# VTD-CLIP: Video-to-Text Discretization via Prompting CLIP

**Wencheng Zhu**[1,4], **Yuexin Wang**[1], **Hongxuan Li**[1], **Pengfei Zhu**[1,2,3*]
[1]School of Artificial Intelligence, Tianjin University
[2]Low-Altitude Intelligence Laboratory, Xiong'an National Innovation Center
[3]Xiong'an Guochuang Lantian Technology Co., Ltd.
[4]Haihe Laboratory of Information Technology Application Innovation
{wenchengzhu, wangyeuxin_207, lihongxuan, zhupengfei}@tju.edu.cn

## Abstract

Vision-language models bridge visual and linguistic understanding and have proven to be powerful for video recognition tasks. Existing methods primarily rely on parameter-efficient fine-tuning of pre-trained image-text models, suffering from limited interpretability and poor generalization due to inadequate temporal modeling. To address these, we propose a simple yet effective video-to-text discretization framework. Our approach leverages the frozen text encoder to build a visual codebook derived from video class labels, exploiting the many-to-one contrastive alignment between visual and textual embeddings in multimodal pretraining. This enables the transformation of temporal visual features into discrete textual tokens via feature lookups, yielding interpretable video representations through explicit video modeling. Then, to improve robustness against noisy or irrelevant frames, we introduce a confidence-aware fusion module that dynamically weights keyframes based on their semantic relevance, as measured by the codebook. Furthermore, we incorporate learnable text prompts to conduct adaptive codebook updates during training. Experiments on four datasets, including *HMDB-51*, *UCF-101*, *Something-Something-v2*, and *Kinetics-400*, validate the superiority of our approach, achieving competitive improvements over state-of-the-art approaches.

**Code** — https://github.com/isxinxin/VTD

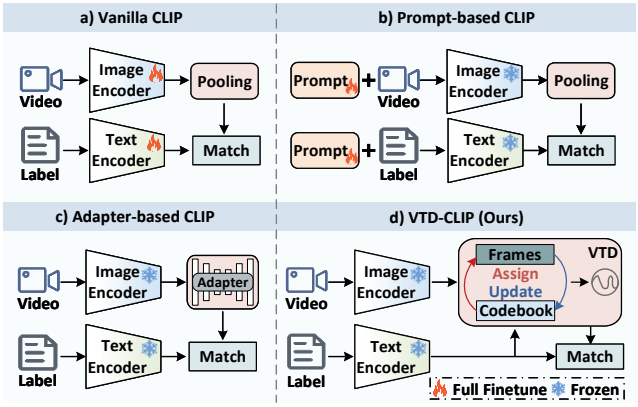


Figure 1: Comparisons with CLIP-based approaches. Complex temporal modeling in CLIP-based video methods remains questionable, as frame averaging often yields competitive performance, indicating that for frame-dominant semantics, naive temporal aggregation may be sufficient. Our method replaces temporal modeling with codebook-based discretization, transforming visual streams into discrete semantic events aligned with textual prototypes. By scoring frames and suppressing low-confidence ones, we enhance robustness to noise and improve interpretability.

## Introduction

Large-scale vision-language models, pre-trained on image-text pairs, have advanced visual-linguistic alignment, machine understanding, and human-like visual description generation (Gao et al. 2024; Li et al. 2023a; Zhang et al. 2024a). They have proven to be highly effective in diverse tasks, including image-text retrieval (Liu et al. 2023a), image caption (Nguyen et al. 2023), visual question answering (Yue et al. 2024; Cao et al. 2025), and multi-modal generation (Wang et al. 2023). Given these successes, there is growing interest in adapting such models to temporally structured video data.

While image-text alignment has achieved remarkable success, extending this paradigm to video-text alignment presents critical challenges (Ju et al. 2022; Li et al. 2025; Liu et al. 2023b). First, effective video-text alignment demands exponentially more paired training data than image-text pretraining, and this resource-intensive demand is impractical. Although parameter-efficient fine-tuning methods help reduce data dependency, they often sacrifice generalization by overfitting to limited task-specific data, thereby undermining the robust cross-modal alignment learned during large-scale pretraining (Zhang et al. 2021; Wu et al. 2023; Yang et al. 2023). This raises an important question: *How can we leverage image-text-aligned models for video understanding without compromising their generalization capabilities?*

An ideal framework should preserve the core architecture of pre-trained vision-language models, leveraging their inherent generalization capabilities (Jia et al. 2024) and maintaining the zero-shot performance derived from large-scale multimodal pretraining (Shi et al. 2024). In practice, this can be implemented through sparse keyframe summariza-

*Corresponding author

tion that select representative frames to encode the essential content of a video (Yu et al. 2024). This strategy avoids the need for large-scale video-text alignment pretraining while enabling efficient frame-level feature extraction (Lafon et al. 2024; Rasheed et al. 2023). Notably, recent studies show that simple temporal aggregation methods, such as average frame-level features, can achieve competitive performance against standard benchmarks (Cao et al. 2024; Wu et al. 2024a; Gupta et al. 2025). This suggests a frame-dominant nature of video semantics: in practice, a small subset of informative frames suffices to capture the overall meaning of a video (Rasheed et al. 2023; Korbar et al. 2025), reducing the necessity for complex temporal modeling (Gaintseva, Benning, and Slabaugh 2024; Chen et al. 2025). Therefore, we are determined to extend the image-text paradigm to video understanding by using keyframe summaries.

As we know, video semantic categories are often correlated with their primary content, motivating the use of pretrained textual categories as prototypes for video summarization and keyframe selection (Zhu et al. 2022). However, ground-truth labels are typically unavailable in advance, necessitating a pseudo-labeling mechanism to infer video semantics (Xue et al. 2023; Wu et al. 2024b). Leveraging the aligned image-text embedding space of CLIP (Gao et al. 2024), we classify each video frame into the most semantically similar textual category, effectively discretizing video content into textual pseudo-labels. This enhances interpretability by guiding keyframe selection toward frames with high semantic confidence while suppressing ambiguous or irrelevant ones (Wu et al. 2023). As illustrated in Figure 1, our framework avoids temporal modeling with heavy computational overhead, yet preserves strong interpretability and zero-shot generalization through frozen CLIP encoders.

In this work, we introduce VTD-CLIP, a video-to-text discretization framework that enhances video representations by mapping visual content into semantically meaningful, text-aligned tokens. Specifically, we treat the frozen text encoder of CLIP as a semantic codebook learner where predefined class-specific text embeddings serve as prototype tokens. For each frame, we extract its visual embedding and quantize it to the nearest text token via maximum similarity, yielding a discrete frame-level embedding. These frame-level embeddings are then aggregated into a global video-level representation through majority voting, capturing dominant semantic themes. We further compute a confidence score for each frame based on semantic alignment, and employ a confidence-aware fusion mechanism to combine the discrete video embedding with the original frame features.

Our contributions can be condensed into three aspects:

- We propose a simple yet effective framework that enhances video representation by quantizing visual content into discretized, text-aligned semantic embeddings.
- We use text embeddings as a semantic codebook, quantizing frames to nearest prototypes via nearest-neighbour lookup and aggregating discrete labels via voting.
- We evaluate the proposed method on four benchmark datasets, and extensive experiments demonstrate competitive performance against the state-of-the-art methods.

