Perceptual Projection Pruning: Diversity-Aware Video Token Pruning for Multimodal Large Language Models

Zhuangqiu Huang[1,2], Minxin Lai[1], Shuo Liu[1], Yu Zhang[1✉], and Jiaqi Wang[2,3✉]

[1] University of Science and Technology of China, China
[2] Shanghai Innovation Institute, China
[3] JD.com, China
{huangzq,mxlai,ls_ustc}@mail.ustc.edu.cn, yuzhang@ustc.edu.cn, wjqdev@gmail.com

Abstract. Applying Multimodal Large Language Models (MLLMs) to long-form video is challenged by the substantial computational cost of processing long token sequences. Current training-free reduction methods present a difficult trade-off between efficient and content diversity. Query-focused approaches are efficient but sacrifice content diversity, which is critical for general-purpose understanding and open-ended dialogue. Conversely, methods that preserve diversity by operating on high-dimensional features often incur significant computational overhead.
We introduce Perceptual Projection Pruning (P^3), a training-free framework that tries to address this trade-off. Our approach is based on the hypothesis that a low-dimensional yet informative representation can serve as an efficient proxy for diversity-aware visual token pruning. We first project high-dimensional features into a 3D perceptual space by computing three complementary saliency axes: dynamic (motion), regional (contextual contrast), and focal (local complexity). This representation then serves as a foundation for a subsequent diversity-aware pruning strategy. Experiments across SOTA MLLMs show our method retains nearly 95% of the vanilla model's performance while pruning up to 75% of tokens. Our work demonstrates that a carefully designed low-dimensional representation is highly effective for guiding diversity-aware visual token pruning in MLLMs.

Keywords: Video token pruning · MLLM · Video understanding

1 Introduction

Multimodal Large Language Models (MLLMs) have shown significant progress in unified reasoning across text and images. However, applying these models

✉ Corresponding author. yuzhang@ustc.edu.cn, wjqdev@gmail.com

to video understanding presents a computational challenge. The self-attention mechanism scales quadratically with the input sequence length. Video inputs generate long sequences of visual tokens, leading to substantial computational costs and memory usage, particularly during the prefill stage. Consequently, developing effective visual token pruning strategies is necessary for the efficient deployment of MLLMs in video applications.

Training-free methods are often preferred as they accelerate pre-trained models without requiring fine-tuning. Current approaches typically follow one of two paradigms, each with specific limitations [2, 6–8, 13]. Query-guided pruning discards tokens based on their relevance to a text prompt [4, 7, 8]. While this reduces computation, it inherently limits the diversity of the retained visual content, potentially affecting performance on open-ended generation or multi-round dialogue tasks. Conversely, content-aware methods aim to preserve visual diversity by clustering or matching tokens based on their raw, high-dimensional features [2, 6, 13, 18]. However, operating on the full high-dimensional feature space (e.g., $D \geq 2048$) introduces its own computational overhead, which may diminish the efficiency gains for long video inputs.

In this work, we aim to reconcile the trade-off between preserving content diversity and reducing the computational overhead of pruning. We posit that for the specific task of token pruning, analyzing the full high-dimensional feature space is not strictly necessary. We hypothesize that a compact, low-dimensional perceptual representation can serve as an efficient and effective proxy for guiding the selection of important tokens.

To implement this, we introduce **Perceptual Projection Pruning ($\mathbf{P^3}$)**, a training-free framework structured around a "Project, Cluster, and Prune" pipeline. First, the method projects original high-dimensional features into a compact 3D perceptual space. This is achieved by computing three complementary saliency metrics from local spatio-temporal neighborhoods:

(1) **Dynamic Saliency**: Derived from the magnitude of local motion vectors, this factor isolates regions with dynamic activity.

(2) **Regional Saliency**: Calculated via center-surround feature contrast, this factor identifies coherent objects and areas that are salient relative to their broader context.

(3) **Focal Saliency**: Based on local feature reconstruction error, this factor pinpoints unpredictable, fine-grained details and complex patterns.

This low-dimensional representation substantially reduces the computational cost of subsequent analysis, keeping the pruning overhead to only a small fraction of the MLLM's prefill time even on long videos. Second, P^3 clusters tokens in a joint space combining these perceptual scores with spatio-temporal coordinates to identify coherent visual entities. Finally, P^3 prunes the sequence using a diversity-aware stratified sampling strategy within each cluster, ensuring that different aspects of visual information are proportionally retained.

Our contributions are summarized as follows:

(1) We propose Perceptual Projection Pruning (P^3), a training-free framework for efficient video token reduction in MLLMs.

(2) We demonstrate that a low-dimensional perceptual projection can serve as an efficient proxy to guide diversity-aware pruning, providing a computationally efficient alternative to high-dimensional feature matching.

(3) Experiments on SOTA MLLMs indicate that our method achieves a favorable trade-off between performance and efficiency, retaining approximately 95% of the baseline performance while pruning up to 75% of tokens.

