\mathcal{L}_{text} = 4.40)$
- The costume or ***bedlah*** (referring to the bra, belt and skirt), of ***Egyptian*** Oriental dancers has also had the distinction as being … $(\mathcal{L}_{text} = 4.42)$

......

- A video of a belly dancing. $(\mathcal{L}_{text} = 4.92)$
- In the late 1960s and early 1970s many dancers began teaching. $(\mathcal{L}_{text} = 4.93)$

Figure 8: **More visualization of text descriptions with corresponding $\mathcal{L}_{\text{text}}$.** The values of $\mathcal{L}_{\text{text}}$ reflect the saliency of these sentences, indicating the effectiveness of our proposed TSR.