## Related Work

**Vision-Language Models.** Vision-language models have made great progress since the advent of CLIP (Radford et al. 2021). Given its strong zero-shot performance, recent efforts focus on efficiently fine-tuning CLIP for video analysis (Wang et al. 2024c,a). Existing methods can be roughly classified into two categories (Li et al. 2024), including prompt-based and adapter-based methods. Typical methods in the first category include ActionCLIP (Wang, Xing, and Liu 2021), ViFi-CLIP (Rasheed et al. 2023), and Vita-CLIP (Wasim et al. 2023). For example, Wang et al. (Wang, Xing, and Liu 2021) proposed a pre-train, prompt, and fine-tune paradigm. Rasheed et al. (Rasheed et al. 2023) fully fine-tuned CLIP encoders. Wasim et al. (Wasim et al. 2023) introduced multiple prompt tokens to CLIP encoders. For the second category, representative methods include XCLIP (Ni et al. 2022), VideoPrompt (Ju et al. 2022), and EVL (Lin et al. 2022). Ni et al. (Ni et al. 2022) employed cross-frame communication and multi-frame integration. Lin et al. (Lin et al. 2022) encoded temporal information via a lightweight Transformer. Wu et al. (Wu, Sun, and Ouyang 2023) employed a pre-trained language model to create semantic targets. Qing et al. (Qing et al. 2023) disentangled spatial and temporal information. Lin et al. (Lin et al. 2023) proposed an unsupervised approach with GPT-3. Kahatapitiya1 et al. (Kahatapitiya et al. 2024) prioritized text augmentation over visual knowledge. Chen et al. (Chen et al. 2024a) enhanced text knowledge to improve video generalizability.

**Discrete Representation Learning.** Discrete tokenizers are essential in vision-language models by bridging multimodal data into unified representations (Liu et al. 2022). For example, Van et al. (Van Den Oord, Vinyals et al. 2017) pioneered neural vector quantization for discrete latent space learning. Razavi et al. (Razavi, Van den Oord, and Vinyals 2019) extended this through a multi-level hierarchical VQ-VAE. Esser et al. (Esser, Rombach, and Ommer 2021) combined the inductive bias of CNNs and the expressive power of Transformers. Ramesh et al. (Ramesh et al. 2021) conducted cross-modal alignment via autoregressive joint token modeling. Bao et al. (Bao et al. 2022) predicted discrete visual tokens via mask image modeling. Discrete methods face codebook collapse, where expanding codebooks exhibit diminishing element diversity. While Mentzer et al. (Mentzer et al. 2024) employed a bounding function to round each feature channel into integers, we propose an alternative approach that utilizes the text encoder as a codebook learner and updates the codebook via learnable text prompts.

## Approach

As shown in Figure 2, our framework consists of three core modules: feature extraction via frozen image encoder $\phi_v(\cdot;\theta_v)$ and text encoder $\phi_t(\cdot;\theta_t)$ for frame features $\boldsymbol{x}$ and text features $\boldsymbol{c}$, video-to-text discretization for video discrete features $\boldsymbol{v}$, and confidence-aware fusion for video features $\hat{\boldsymbol{v}}$.

### Feature Extraction

Given an input video $\mathcal{V} = \{\boldsymbol{I}_t\}$ where $\boldsymbol{I}_t \in \mathbb{R}^{h\times w\times 3}$, our method divides $\mathcal{V}$ into $T$ uniform temporal segments

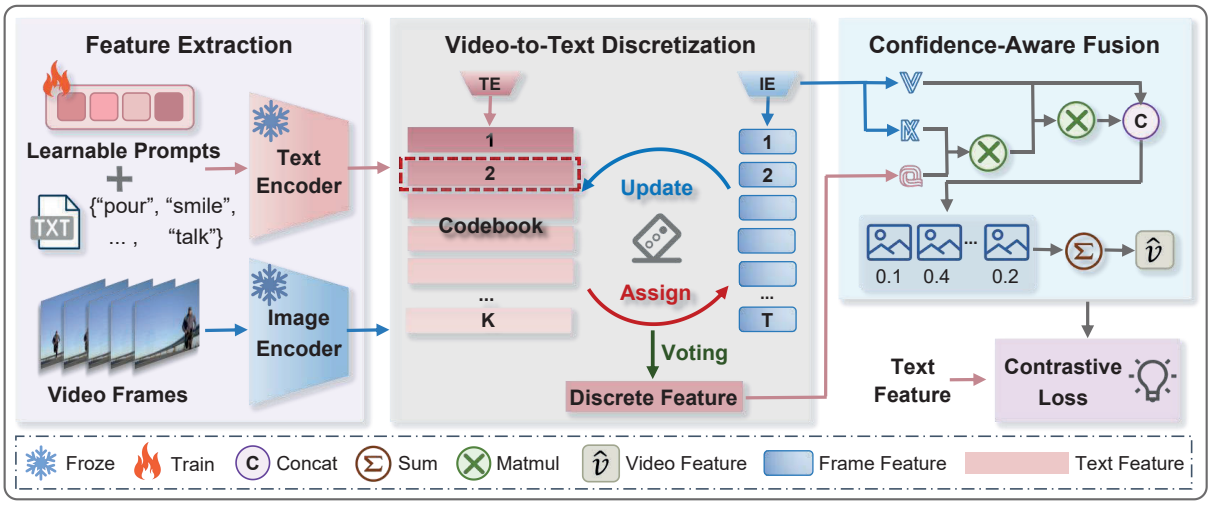


Figure 2: The architecture of VTD-CLIP. We first extract frame and text embeddings using pre-trained CLIP encoders and employ text embeddings to construct a visual codebook. Then, we obtain the discrete feature by discretizing visual embeddings through video-to-text discretization. Finally, we produce video features with confidence-aware fusion for recognition.

and randomly sample one frame from each segment. These frames are then fed into the image encoder $\phi_v(\cdot;\theta_v)$, which decomposes one frame into $\frac{h}{p} \times \frac{w}{p}$ non-overlapping $p \times p$ patches. Following previous methods (Jia et al. 2022; Zhang et al. 2023), we inject learnable visual prompts into the patch sequence to mitigate frame-text modality semantic gaps,

$$\boldsymbol{s}_t = [\boldsymbol{u}_1][\boldsymbol{u}_2]\cdots[\boldsymbol{u}_m][\boldsymbol{e}_t], \quad t \in [1, T], \tag{1}$$

where $\boldsymbol{s}_t$ is the $t$-th frame input, $\boldsymbol{e}_t$ is its total patches, and $\boldsymbol{u}_m$ is trainable prompt tokens with $m = 16$. We derive the $t$-th frame feature $\boldsymbol{x}_t \in \mathbb{R}^d$ as $\boldsymbol{x}_t = \phi_v(\boldsymbol{s}_t;\theta_v)$.

## Video-to-Text Discretization

By leveraging vision-language alignment from large-scale contrastive pre-training, we repurpose the text encoder as a visual codebook learner, where textual category embeddings serve as prototypes for video understanding.