2 Related Works

Our research intersects with two primary fields: efficiency in Multimodal Large Language Models (MLLMs), and training-free token reduction techniques. We also draw principles from the classic field of spatio-temporal redundancy in video.

2.1 Efficient Inference for Multimodal Large Language Models

The substantial computational requirements of MLLMs [1,3,5,9,16,27,28], driven by the self-attention mechanism's quadratic complexity, have motivated extensive research into efficiency optimizations. Broadly, these efforts include model compression techniques such as quantization and weight pruning [20, 24, 36], as well as the design of efficient architectures incorporating alternatives to full self-attention [26, 35]. Our work complements these approaches by focusing on token reduction, a plug-and-play strategy that decreases the input sequence length for any standard Transformer architecture without requiring modifications to its weights or extensive retraining.

2.2 Training-Free Token Reduction in Vision

Training-free methods are advantageous as they can be applied to off-the-shelf pretrained models. Current approaches can be categorized by the information source used to guide the reduction.

One line of work is **Query-Aware Pruning** [4, 7, 8, 13, 21, 31], which utilizes the text query to assess the importance of each visual token. Tokens with low relevance to the query are discarded. While this is effective for specific, query-focused tasks, its reliance on a given prompt makes the resulting token set less suitable for general-purpose video representation, which is often required for open-ended dialogue.

Another major paradigm is **Content-Aware Reduction**, which aims to preserve the intrinsic diversity of visual content by operating solely on features.This includes both pruning [2, 18] and merging methods [6, 13], which progressively crops or combines feature-similar tokens. A key challenge for these methods is the computational cost. To make informed decisions, they often rely on similarity comparisons or clustering within the high-dimensional feature space. The cost of these operations, particularly when performed globally across all tokens, can itself become a significant overhead, creating a trade-off between the quality of the reduction and its own efficiency. Our work restrict computations in the

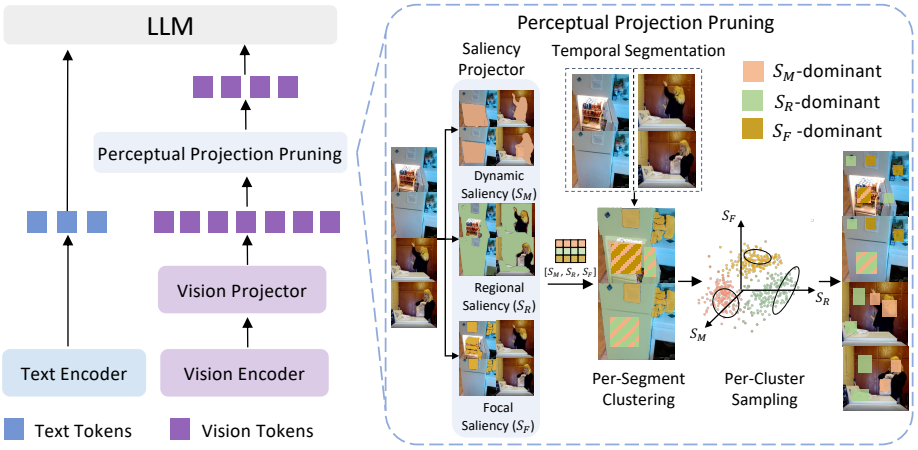

Fig. 1: Overview of the proposed framework. Visual tokens are first projected into a low-dimensional perceptual space comprising three saliency scores (see Section 3.1). The video is then divided into temporal segments, followed by clustering and stratified sampling to produce a compact yet diversity-preserving set of visual tokens for downstream MLLM inference.

original feature space to local neighborhoods and perform decisions in a compact perceptual space, reducing pruning overhead while preserving visual diversity.

2.3 Spatio-Temporal Redundancy in Video

Exploiting redundancy is a foundational concept in video processing. Classic video codecs [15] differentiate between full-content frames (I-frames) and predicted frames (P/B-frames) to remove high degrees of temporal redundancy between adjacent frames. Similarly, image compression standards address spatial redundancy within a single frame, such as large areas of uniform color. Our work explicitly target and remove temporal redundancy by calculating Dynamic Saliency and applying temporal coalescing. Meanwhile, by using Regional and Focal Saliency to identify salient entities, we implicitly determine the remaining large, non-salient areas constitute spatial redundancy and can be pruned.

3 Method

We propose a training-free framework for video token reduction in MLLMs. The method operates on the feature sequences extracted by the vision projector. As illustrated in Figure 1, the pipeline consists of three phases: (1) projecting high-dimensional tokens into a compact perceptual saliency space; (2) partitioning the video into temporal segments; and (3) performing diversity-aware pruning through clustering and stratified sampling.

3.1 Perceptual Saliency Projection

We posit that for the purpose of pruning, the visual importance of a token can be effectively represented by projecting its high-dimensional feature $\mathbf{v}_i \in \mathbb{R}^D$ into a compact 3D space comprising three orthogonal perceptual axes. Let $V = \{\mathbf{v}_i\}_{i=1}^N$ denote the input video tokens, where each token is associated with spatio-temporal coordinates (t_i, h_i, w_i).

Dynamic Saliency (S_M) This component quantifies motion magnitude. For a token $\mathbf{v}_i$ at time $t_i > 0$, we estimate a discrete motion vector $\mathbf{m}_i \in \mathbb{Z}^2$ by identifying the spatial offset of the most feature-similar token in the preceding frame $(t_i - 1)$. To improve robustness against local noise, we aggregate this information over a local spatio-temporal neighborhood $\mathcal{N}(i)$. The dynamic saliency score is defined as the average magnitude of these motion vectors:

$$S_M(i) = \frac{1}{|\mathcal{N}(i)|} \sum_{j \in \mathcal{N}(i)} \|\mathbf{m}_j\|_2. \tag{1}$$

For the first frame ($t_i = 0$), we define $\|\mathbf{m}_i\|_2 = 0$.

Regional Saliency (S_R) This component identifies coherent regions that are visually distinct from their surroundings. We define a local "center" neighborhood $\mathcal{N}_c(i)$ and a larger, disjoint "surround" neighborhood $\mathcal{N}_s(i)$. Let $\boldsymbol{\mu}_c(i)$ and $\boldsymbol{\mu}_s(i)$ be the mean feature vectors of these respective regions. The raw regional saliency is computed as the cosine distance between them:

$$S_{R,\text{raw}}(i) = 1 - \frac{\boldsymbol{\mu}_c(i) \cdot \boldsymbol{\mu}_s(i)}{\|\boldsymbol{\mu}_c(i)\|_2 \|\boldsymbol{\mu}_s(i)\|_2}. \tag{2}$$

To mitigate statistical instability at frame boundaries where the neighbor count $N_s = |\mathcal{N}_s(i)|$ is low, we apply a credibility weighting $\alpha(i) = \text{norm}(\log(1 + N_s))$, where $\text{norm}(\cdot)$ denotes min-max normalization across tokens in the video. The final score interpolates between the raw value and its global median $\tilde{S_R}$:

$$S_R(i) = \alpha(i) \cdot S_{R,\text{raw}}(i) + (1 - \alpha(i)) \cdot \tilde{S_R}. \tag{3}$$

Focal Saliency (S_F) This component captures fine-grained details by measuring local unpredictability (reconstruction error). We approximate a token $\mathbf{v}_i$ as a weighted sum of its spatial neighbors $j \in \mathcal{N}_{sp}(i)$, where weights are determined by feature similarity:

$$\hat{\mathbf{v}}_i = \sum_{j \in \mathcal{N}_{sp}(i)} \frac{\exp(\cos(\mathbf{v}_i, \mathbf{v}_j))}{\sum_k \exp(\cos(\mathbf{v}_i, \mathbf{v}_k))} \mathbf{v}_j. \tag{4}$$

The raw focal saliency is the cosine distance $1 - \cos(\mathbf{v}_i, \hat{\mathbf{v}}_i)$. Similarly, we apply credibility weighting based on the valid spatial neighbor count, using the same interpolation scheme as for $S_R(i)$ to obtain the final $S_F(i)$.

Justification for Perceptual Saliency The necessity of this design is based on its ability to capture complementary aspects of visual importance.

S_M serves as the unique channel for temporal information, which could indicate motion as well as the change of scenes. Within the static domain, S_R and S_F are designed to be orthogonal by measuring two different types of information. S_R evaluates **inter-region heterogeneity**, identifying importance based on the contrast between a region's average features and its surroundings (e.g., a smooth, solid-colored object). In contrast, S_F evaluates **intra-region heterogeneity**, identifying importance based on the complexity and unpredictability of a region's internal structure. A model relying on only one of these scores would fail in common scenarios; for instance, a model without S_R would be blind to simple salient objects, while one without S_F would miss information-rich textures.

3.2 Temporal Segmentation

Long videos contain visually heterogeneous moments, and directly clustering all visual tokens across the entire sequence may mix unrelated temporal contexts. We therefore divide the video into temporal segments and apply the subsequent pruning procedure within each segment. This design keeps the clustering local in time, preserves temporal coherence, and allows the token budget to be distributed across different video moments.

By default, we use a simple uniform temporal partitioning strategy. The partition interval is specified in the implementation details. We also study an adaptive partitioning variant based on dynamic saliency in the ablation study, with its full definition provided in the Supplementary Material Sec. S2.

3.3 Diversity-Aware Pruning Strategy

For each segment, we apply a pruning strategy designed to preserve the diversity of visual entities. This process is formalized as a two-stage pipeline: spatially-aware clustering followed by stratified sampling. The first frame (anchor frame) of each segment is processed separately from the remaining frames using the same clustering and sampling rule. This prevents tokens in the anchor frame from being suppressed by temporally redundant non-anchor tokens and helps preserve the boundary state of each segment. The pseudocode for the per-segment pruning process is provided in Algorithm S1 of the Supplementary Material.