**Text-Semantic Prototype.** We employ the frozen text encoder to extract textual embeddings, $\boldsymbol{c}_k = \phi_t(\boldsymbol{y}_k;\theta_t)$, where $\boldsymbol{y}_k$ denotes the text prompt of the $k$-th class, and $\boldsymbol{c}_k \in \mathbb{R}^d$ is the corresponding embedding. These category embeddings serve as semantic prototypes for identifying dominant content in videos. Subsequently, we construct the visual codebook by aggregating the prototype vectors into a codebook matrix $\mathcal{C} = \{\boldsymbol{c}_1, \boldsymbol{c}_2, ..., \boldsymbol{c}_K\} \in \mathbb{R}^{d\times K}$, where $K$ is the number of distinct categories in the dataset. Each video frame $\boldsymbol{x}_i$ is then quantized to the nearest codebook element,

$$\boldsymbol{x}_i^q = Q(\boldsymbol{x}_i) := \underset{\boldsymbol{c}_k \in \mathcal{C}}{\arg\min} \|\boldsymbol{x}_i - \boldsymbol{c}_k\|, \tag{2}$$

To be specific, our method aligns the codebook structure with the dataset's category set, naturally setting the codebook size K to the number of classes, eliminating the need for manual dimensionality tuning. Unlike the VQ-VAE codebooks, which are trained from scratch and prone to codebook collapse, our text-derived codebook $\mathcal{C}$ is initialized via the frozen text encoder and inherits diverse semantic structure. This strong semantic prior effectively prevents collapse and ensures stable and meaningful discretization.

**Codebook Adaptation with Prompts.** The text encoder receives a structured set of template tokens, e.g., "*a photo of a* $\{class\}$," where the $class$ token is replaced with the categorical label from the dataset. These prompts are projected into an embedding space using the frozen text encoder $\phi_t(\cdot;\theta_t)$. Following CoOp (Zhou et al. 2022), we construct an adaptive codebook that aligns visual semantics with prototypes during training while preserving cross-modal consistency through pre-trained language representations,

$$\boldsymbol{y}_k = [\boldsymbol{w}_1]\,[\boldsymbol{w}_2]\cdots[\boldsymbol{w}_n]\,[class_k], \quad k \in [1, K]. \tag{3}$$

We define $K \in \mathbb{N}^+$ as the number of categories. Let $\boldsymbol{w}_i$ denote the $i$-th learnable text prompt, which has the same dimensionality as the input tokens. Each prompt consists of 16 learnable tokens, initialized from a standard Gaussian distribution. While the text encoder produces both token-wise embeddings and a global [CLS] embedding, we only leverage the [CLS] embedding as the category-level textual representation. This yields $\mathcal{C}$, and each $\mathbf{c}_i \in \mathbb{R}^d$ corresponds to the [CLS] embedding of an $i$-th category-specific prompt.

Unlike conventional methods such as VQ-VAE, whose representations are fixed after training and necessitate complete re-optimization for domain shifts, our codebook is dynamic and adaptable via efficient prompt tuning.

**Hard Assignment via Nearest Neighbour.** Our method employs hard assignment, mapping each frame feature to its nearest textual prototype, instead of using soft weighting to generate frame-level pseudo-labels. This enforces clear boundaries between categories, reducing ambiguity between visually similar classes, e.g., "run" vs. "fast walk". Moreover, frames with high similarity to the target text prototypes are retained as meaningful content, while low-similarity seg-

ments, such as background or irrelevant motion, are automatically excluded. By focusing on high-confidence assignments, hard assignment adaptively identifies keyframes without requiring complex temporal modeling.

Formally, given frame features $\boldsymbol{x}_t$ and the text-derived codebook $\mathcal{C}$, we compute the cross-modal affinity matrix $\boldsymbol{S} \in \mathbb{R}^{T\times K}$, where each element $s_{tk} = \cos(\boldsymbol{x}_t, \boldsymbol{c}_k)$ denotes the cosine similarity between the $t$-th frame and the $k$-th textual prototype in the shared space. For coarse-grained frame-to-text alignment, we perform hard assignment by selecting the prototype with the maximum affinity score,

$$\hat{k}_t = \arg\max_{k\in\{1,2,\ldots,K\}} s_{tk}. \tag{4}$$

Hard assignment induces sparsity in the alignment matrix by retaining only the strongest semantic correspondence per frame. The resulting sparsity suppresses noisy or ambiguous mappings while maintaining interpretable pseudo-labels.

**Bag-of-Prototypes Video Category.** We compute semantic prototype activations through sparse correlation aggregation. Given the binary assignment mask $\boldsymbol{M} \in \{0,1\}^{T\times K}$, where $m_{t\hat{k}_t} = 1$ if the $t$-th frame is assigned to the $\hat{k}_t$-th text prototype. The video-level prototype is then defined as,

$$k_{max} = \arg\max_{k\in\{1,2,\ldots,K\}} \boldsymbol{m}_k \odot \boldsymbol{s}_k. \tag{5}$$

where $\boldsymbol{m}_k$ and $\boldsymbol{s}_k$ denote the $k$-th column of $\boldsymbol{M}$ and the affinity matrix $\boldsymbol{S}$ respectively. $\odot$ is the Hadamard product and $k_{max}$ corresponds to the index of the most activated prototype. The final video representation $\boldsymbol{v}$ inherits the semantic embedding of the most activated prototype as $\boldsymbol{v} = \boldsymbol{c}_{k_{max}}$.

## Confidence-Aware Fusion

Conventional methods typically employ simple average temporal pooling to fuse frame features. However, such approaches are prone to temporal redundancy from irrelevant frames. Recent work has explored frame filtering to improve temporal coherence. In contrast, we propose a confidence-aware fusion mechanism shown in Fig. (3) that dynamically weights each frame according to its text-aligned similarity.

To be specific, given the discrete video representation $\boldsymbol{v}$ from the video-to-text discretization module, we fuse it with sequential frame-level features as $\boldsymbol{f} = \text{CrossAttn}(\boldsymbol{v}, \boldsymbol{x}) + \boldsymbol{x}$, where the cross-attention layer enables the global semantic context in $\boldsymbol{v}$ to modulate $\boldsymbol{x}$, while the skip connection preserves spatio-temporal details. For each frame, we first extract its relevance score $\boldsymbol{s}_{tk_{max}}$ to the dominant text prototype, where $k_{max}$ is computed as in Eq.(5) and $\tau$ controls the sharpness of the weighting distribution. We then normalize the confidence scores across the frames using a temperature-scaled softmax and obtain the global video embedding by aggregating the confidence-weighted features as,

$$\hat{\boldsymbol{v}} = \sum_{t=1}^{T} \frac{\exp\left(\boldsymbol{s}_{tk_{max}}/\tau\right)}{\sum_{t=1}^{T}\exp\left(\boldsymbol{s}_{tk_{max}}/\tau\right)} \times \boldsymbol{f}_t, \quad \tau > 0. \tag{6}$$

We maximize cross-modal alignment between video representations and ground-truth text embeddings within a batch via the cross-entropy loss, leading to the following objective,

$$\mathcal{L} = -\sum_i \log \frac{\exp\left(\cos\left(\hat{\boldsymbol{v}}_i, c_i\right)/\tau\right)}{\sum_j \exp\left(\cos\left(\hat{\boldsymbol{v}}_i, c_j\right)/\tau\right)}. \tag{7}$$

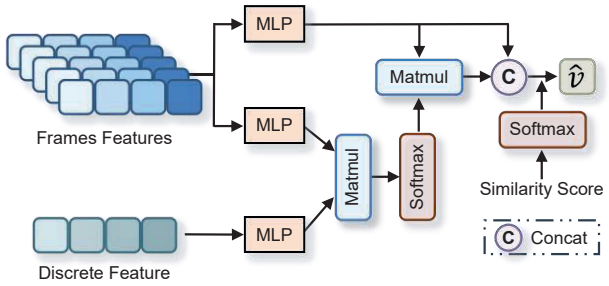


Figure 3: Illustration of confidence-aware fusion. The video-to-text discretization module provides the similarity score.