Spatially-Aware Clustering. The goal of this stage is to partition the token set V_{seg} of a segment into K disjoint clusters $\{\mathcal{C}_1, \ldots, \mathcal{C}_K\}$ that represent coherent visual entities. We construct a 6D feature vector $\mathbf{f}_i$ for each token v_i: $\mathbf{f}_i = \text{Concat}([S_M(i), S_R(i), S_F(i)], [t_i, h_i, w_i])$. The inclusion of spatio-temporal coordinates acts as a prior, encouraging clusters to be contiguous. We employ K-Means to partition 6D feature into clusters.

Stratified Sampling. To prevent information collapse, where a saliency type (e.g., motion) dominates the selection within an entity, we perform stratified sampling. For each cluster $\mathcal{C}_k$, we first partition its tokens into three strata based on their dominant saliency:

$$\mathcal{S}_{k,T} = \{v_i \in \mathcal{C}_k \mid T = \underset{T' \in \{M,R,F\}}{\text{argmax}} \; S_{T'}(i)\}. \tag{5}$$

The total token budget for the cluster N_k is determined by the segment's global retention ratio r. We then allocate this budget to each stratum in proportion to its size $|\mathcal{S}_{k,\mathrm{T}}|$:

$$N_{k,T} = \left\lfloor N_k \times \frac{|\mathcal{S}_{k,T}|}{|\mathcal{C}_k|} \right\rfloor. \tag{6}$$

Finally, within each stratum $\mathcal{S}_{k,T}$, we select the top $N_{k,T}$ tokens with the highest corresponding saliency score S_T. The final set of kept tokens for the segment is the union of these selected subsets:

$$\mathcal{I}_{seg} = \bigcup_{k=1}^{K} \bigcup_{T \in \{M,R,F\}} \mathrm{TopK}(\mathcal{S}_{k,T}, N_{k,T}). \tag{7}$$

Global Temporal Coalescing. As a post-processing step, we remove static redundancies across the entire video. Let $\mathcal{I}_{global}$ be the union of all per-segment kept sets. We define the set of static tokens as $V_{static} = \{v_i \mid \|\mathbf{m}_i\|_2 = 0, t_i > 0\}$. The final pruned set is obtained by filtering out these static tokens: $\mathcal{I}_{final} = \mathcal{I}_{global} \setminus V_{static}$. This operation ensures that while the first appearance of a static object (at $t = 0$ or scene cuts) is preserved, subsequent identical tokens are efficiently removed.

4 Experiments

In this section, we evaluate the proposed Perceptual Projection Pruning (P^3) framework. Our evaluation is designed to demonstrate the framework's robustness across multiple pruning ratios, its scalability to 32B-parameter models, its generalization on diverse video understanding tasks, and its performance under extended context lengths (e.g., 1024 frames).

4.1 Experimental Setup

Models and Baselines. We implement P^3 on representative MLLMs: Qwen3-VL (8B and 32B) [27], InternVL3.5-8B [28], and LLaVA-OneVision-1.5 [3]. Baseline methods include query-agnostic approaches (DivPrune [2], PruneVid [13], PruMerge [25]), as well as query-aware methods (CDPruner [34], FastV [8]).

Datasets and Scope. The evaluation is conducted primarily on VLMEvalKit [10] for video understanding benchmarks (VideoMME [11], LongVideoBench [29] and MLVUc̃itezhou2024mlvu, MVBench [19]).We also conduct evaluation with lmms-eval [33] for general video tasks (EgoSchema [22], NExTQA [30], SEED-Bench [17]) at 25% retain ratio. **Note on Task Scope:** The P^3 framework is explicitly designed to reduce spatio-temporal redundancy in videos. Its dynamic saliency component (S_M) relies on temporal variations. Consequently, on static image benchmarks where temporal context is absent, S_M degenerates, leading to suboptimal performance (e.g., 91.5% relative performance on SEED-Bench Image). Based on this structural characteristic, our experiments and claims are

scoped to video understanding tasks. On video-oriented benchmarks, the default P^3 achieves 100.7% retained performance, while its lower performance on static images further clarifies the intended video-centric scope of our method.

Implementation Details. All experiments are conducted on 4× NVIDIA H100 80GB GPUs. Our method is implemented in PyTorch and Triton. All methods use same optimized settings (e.g., Triton/CUDA kernels) and implementations for fair comparison (setup costs amortized). For PruneVid, we report Partial PruneVid (vision-only, denoted as "PruneVid†") to prune at the vision-encoder output, avoiding LLM changes for fair representation comparison. For PruMerge, since Qwen3-VL has no [CLS] token, we use the mean attention map of the last vision layer as a surrogate. To demonstrate robustness, the following key hyperparameters are kept fixed across all models and datasets. Extended configuration details and hardware specifications are documented in the Supplementary Material Sec. S3.

4.2 Main Results: Performance Across Varying Retention Ratios

We first evaluate the effectiveness of P^3 under different token retention ratios. Table 1 reports results on four representative video understanding benchmarks: VideoMME, LongVideoBench, MLVU, and MVBench. We compare P^3 with three types of training-free token reduction methods: query-agnostic pruning methods (DivPrune, PruneVid†), query-aware pruning methods (CDPruner, FastV), and token merging methods (PruMerge).

General Performance and Comparison. On Qwen3-VL-8B, P^3 achieves strong performance across all retention ratios. At 50% retention, P^3 retains 99.2% of the vanilla model performance, closely matching the query-aware CDPruner and outperforming other query-agnostic baselines. At 25% retention, P^3 retains 97.3% performance, reducing the gap to CDPruner to 0.8 points while outperforming PruMerge, DivPrune, PruneVid, and FastV. Even under the aggressive 10% retention setting, P^3 retains 92.8% performance and remains competitive with the strongest query-aware baseline.

Across different models, P^3 shows consistent behavior. On InternVL3.5-8B at 25% retention, P^3 retains 94.3% performance, remaining close to CDPruner and PruMerge. On LLaVA-OneVision-1.5, multiple pruning methods achieve retained performance above 100%, suggesting that this model is sensitive to long visual contexts and can benefit from removing redundant tokens. We therefore treat the LLaVA-OneVision-1.5 results as complementary evidence rather than the primary basis for comparing pruning quality.

Analysis of Dataset-Specific Variance. We observe noticeable performance fluctuations across different datasets and models. For instance, while P^3 performs relatively well on MVBench with InternVL3.5-8B, it is outperformed by DivPrune on the same dataset when using LLaVA-OneVision-1.5.

These fluctuations highlight the inherent limitations of our methodology. Projecting 2048-dimensional features into a 3D perceptual space (S_M, S_R, S_F) acts as a lossy compression. While this projection efficiently captures coarse physical and temporal variations, it inevitably discards fine-grained semantic

Model	Method	Retain Ratio	VideoMME	LongVideoBench	MLVU	MVBench	Avg. Retained (%)
Qwen3-VL-8B	Vanilla	100%	67.8	62.5	68.6	69.3	100.0
	DivPrune	50%	66.5	61.9	67.2	65.8	97.5
	PruneVid†		64.4	61.9	66.1	65.9	96.4
	FastV		64.3	60.9	65.0	66.5	95.7
	CDPruner		67.0	**62.7**	**68.2**	68.5	**99.4**
	PruMerge		67.0	61.8	67.3	**69.0**	98.9
	P^3		**67.3**	62.5	67.7	68.5	99.2
	DivPrune	25%	64.8	60.4	64.8	64.8	95.1
	PruneVid†		61.7	59.4	63.7	63.3	92.5
	FastV		59.4	59.0	62.1	62.2	90.6
	CDPruner		66.0	**62.2**	**66.7**	**68.1**	**98.1**
	PruMerge		65.3	60.6	65.9	67.5	96.6
	P^3		**66.1**	61.3	66.2	67.2	97.3
	DivPrune	10%	61.7	56.5	62.3	62.2	90.5
	PruneVid†		58.5	**59.7**	61.5	61.0	89.9
	FastV		54.3	53.8	57.9	56.7	83.1
	CDPruner		**64.3**	58.1	**63.9**	64.6	**93.5**
	PruMerge		63.3	57.8	62.5	62.3	91.7
	P^3		63.5	58.6	61.8	**64.8**	92.8
InternVL3.5-8B	Vanilla	100%	66.1	61.6	70.4	71.9	100.0
	DivPrune	50%	65.3	**60.9**	**70.2**	67.4	97.8
	PruneVid†		61.6	60.8	68.7	63.6	94.5
	FastV		62.3	59.2	67.6	66.4	94.7
	CDPruner		**65.7**	60.1	69.5	71.1	**98.7**
	PruMerge		65.5	60.4	68.7	**71.5**	98.5
	P^3		64.0	59.8	69.4	65.4	95.8
	DivPrune	25%	63.1	59.3	67.1	65.1	94.4
	PruneVid†		60.6	**60.0**	**68.2**	62.5	93.2
	FastV		58.7	55.7	62.2	59.6	87.6
	CDPruner		**63.4**	58.3	66.7	69.0	95.3
	PruMerge		63.2	56.9	67.1	**70.6**	**95.4**
	P^3		63.2	58.7	67.5	65.1	94.3
	DivPrune	10%	59.6	57.1	64.5	62.9	90.5
	PruneVid†		58.6	**58.2**	**65.8**	60.1	90.0
	FastV		53.7	52.1	56.7	51.8	79.6
	CDPruner		**60.0**	54.7	64.0	63.3	89.6
	PruMerge		56.6	53.6	61.2	**63.8**	87.1
	P^3		59.9	56.9	65.5	62.4	**90.7**
LLaVA-OneVision-1.5	Vanilla	100%	56.7	52.6	51.0	46.9	100.0
	DivPrune	50%	56.9	53.3	56.0	**50.2**	104.7
	PruneVid†		54.0	51.8	54.0	49.0	101.0
	CDPruner		**57.6**	**53.9**	**57.8**	49.7	**105.8**
	PruMerge		56.6	51.7	54.1	48.6	102.0
	P^3		56.8	52.7	55.1	49.3	103.4
	DivPrune	25%	56.9	52.4	57.2	**50.4**	104.9
	PruneVid†		53.1	52.1	56.3	49.2	102.0
	CDPruner		**57.7**	**53.3**	**57.6**	49.6	**105.5**
	PruMerge		54.0	51.5	55.9	48.9	101.7
	P^3		55.7	52.7	56.1	49.4	103.5
	DivPrune	10%	56.5	52.3	56.3	**50.8**	104.5
	PruneVid†		52.9	51.5	56.4	48.4	101.3
	CDPruner		**56.8**	**53.0**	**57.0**	50.2	**104.9**
	PruMerge		53.4	50.9	56.5	49.5	101.9
	P^3		53.0	51.2	55.7	49.9	101.6