# Experiments

## Implementation

We conducted experiments on *Something-Something-v2 (SSv2)*, *Kinetics-400 (K-400)*, *HMDB-51* and *UCF-101*. Our method employed a frozen CLIP with a ViT-B/16 backbone for the text and visual encoders. We evaluated the effectiveness of our method through full supervision, base-to-noval, and few-shot experiments. We utilized 8 $\times$ NVIDIA A100 GPUs with per-GPU batch sizes of 8. We used the Adam optimizer with a weight decay of 0.001 and learning rates of 0.004 for *K-400* and 0.0004 for the other datasets.

## Main Results

**Fully-Supervised Video Recognition.** We perform fully-supervised experiments on *K-400* to validate the effectiveness of our method. As shown in Table 1, our method achieves competitive performance compared to state-of-the-art CLIP-based approaches. With only 194 GFLOPs, our method is significantly more computationally efficient than ViViT-H (17352 GFLOPs) and even lighter than XCLIP-B/16 (287 GFLOPs). While MoTED-B/16 achieves comparable performance by incorporating an additional large language model, it comes at the cost of increased complexity and resource demands. By mapping visual features to a text-aligned space, VTD-CLIP provides a lightweight alternative, well-suited for applications where model size, inference cost, and training scalability are critical.

**Base-to-Novel Video Recognition.** To evaluate the generalization capability, we establish a non-overlapping split between base and novel categories, ensuring that $\mathsf{C}_{base} \cap \mathsf{C}_{novel} = \emptyset$. For each dataset, we randomly divide the categories into two equal groups: 50% for base categories and the remaining 50% for novel categories. Models are trained exclusively on base categories and tested on novel ones. During evaluation on novel categories, the semantic codebook is constructed non-parametrically using the fixed text prompt template: "a photo of a {class}", without any further training. As listed in Table 2, our method outperforms comparison methods in both base and novel settings. Notably, it achieves substantial improvements of 6.0%, 3.2%, 1.1%, and 0.5% in novel category recognition over the other methods. These results indicate that aligning visual features with text-guided prototypes helps mitigate overfitting to base categories and enables effective semantic generalization.

| Method | Pretrain | Frames | Views | Top-1 | Top-5 | Avg | GFLOPs | FFT |
|---|---|---|---|---|---|---|---|---|
| *Large-scale Image Pretraining* | | | | | | | | |
| Uniformer-B (Li et al. 2023b) | IN-1k | 32 | 4×3 | 83.0 | 95.4 | 89.2 | 259 | ✓ |
| Swin-B (Liu et al. 2021) | IN-1K | 32 | 4×3 | 80.6 | 94.6 | 87.6 | 590 | ✓ |
| ViViT-H (Arnab et al. 2021) | JFT-300M | 32 | 4×3 | **84.8** | **95.8** | **90.3** | 17352 | ✓ |
| MViTv2-B (Li et al. 2022) | ✗ | 32 | 5×1 | 82.9 | 95.7 | 89.3 | 225 | ✓ |
| *Unimodal Visual Pretraining from CLIP* | | | | | | | | |
| ActionCLIP-B/16 (Wang et al.2021) | CLIP-400M | 32 | 10×3 | 83.8 | 96.2 | 90.0 | 563 | ✓ |
| XCLIP-B/16 (Ni et al. 2022) | CLIP-400M | 16 | 4×3 | **84.7** | **96.8** | **90.8** | 287 | ✓ |
| ViFi-CLIP-B/16 (Rasheed et al. 2023) | CLIP-400M | 16 | 4×3 | 83.9 | 96.3 | 90.1 | 281 | ✓ |
| VideoPrompt-B/16 (Ju et al. 2022) | CLIP-400M | 16 | - | 76.9 | 93.5 | 85.2 | - | ✗ |
| EVL-B/16 (Lin et al. 2022) | CLIP-400M | 16 | 1×3 | 83.6 | - | - | 888 | ✗ |
| ST-Adapter-B/16 (Pan et al. 2022) | CLIP-400M | 32 | 1×3 | 82.7 | 96.2 | 89.5 | 607 | ✗ |
| *Mutimodal Visual Pretraining from CLIP* | | | | | | | | |
| STAN-conv-B/16 (Liu et al. 2023b) | CLIP-400M | 8 | 1×3 | 83.1 | 96.0 | 89.6 | 238 | ✗ |
| ILA-B/16 (Tu et al. 2023) | CLIP-400M | 8 | 4×3 | 84.0 | 96.6 | 90.3 | 149 | ✗ |
| MoTED-B/16 (Zhang et al. 2024b) | CLIP-400M | 8 | 1x3 | **85.1** | 97.0 | 91.0 | 180 | ✗ |
| DiST-B/16 (Qing et al. 2023) | CLIP-400M | 16 | 1x3 | 84.4 | 96.7 | 90.6 | 320 | ✗ |
| ALT-B/16 (Chen et al. 2024b) | CLIP-400M | 16 | 1x3 | 84.8 | 96.4 | 90.4 | 657 | ✓ |
| Vita-CLIP-B/16 (Wasim et al. 2023) | CLIP-400M | 16 | 4×3 | 82.9 | 96.3 | 89.6 | 190 | ✗ |
| M2-CLIP-B/16 (Wang et al. 2024b) | CLIP-400M | 16 | 4×3 | 83.7 | 96.7 | 89.6 | 422 | ✗ |
| ViLT-CLIP (Wang et al. 2024a) | CLIP-400M | 16 | 4×3 | 77.6 | 94.5 | 86.1 | 287 | ✗ |
| FocusVideo-B/16 (Wang et al. 2025) | CLIP-400M | 32 | 4×3 | 84.7 | 96.8 | 90.8 | 816 | ✗ |
| **VTD-CLIP-B/16 (Ours)** | CLIP-400M | 16 | 4×3 | **85.1** | **97.1** | **91.1** | 194 | ✗ |

Table 1: Fully-supervised performance (%) on *K-400*. We classify comparison methods into three branches and describe our experimental setup, including pre-trained datasets, sampling frames, inference strategies, and model fully fine-tuning (FFT). We evaluate performance using Top-1 and Top-5 accuracies, their average, and GFLOPs.

| Method | HMDB-51 | | | UCF-101 | | | SSv2 | | | K-400 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Base | Novel | HM | Base | Novel | HM | Base | Novel | HM | Base | Novel | HM |
| Vanilla CLIP (Radford et al. 2021) | 53.3 | 46.8 | 49.8 | 78.5 | 63.6 | 70.3 | 4.9 | 5.3 | 5.1 | 62.3 | 53.4 | 57.5 |
| ActionCLIP (Wang et al.2021) | 69.1 | 37.3 | 48.4 | 90.1 | 58.1 | 70.6 | 13.3 | 10.1 | 11.5 | 61.0 | 46.2 | 52.6 |
| XCLIP (Ni et al. 2022) | 69.4 | 45.5 | 55.0 | 89.9 | 58.9 | 71.2 | 8.5 | 6.6 | 7.4 | 74.1 | 56.4 | 64.0 |
| VideoPrompt (Ju et al. 2022) | 46.2 | 16.0 | 23.8 | 90.5 | 40.4 | 55.9 | 8.3 | 5.3 | 6.5 | 69.7 | 37.6 | 48.8 |
| ViFi-CLIP (Rasheed et al. 2023) | 73.8 | 53.3 | 61.9 | 92.9 | 67.7 | 78.3 | 16.2 | 12.1 | 13.9 | 76.4 | 61.1 | 67.9 |
| ViLT-CLIP (Wang et al. 2024a) | 76.7 | 57.5 | 65.7 | 95.2 | 70.5 | 81.0 | 17.3 | 12.8 | 14.7 | 77.4 | 63.0 | 69.5 |
| **VTD-CLIP (Ours)** | **78.4** | **63.5** | **70.2** | **95.5** | **73.7** | **83.2** | **17.8** | **13.9** | **15.6** | **78.5** | **63.5** | **70.2** |

Table 2: Base-to-novel performance (%) on *HMDB-51*, *UCF-101*, *SSv2*, and *K-400*, where *Base* refers to half of the video categories randomly chosen for training, whereas *Novel* consists of the remaining categories for testing. *HM* denotes harmonic mean, balancing the performance between base and novel classes.

**Few-Shot Video Recognition.** To evaluate the effectiveness and generalization under limited data conditions, we conduct experiments across varying few-shot settings. As shown in Table 4, the accuracy of our method increases steadily with the number of training samples. While not superior in all settings, our method achieves significant gains in the extreme low-data setting, improving by +8.5% on *HMDB-51* and +2.9% on *UCF-101* in the 2-shot setting. This demonstrates that leveraging supplementary discrete text features through the dynamic codebook enhances generalization performance and reliance on large-scale visual data.

| Method | Vanilla CLIP *w/* LP | VTD-CLIP |
|---|---|---|
| Pooling | 82.2 | 85.3 |
| RNN | 75.2 | 78.2 |
| LSTM | 80.4 | 83.4 |
| Seq Transformer | 81.3 | 84.8 |
| **CAF (Ours)** | **84.9** | **87.6** |

Table 3: Performance comparison of different temporal fusion mechanisms (%). *w/* LP means a model with learnable prompts, and CAF means confidence-aware fusion module.

## Ablation Studies

**Temporal Fusion Mechanism Analysis.** We conduct ablation studies to evaluate the impact of different temporal fusion strategies, including RNN (Jain et al. 2016), LSTM (Greff et al. 2016), and Seq Transformer (Dong, Xu, and Xu 2018) with a 4-layer architecture for fair comparison. Table 3 reports results under the 4-shot learning on *UCF-101*. Our method outperforms all temporal models by leveraging text-aligned semantic features fused with discretized visual

| Method | HMDB-51 | | | | UCF-101 | | | | SSv2 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | N=2 | N=4 | N=8 | N=16 | N=2 | N=4 | N=8 | N=16 | N=2 | N=4 | N=8 | N=16 |
| Vanilla CLIP (Radford et al. 2021) | 41.9 | 41.9 | 41.9 | 41.9 | 63.6 | 63.6 | 63.6 | 63.6 | 2.7 | 2.7 | 2.7 | 2.7 |
| ActionCLIP (Wang et al.2021) | 47.5 | 57.9 | 57.3 | 59.1 | 70.6 | 71.5 | 73.0 | 91.4 | 4.1 | 5.8 | 8.4 | 11.1 |
| XCLIP (Ni et al. 2022) | 53.0 | 57.3 | 62.8 | 64.0 | 48.5 | 75.6 | 83.7 | 91.4 | 3.9 | 4.5 | 6.8 | 10.0 |
| VideoPrompt (Ju et al. 2022) | 39.7 | 50.7 | 56.0 | 62.4 | 71.4 | 79.9 | 85.7 | 89.9 | 4.4 | 5.1 | 6.1 | 9.7 |
| ViFi-CLIP (Rasheed et al. 2023) | 57.2 | 62.7 | 64.5 | 66.8 | 80.7 | 85.1 | 90.0 | 92.7 | 6.2 | 7.4 | 8.5 | 12.4 |
| ViLT-CLIP (Wang et al. 2024a) | 60.6 | 61.9 | 66.9 | 69.6 | 85.3 | **90.0** | 91.3 | 93.8 | **7.8** | **9.4** | 10.3 | 13.2 |
| OST (Chen et al. 2024a) | 59.1 | 62.9 | 64.9 | 68.9 | 82.5 | 87.5 | **91.7** | **93.9** | 7.0 | 7.7 | 8.9 | 12.2 |
| **VTD-CLIP (Ours)** | **67.6** | **68.7** | **69.7** | **75.7** | **85.4** | 87.6 | 91.3 | 93.0 | 7.1 | 8.7 | **10.4** | **13.4** |

Table 4: Few-shot performance (%) on *HMDB-51*, *UCF-101*, and *SSv2*. We conducted few-shot experiments by using 2, 4, 8, and 16 video sequences for each category, respectively.

| Method | Top-1 Accuracy (4-shot) | |
|---|---|---|
| | HMDB-51 | UCF-101 |
| Vanilla CLIP | 41.9 | 63.6 |
| Vanilla CLIP *w/* LP | 62.4 | 82.2 |
| VTD-CLIP *w/o* LP | 66.3 | 85.1 |
| VTD-CLIP *w/o* VTD | 66.6 | 85.3 |
| VTD-CLIP *w/o* CAF | 65.8 | 84.9 |
| **VTD-CLIP (Ours)** | **67.6** | **87.6** |

Table 5: Ablation study on different components (%). LP, VTD, and CAF represent the learnable prompt, video-to-text discretization, and confidence-aware fusion, respectively.

| Text Prompt | Top-1 Accuracy (4-shot) | |
|---|---|---|
| | HMDB-51 | UCF-101 |
| Learnable prompt + {*class*} | **67.6** | **87.6** |
| "*a photo of a* " + {*class*} | 66.3 | 85.1 |
| {*class*} | 64.8 | 83.5 |

Table 6: Ablation study on learnable, fixed, and no prompts (%), related to dynamic and static codebooks.

| Method | Top-1 Accuracy (4-shot) | |
|---|---|---|
| | Class | GPT-generated |
| Vanilla CLIP | 63.6 | 71.6 |
| Vanilla CLIP *w/* LP | 84.9 | 85.4 |
| **VTD-CLIP (Ours)** | **87.6** | **86.5** |

Table 7: Ablation study on codebooks with GPT descriptions (%). GPT extends text descriptions related to video labels. *w/* LP means that learnable prompts are adopted.