Table 1: Robustness across varying retention ratios (10%, 25% and 50%).

nuances and non-salient local attributes. Query-aware methods (which can cross-attend to text cues) or high-dimensional clustering methods (which compute similarities across the full feature space) may be better equipped to retrieve such dataset-dependent cues.

Overall, these results indicate that compact perceptual pruning provides a favorable accuracy–efficiency trade-off. Although query-aware methods can achieve slightly higher single-turn accuracy in some settings, P^3 is query-agnostic and produces a reusable token set. This property is important for long-video and multi-turn scenarios, which we analyze in later sections.

4.3 Scalability to Larger MLLMs

We further examine whether P^3 scales to larger MLLMs. Table 2 reports the results on Qwen3-VL-32B at 25% retention. P^3 retains 97.5% of the vanilla model performance and outperforms CDPruner under the same setting. This suggests that the proposed perceptual pruning strategy remains effective when applied to a larger MLLM without substantial performance degradation.

Method	VideoMME Long	VideoBench	MLVU	MVBench	Avg. Retained (%)
Vanilla (100%)	71.8	65.9	72.1	73.7	100.0
DivPrune	70.1	63.6	70.9	69.1	96.6
PruneVid†	67.8	62.8	68.8	65.1	93.4
FastV	66.2	60.4	67.7	69.6	93.0
CDPruner	70.0	62.9	69.7	70.0	96.2
PruMerge	69.8	**63.9**	70.0	70.6	96.8
P^3	**70.2**	63.8	**71.1**	**71.4**	**97.5**

Table 2: Scalability on Qwen3-VL-32B at 25% retention.

4.4 Computational Overhead and Long-Video Scalability

The practical utility of any training-free pruning algorithm is determined by its end-to-end acceleration, which depends on the cost of the pruner itself. To evaluate this, we benchmark the wall-clock overhead and pruned prefill time of each method on the Qwen3-VL-8B model under different retention ratios. In this evaluation, the video input resolution is set to $64 \times 1344 \times 768$, which generates a dense sequence of 32,256 visual tokens before pruning. Table 3 reports the net speed-up at 25% retention ratios.

Overhead of Iterative Methods. We observe that the end-to-end processing times for DivPrune (3.89s) and PruMerge (2.33s) exceed the time required by the Vanilla model (2.23s), resulting in net speed-ups of less than 1.00×. This phenomenon stems from the algorithmic complexity of these methods. Both DivPrune and PruMerge employ iterative sampling or merging strategies, which select/merge one token at a time. In our long-video setting, retaining 25% of a 32,256-token sequence requires more than 8,000 sequential iterative steps. While such iterative approaches are highly effective and efficient on shorter sequences (e.g., the ∼5k token sequences corresponding to 16-frame inputs with lower resolution often evaluated in their original studies), their overhead scales poorly for dense, long-form video inputs, causing severe bottlenecks.

Efficiency of Projection-based and Query-aware Methods. CDPruner achieves a low pruning overhead of 0.12s, yielding a 3.91× net speed-up. However, in real-world multi-turn interactive scenarios involving the same video, the text query changes across turns, requiring CDPruner to re-execute its similarity masking (incurring the 0.12s overhead) for every new question.

Our proposed method, P^3, operates in an entirely query-agnostic manner. By projecting the high-dimensional features into a compact 3D perceptual space and performing localized clustering, P^3 avoids global $N \times N$ matrix multiplications and sequential iteration. P^3 introduces only 0.10s of overhead, achieving the fastest end-to-end time (0.55s) and the highest net speed-up (4.05×). Moreover, the pruned visual tokens can be cached and reused across multiple rounds of dialogue with zero additional pruning overhead, solidifying its practicality for interactive long-video understanding.

Method	Pruner Overhead (s)	Pruned Prefill (s)	Total Time (s)	Net Speed-up
Vanilla	-	2.23	2.23	1.00×
PruneVid†	0.14	0.45	0.59	3.78×
DivPrune	3.43	0.46	3.89	0.57×
CDPruner	0.12	0.45	0.57	3.91×
PruMerge	1.88	0.45	2.33	0.96×
P^3	**0.10**	0.45	**0.55**	**4.05**×

Table 3: Analysis of computational overhead and net end-to-end time on Qwen3-VL-8B at 25% retention. Our method achieves a significant net speed-up, while the overhead of DivPrune makes it slower than the vanilla model.