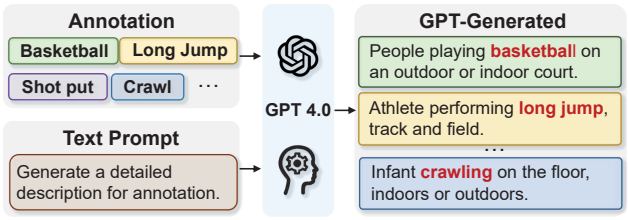


Figure 4: An example of GPT-generated descriptions.

representations. Notably, simple average pooling achieves the second-best performance, proving the effectiveness of frame-level alignment in reducing the need for complex temporal modeling. In contrast, RNNs underperform due to their limited capacity in capturing long-range temporal dynamics.

**Component Analysis.** To assess contributions of individual components, we conduct ablation studies on *HMDB-51* and *UCF-101*. As shown in Table 5, learning prompts, video-to-text discretization, and confidence fusion jointly enhance model performance. Learning prompts fine-tune the visual codebook. VTD generates text-aligned features, enabling CAF to compute more reliable confidence scores. In turn, CAF refines the frame-level representations by emphasizing semantically salient segments, which strengthens the alignment between visual content and textual prototypes in VTD.

**Prompt Analysis.** To investigate the impact of codebook design, we compare three variants: 1) a dynamic codebook with learnable prompts optimized via backpropagation, 2) a static codebook using fixed templates, and 3) a codebook with raw category labels. Table 6 shows that the dynamic codebook with learnable prompt achieves the highest accuracy, as it allows adaptive refinement of textual representations to better align semantic priors with visual content. Raw labels lack contextual information, while fixed templates are inflexible and unable to adapt to temporal visual patterns.

**Codebook Enhancement Analysis.** We investigate the impact of codebook enhancement using GPT 4.0 to generate category descriptions, shown in Figure 4. These descriptions provide richer semantic context. Table 7 shows that GPT-generated prompts can improve performance with additional contextual detail when base visual semantics are limited. For strong pre-trained models with well-aligned visual and textual representations, GPT descriptions may disrupt the learned cross-modal alignment, leading to marginal gains.

**Backbone Analysis.** Since our approach is based on the CLIP architecture, it can be adapted to different vision transformer backbones. We evaluate our method using ViT-B/16, ViT-B/32, and ViT-L/16. As shown in Table 9, performance improves consistently with stronger backbones, revealing a clear positive scaling trend. This is because larger models provide greater representational capacity, enabling effective cross-modal alignment and robust video understanding.

| Aggregation Strategy | Top-1 Accuracy (4-shot) | |
|---|---|---|
| | HMDB-51 | UCF-101 |
| Visual features | 66.5 | 85.9 |
| Discrete features | 57.2 | 80.0 |
| Discrete + Visual features | **67.6** | **87.6** |

Table 8: Ablation study on feature aggregation (%). Visual features, Discrete features, and Discrete + Visual features denote aggregation by using Visual features, discrete features, and these two integrated features, respectively.

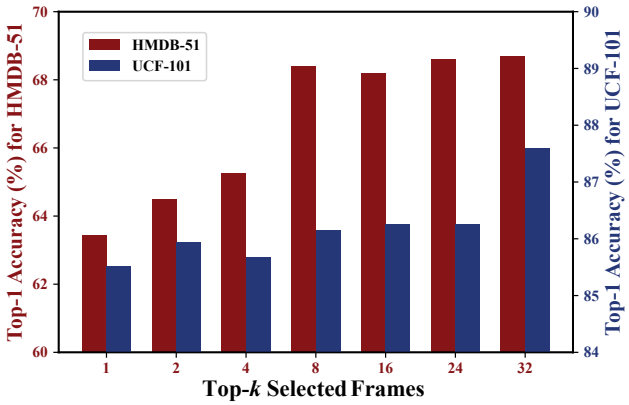


Figure 5: Ablation study on the different numbers of selected frames in the confidence-aware fusion module.

| Backbone | Top-1 Accuracy (4-shot) | | |
|---|---|---|---|
| | HMDB-51 | UCF-101 | SSv2 |
| CLIP-B/32 | 65.4 | 85.2 | 7.5 |
| CLIP-B/16 | 68.7 | 87.6 | 8.7 |
| CLIP-L/16 | **72.3** | **91.2** | **9.8** |

Table 9: Ablation study on different CLIP backbones.

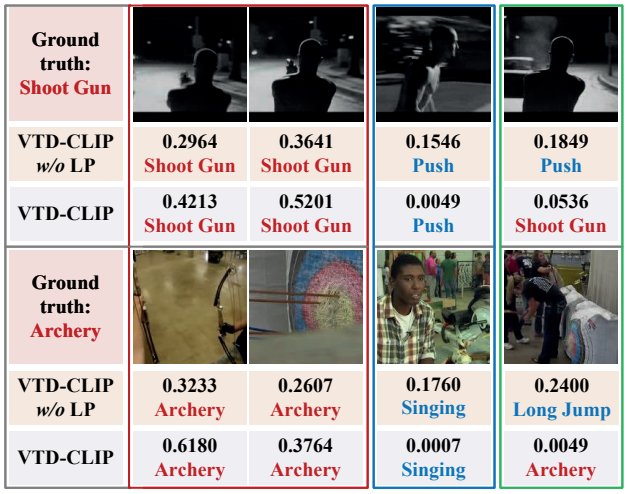


Figure 6: Visualization results of the VTD-CLIP. LP (learnable prompts): " *w/o* LP " denotes the VTD-CLIP employing a non-adaptive static codebook, while our method utilizes an adaptive dynamic codebook with prompt updates.

**Feature Aggregation Analysis.** To investigate the impact of different feature types, we conduct ablation studies on feature aggregation strategies. We evaluate three variants for confidence-aware fusion: visual features, discrete features, and fused features combining both. As shown in Table 8, the dual-stream aggregation strategy combining discrete and visual features achieves the best performance. Discrete features, while semantically meaningful, lack fine-grained spatial information and perform significantly worse than frame features. Visual features alone risks misalignment with text descriptions due to noise or irrelevant content, leading to suboptimal cross-modal matching. Fusing both, our method yields more robust and generalizable video representations.

**Frame Number Analysis.** To evaluate the impact of the number of frames used in the confidence-aware fusion module, we conduct ablation studies under 4-shot settings. We select the top-$k$ frames with the highest confidence scores for fusion. As shown in Figure 5, using too few frames leads to performance degradation, as important content may be missed. As $k$ increases, performance improves and eventually stabilizes, Further increasing $k$ yields only marginal gains by including lower-confidence or redundant frames.

## Visualization

We present qualitative results of VTD-CLIP by visualizing frame-level assignments and confidence scores using both dynamic and static codebooks in Figure 6. For each video, we randomly sample four frames. Compared to the static codebook, the dynamic codebook achieves tighter alignment between video frames and text features while exhibiting stronger discriminative power at the frame level. Moreover, with the dynamic codebook, our confidence-aware fusion strategy effectively assigns higher weights to keyframes (red boxes), while adaptively down-weighting frames that are either misaligned (blue boxes) or semantically redundant despite correct classification (green boxes). The integration of a dynamic codebook with video-to-text discretization and confidence-aware fusion enhances recognition accuracy and suppresses feature redundancy through selective weights.