Table 4 reports the pruning overhead at 50%, 25%, and 10% retention ratios. P^3 introduces consistently low overhead across retention ratios, with 0.12s, 0.10s, and 0.10s pruning time, respectively. Compared with iterative token merging or sampling methods, the proposed projection-based pipeline avoids expensive high-dimensional global matching and sequential token-level operations. CDPruner is also efficient in short-video settings, but it is query-aware and needs to recompute the pruning mask when the text query changes.

Performance and Latency under 1024-frame inputs. To further stress-test scalability, we evaluate the methods under 1024-frame video inputs on Qwen3-VL-8B at a 25% retention ratio.

As presented in Table 5, P^3 outperforms other baseline methods evaluated under the same conditions. We hypothesize that in ultra-long sequences, query-guided or [CLS]-token-guided methods may experience attention dilution, where the cross-attention mechanisms distribute weights across a massive number of noisy tokens, potentially degrading selection accuracy. In contrast, the query-agnostic, localized perceptual filtering in P^3 performs a deterministic background reduction before LLM processing. This indicates that P^3 is particularly well-suited for ultra-long video scenarios, providing a stable pre-filtering mechanism that mitigates the context-window challenges in current MLLMs.

Setting	PruMerge	CDPruner	PruneVid†	P^3
50% Prune	3.12s	0.19s	0.51s	0.12s
25% Prune	1.88s	0.12s	0.14s	0.10s
10% Prune	0.69s	0.09s	0.12s	0.10s

Table 4: Pruning overhead under different retention ratios.

1024-Frame Evaluation on Qwen3-VL-8B (25% Retention)					
Method	VideoMME Long	VideoBench	MLVU	MVBench	Avg. (%)
Vanilla (100%)	67.7	59.1	75.0	66.6	100.0
PruneVid†	67.1	65.2	77.0	66.0	102.8
CDPruner	69.9	63.1	**77.4**	66.3	103.2
PruMerge	70.0	64.6	**77.4**	66.0	103.8
P^3	**70.4**	**65.1**	**77.4**	**66.4**	**104.3**

Table 5: In 1024-frame contexts, P^3 effectively filters redundancy, outperforming other methods which may suffer from attention dilution. This evaluation is conducted on VLMEvalKit with vllm [14]. Results of DivPrune are missing due to OOM.

As shown in Table 6, the unpruned prefill time reaches 19.57s. After pruning to 25% tokens, the prefill time of all methods becomes similar because they retain the same number of visual tokens. The difference therefore mainly comes from the pruning overhead. In this long-video setting, P^3 requires only 0.53s of pruning time and achieves the lowest total latency of 2.22s, corresponding to an 8.8× speedup over vanilla inference. This shows that the compact perceptual projection is particularly effective when the input sequence becomes long and visual redundancy is severe. A breakdown of P^3 shows that the projection cost grows approximately linearly with the number of frames (0.046s → 0.450s), while the clustering cost remains small (0.035s → 0.036s) because clustering is performed locally in a compact low-dimensional space.

Setting	PruMerge	CDPruner	PruneVid†	P^3
1024f 100% Prefill	19.57s	19.57s	19.57s	19.57s
1024f 25% Prefill	1.70s	1.67s	1.69s	1.69s
1024f 25% Prune	12.94s	1.98s	0.98s	0.53s
1024f 25% Total	14.64s	3.65s	2.67s	2.22s

Table 6: End-to-end latency in the 1024-frame setting. The total time is the sum of pruning time and pruned prefill time.

We also note that P^3 is query-agnostic. Once the visual tokens of a video are pruned, the same retained token set can be cached and reused for different user queries. Therefore, in multi-turn dialogue over the same video, the pruning overhead of P^3 is paid only once, while query-aware pruning methods need to update their token selection for each new question. This amortized property makes P^3 suitable for interactive long-video understanding.

4.5 Multi-turn and Fine-grained Diagnostics

A key advantage of query-agnostic pruning is that the retained visual tokens can be computed once and reused across different user queries. We evaluate this setting on MT-Video-Bench [23], which focuses on multi-turn video understanding. As shown in Table 7, the default P^3 achieves 62.5, matching DivPrune and outperforming CDPruner. The adaptive variant further improves the score to 63.0, suggesting that content-driven partitioning can be beneficial for heterogeneous multi-turn reasoning.

Benchmark	Vanilla	PruneVid†	DivPrune	CDPruner	PruMerge	P^3	P^3_{ada}
MT-Video-Bench	61.9	61.5	62.5	62.0	62.1	62.5	**63.0**

Table 7: Multi-turn dialogue evaluation on MT-Video-Bench at 25% retention. P^3 denotes the default configuration, and P^3_{ada} denotes the adaptive partitioning variant.