## Conclusion

In this paper, we have proposed a simple yet effective video-to-text discretization framework for video understanding. We reformulate the text encoder as a trainable visual codebook learner, in which learnable prompts enable adaptive codebook updates. Then, we discretize frame features into textual prototypes and obtain discrete video features through confidence scoring. Finally, we integrate discrete video and frame features for confidence-aware fusion and recognition. Experimental results demonstrate that our method achieves competitive results against state-of-the-art methods.

**Limitations:** Currently, we only explore keyframe-based video summaries to adapt image-text models, but temporal modeling is essential for tasks like interactions or motions yet. In future work, we will design a flexible framework to utilize video shots to address dynamic visual cues.

## Acknowledgments

This work was sponsored by the National Natural Science Foundation of China (No.s 62222608, 62436002, 62406221), the Tianjin Natural Science Funds for Distinguished YoungScholar (No.23JCJQJC00270), the Natural Science Foundation of Tianjin (No.25JCQNJC00770), the National Key Research and Development Program of China under Grant 2025YFA0921700, the Zhejiang Provincial Natural Science Foundation of China (No.LD24F020004).

## References

Arnab, A.; Dehghani, M.; Heigold, G.; Sun, C.; Lučić, M.; and Schmid, C. 2021. Vivit: A video vision transformer. In *ICCV*, 6836–6846.

Bao, H.; Dong, L.; Piao, S.; and Wei, F. 2022. Beit: Bert pre-training of image transformers. In *ICLR*, 1–18.

Cao, C.; Zhang, Y.; Yu, Y.; Lv, Q.; Min, L.; and Zhang, Y. 2024. Task-Adapter: Task-specific Adaptation of Image Models for Few-shot Action Recognition. In *ACM MM*, 9038–9047.

Cao, Y.; Zhang, J.; Frittoli, L.; Cheng, Y.; Shen, W.; and Boracchi, G. 2025. Adaclip: Adapting clip with hybrid learnable prompts for zero-shot anomaly detection. In *ECCV*, 55–72.

Chen, T.; Yu, H.; Yang, Z.; et al. 2024a. OST: Refining Text Knowledge with Optimal Spatio-Temporal Descriptor for General Video Recognition. In *CVPR*, 18888–18898.

Chen, Y.; Chen, D.; Liu, R.; et al. 2024b. Align before adapt: Leveraging entity-to-region alignments for generalizable video action recognition. In *CVPR*, 18688–18698.

Chen, Y.; Li, K.; Bao, W.; et al. 2025. Learning to Localize Actions in Instructional Videos with LLM-Based Multi-Pathway Text-Video Alignment. In *ECCV*, 193–210.

Dong, L.; Xu, S.; and Xu, B. 2018. Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition. In *ICASSP*, 5884–5888.

Esser, P.; Rombach, R.; and Ommer, B. 2021. Taming transformers for high-resolution image synthesis. In *CVPR*, 12873–12883.

Gaintseva, T.; Benning, M.; and Slabaugh, G. 2024. RAVE: Residual Vector Embedding for CLIP-Guided Backlit Image Enhancement. In *ECCV*, 412–128.

Gao, P.; Geng, S.; Zhang, R.; Ma, T.; Fang, R.; Zhang, Y.; Li, H.; and Qiao, Y. 2024. Clip-adapter: Better vision-language models with feature adapters. *IJCV*, 132(2): 581–595.

Greff, K.; Srivastava, R. K.; Koutník, J.; Steunebrink, B. R.; and Schmidhuber, J. 2016. LSTM: A search space odyssey. *TNNLS*, 28(10): 2222–2232.

Gupta, R.; Rizve, M. N.; Unnikrishnan, J.; Tawari, A.; Tran, S.; Shah, M.; Yao, B.; and Chilimbi, T. 2025. Open Vocabulary Multi-Label Video Classification. In *ECCV*, 276–293.

Jain, A.; Zamir, A. R.; Savarese, S.; and Saxena, A. 2016. Structural-rnn: Deep learning on spatio-temporal graphs. In *CVPR*, 5308–5317.

Jia, C.; Luo, M.; Chang, X.; et al. 2024. Generating action-conditioned prompts for open-vocabulary video action recognition. In *ACM MM*, 4640–4649.

Jia, M.; Tang, L.; Chen, B.-C.; Cardie, C.; Belongie, S.; Hariharan, B.; and Lim, S.-N. 2022. Visual prompt tuning. In *ECCV*, 709–727.

Ju, C.; Han, T.; Zheng, K.; Zhang, Y.; and Xie, W. 2022. Prompting visual-language models for efficient video understanding. In *ECCV*, 105–124.

Kahatapitiya, K.; Arnab, A.; Nagrani, A.; and Ryoo, M. S. 2024. Victr: Video-conditioned text representations for activity recognition. In *CVPR*, 18547–18558.

Korbar, B.; Xian, Y.; Tonioni, A.; Zisserman, A.; and Tombari, F. 2025. Text-conditioned resampler for long form video understanding. In *ECCV*, 271–288.

Lafon, M.; Ramzi, E.; Rambour, C.; Audebert, N.; and Thome, N. 2024. Gallop: Learning global and local prompts for vision-language models. In *ECCV*, 264–282.

Li, J.; Li, D.; Savarese, S.; and Hoi, S. 2023a. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, 19730–19742.

Li, K.; Wang, Y.; Zhang, J.; et al. 2023b. Uniformer: Unifying convolution and self-attention for visual recognition. *TPAMI*, 45(10): 12581–12600.

Li, R.; Feng, Z.; Xu, T.; Li, L.; Wu, X.-J.; Awais, M.; Atito, S.; and Kittler, J. 2025. C2c: Component-to-composition learning for zero-shot compositional action recognition. In *ECCV*, 369–388.

Li, S.; Li, B.; Sun, B.; and Weng, Y. 2024. Towards Visual-Prompt Temporal Answer Grounding in Instructional Video. *TPAMI*, 46(12): 8836–8853.

Li, Y.; Wu, C.-Y.; Fan, H.; Mangalam, K.; Xiong, B.; Malik, J.; and Feichtenhofer, C. 2022. Mvitv2: Improved multiscale vision transformers for classification and detection. In *CVPR*, 4804–4814.

Lin, W.; Karlinsky, L.; Shvetsova, N.; Possegger, H.; Kozinski, M.; Panda, R.; Feris, R.; Kuehne, H.; and Bischof, H. 2023. Match, expand and improve: Unsupervised finetuning for zero-shot action recognition with language knowledge. In *ICCV*, 2851–2862.

Lin, Z.; Geng, S.; Zhang, R.; Gao, P.; De Melo, G.; Wang, X.; Dai, J.; Qiao, Y.; and Li, H. 2022. Frozen clip models are efficient video learners. In *ECCV*, 388–404.

Liu, A. H.; Jin, S.; Lai, C.; Rouditchenko, A.; Oliva, A.; and Glass, J. R. 2022. Cross-Modal Discrete Representation Learning. In *ACL*, 3013–3035.

Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2023a. Visual instruction tuning. In *NeurIPS*, volume 36, 34892–34916.

Liu, R.; Huang, J.; Li, G.; Feng, J.; Wu, X.; and Li, T. H. 2023b. Revisiting temporal modeling for clip-based image-to-video knowledge transferring. In *CVPR*, 6555–6564.

Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; and Guo, B. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 10012–10022.

Mentzer, F.; Minnen, D.; Agustsson, E.; and Tschannen, M. 2024. Finite scalar quantization: Vq-vae made simple. In *ICLR*, 1–12.

Nguyen, T.; Gadre, S. Y.; Ilharco, G.; Oh, S.; and Schmidt, L. 2023. Improving multimodal datasets with image captioning. In *NeurIPS*, volume 36, 22047–22069.

Ni, B.; Peng, H.; Chen, M.; Zhang, S.; Meng, G.; Fu, J.; Xiang, S.; and Ling, H. 2022. Expanding language-image pretrained models for general video recognition. In *ECCV*, 1–18.

Pan, J.; Lin, Z.; Zhu, X.; Shao, J.; and Li, H. 2022. St-adapter: Parameter-efficient image-to-video transfer learning. In *NeurIPS*, volume 35, 26462–26477.

Qing, Z.; Zhang, S.; Huang, Z.; Zhang, Y.; Gao, C.; Zhao, D.; and Sang, N. 2023. Disentangling spatial and temporal learning for efficient image-to-video transfer learning. In *ICCV*, 13934–13944.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *ICML*, 8748–8763.

Ramesh, A.; Pavlov, M.; Goh, G.; Gray, S.; Voss, C.; Radford, A.; Chen, M.; and Sutskever, I. 2021. Zero-shot text-to-image generation. In *ICML*, 8821–8831.

Rasheed, H.; Khattak, M. U.; Maaz, M.; Khan, S.; and Khan, F. S. 2023. Fine-tuned clip models are efficient video learners. In *CVPR*, 6545–6554.

Razavi, A.; Van den Oord, A.; and Vinyals, O. 2019. Generating diverse high-fidelity images with vq-vae-2. In *NeurIPS*, volume 32, 14837–14847.

Shi, Y.; Wu, X.; Lin, H.; and Luo, J. 2024. Commonsense Knowledge Prompting for Few-shot Action Recognition in Videos. *TMM*, 26: 8395–8405.

Tu, S.; Dai, Q.; Wu, Z.; Cheng, Z.-Q.; Hu, H.; and Jiang, Y.-G. 2023. Implicit temporal modeling with learnable alignment for video recognition. In *ICCV*, 19936–19947.

Van Den Oord, A.; Vinyals, O.; et al. 2017. Neural discrete representation learning. In *NeurIPS*, 6306–6315.

Wang, H.; Liu, F.; Jiao, L.; Wang, J.; Hao, Z.; Li, S.; Li, L.; Chen, P.; and Liu, X. 2024a. ViLT-CLIP: Video and Language Tuning CLIP with Multimodal Prompt Learning and Scenario-Guided Optimization. In *AAAI*, volume 38, 5390–5400.

Wang, M.; Huang, Z.; Kong, X.; et al. 2025. Action Detail Matters: Refining Video Recognition with Local Action Queries. In *CVPR*, 19132–19142.

Wang, M.; Xing, J.; Jiang, B.; Chen, J.; Mei, J.; Zuo, X.; Dai, G.; Wang, J.; and Liu, Y. 2024b. A Multimodal, Multi-Task Adapting Framework for Video Action Recognition. In *AAAI*, 5517–5525.

Wang, M.; Xing, J.; and Liu, Y. 2021. Actionclip: A new paradigm for video action recognition. *arXiv preprint arXiv:2109.08472*.

Wang, W.; Bao, H.; Dong, L.; Bjorck, J.; Peng, Z.; Liu, Q.; Aggarwal, K.; Mohammed, O. K.; Singhal, S.; Som, S.; et al. 2023. Image as a foreign language: Beit pretraining for vision and vision-language tasks. In *CVPR*, 19175–19186.

Wang, X.; Zhang, S.; Cen, J.; Gao, C.; Zhang, Y.; Zhao, D.; and Sang, N. 2024c. CLIP-guided prototype modulating for few-shot action recognition. *IJCV*, 132(6): 1899–1912.

Wasim, S. T.; Naseer, M.; Khan, S.; Khan, F. S.; and Shah, M. 2023. Vita-clip: Video and text adaptive clip via multimodal prompting. In *CVPR*, 23034–23044.

Wu, P.; Zhou, X.; Pang, G.; Zhou, L.; Yan, Q.; Wang, P.; and Zhang, Y. 2024a. Vadclip: Adapting vision-language models for weakly supervised video anomaly detection. In *AAAI*, volume 38, 6074–6082.

Wu, W.; Sun, Z.; and Ouyang, W. 2023. Revisiting classifier: Transferring vision-language models for video recognition. In *AAAI*, volume 37, 2847–2855.

Wu, W.; Wang, X.; Luo, H.; Wang, J.; Yang, Y.; and Ouyang, W. 2023. Bidirectional cross-modal knowledge exploration for video recognition with pre-trained vision-language models. In *CVPR*, 6620–6630.

Wu, Z.; Weng, Z.; Peng, W.; Yang, X.; Li, A.; Davis, L. S.; and Jiang, Y.-G. 2024b. Building an open-vocabulary video CLIP model with better architectures, optimization and data. *TPAMI*, 46(10): 4747–4762.

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

Yang, T.; Zhu, Y.; Xie, Y.; Zhang, A.; Chen, C.; and Li, M. 2023. Aim: Adapting image models for efficient video action recognition. In *ICLR*, 1–18.

Yu, C.; Liu, X.; Wang, Y.; Zhang, P.; and Lu, H. 2024. TF-CLIP: Learning text-free CLIP for video-based person re-identification. In *AAAI*, volume 38, 6764–6772.

Yue, X.; Ni, Y.; Zhang, K.; Zheng, T.; Liu, R.; Zhang, G.; Stevens, S.; Jiang, D.; Ren, W.; Sun, Y.; et al. 2024. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *CVPR*, 9556–9567.

Zhang, J.; Huang, J.; Jin, S.; and Lu, S. 2024a. Vision-language models for vision tasks: A survey. *TPAMI*, 46(8): 5625–5644.

Zhang, P.; Li, X.; Hu, X.; Yang, J.; Zhang, L.; Wang, L.; Choi, Y.; and Gao, J. 2021. Vinvl: Revisiting visual representations in vision-language models. In *CVPR*, 5579–5588.

Zhang, R.; Hu, X.; Li, B.; Huang, S.; Deng, H.; Qiao, Y.; Gao, P.; and Li, H. 2023. Prompt, generate, then cache: Cascade of foundation models makes strong few-shot learners. In *CVPR*, 15211–15222.

Zhang, W.; Wan, C.; Liu, T.; Tian, X.; Shen, X.; and Ye, J. 2024b. Enhanced Motion-Text Alignment for Image-to-Video Transfer Learning. In *CVPR*, 18504–18515.

Zhou, K.; Yang, J.; Loy, C. C.; and Liu, Z. 2022. Learning to prompt for vision-language models. *IJCV*, 130(9): 2337–2348.

Zhu, W.; Han, Y.; Lu, J.; and Zhou, J. 2022. Relational reasoning over spatial-temporal graphs for video summarization. *TIP*, 31: 3017–3031.