We further report fine-grained diagnostics in Table 8. On Charades STA [12], P^3 trails CDPruner, PruneVid and PruMerge, indicating that query-dependent temporal localization remains challenging for query-agnostic pruning. In contrast, P^3 achieves 59.5 on VSI-Bench [32], outperforming the vanilla model and all compared pruning baselines. This suggests that the proposed perceptual pruning strategy can preserve spatially informative visual tokens and reduce redundant or distracting context for visual-spatial reasoning.

Benchmark	Vanilla	PruneVid†	DivPrune	CDPruner	PruMerge	P^3
Charades STA	**48.5**	46.0	44.5	47.9	45.7	44.8
VSI-Bench	55.4	51.8	52.0	52.8	53.0	**59.5**

Table 8: Fine-grained task diagnostics at 25% retention.

4.6 Ablation Studies

To validate the core design choices of our framework, we conduct ablation studies using the Qwen3-VL-8B model at a 25% retention ratio. The default configuration uses uniform temporal partitioning, as described in Sec. 3.2. Since the subsequent perceptual projection, clustering, and saliency-stratified sampling modules are shared by both uniform and adaptive partitioning, we additionally report diagnostic ablations under the adaptive partitioning variant. These diagnostic results are used to analyze the behavior of the pruning module itself, while the effect of the partitioning strategy is isolated separately in Table 9.

Temporal segmentation. Table 9 compares adaptive partitioning with uniform temporal partitioning under the same token budget. Uniform partitioning

Setting	VideoMME	LongVideoBench	MLVU	MVBench	Avg.
Adaptive	64.8	61.0	64.5	66.9	64.3
Uniform step=2	64.4	60.2	65.5	66.8	64.2
Uniform step=4	**66.1**	**61.3**	**66.2**	67.2	**65.2**
Uniform step=8	65.7	60.4	64.7	**67.6**	64.6

Table 9: Ablation of temporal partitioning on Qwen3-VL-8B at 25% retention. Uniform step=4 is used as the default configuration in the main experiments.

with a step size of 4 achieves the best average performance on the standard benchmark suite, and is therefore used as the default configuration in our main experiments. The adaptive variant remains competitive and is further useful for heterogeneous temporal reasoning, as shown in Sec. 4.5.

Effectiveness of Low-Dimensional Perceptual Projection. Table 10 analyzes the effect of the clustering features and the three perceptual saliency components. This diagnostic study is conducted under the adaptive partitioning variant, where the same clustering and saliency-stratified sampling modules are applied within each temporal segment. The comparison with high-dimensional raw features shows that the proposed 3D perceptual projection provides a competitive low-cost proxy for diversity-aware pruning. The component ablation further shows that removing any of S_M, S_R, or S_F degrades performance, indicating the three saliency cues capture complementary aspects of token importance.

Clustering Feature	**Saliency Scores** S_M	S_R	S_F	**VideoMME**	**LongVideoBench**	**MLVU**	**MVBench**	**Avg.** Δ
3D Saliency (Ours)	✓	✓	✓	**64.8**	**61.0**	64.5	**66.9**	Ref.
High-Dim Raw Feature	-	-	-	64.5	60.2	**66.0**	**66.9**	**+0.10**
Ablation (w/o S_F)	✓	✓		64.4	60.6	64.1	66.4	-0.43
Ablation (w/o S_R)	✓		✓	64.1	59.5	64.2	66.4	-0.75
Ablation (w/o S_M)		✓	✓	63.7	58.4	63.7	65.8	-1.40

Table 10: Ablation of clustering features on Qwen3-VL-8B at 25% retention.

Clustering algorithm. We compare K-Means with DPC-KNN to examine the effect of the clustering algorithm. As shown in Table 11, K-Means achieves better average performance than DPC-KNN under the same pruning setting.

Method	VideoMME	LongVideoBench	MLVU	MVBench	Avg.
DPC-KNN	63.7	58.9	63.6	62.4	62.1
K-Means	64.8	61.0	64.5	66.9	64.3

Table 11: Ablation on clustering algorithms on Qwen3-VL-8B at 25% retention.

4.7 Sensitivity Analysis

We further analyze the sensitivity of several hyperparameters. The number of clusters K controls the shared spatially-aware clustering stage and is therefore relevant to both uniform and adaptive partitioning. The adaptive threshold τ_{abs}, in contrast, only applies to the adaptive partitioning variant and is reported as a variant-specific diagnostic rather than a default hyperparameter.

Number of clusters. We evaluate the effect of varying number of K-Means clusters. The results show P^3 is relatively stable within a reasonable range of K, suggesting pruning behavior is not overly sensitive to the number of clusters.

Adaptive threshold. For completeness, we also report the sensitivity of the adaptive partitioning variant to the absolute pace threshold τ_{abs}. This threshold is not used by the default uniform-partition configuration. The analysis is included to characterize the adaptive variant and to show how content-driven partitioning behaves under different boundary detection strengths.

Parameter	Value	VideoMME Long	VideoBench	MLVU	MVBench
Cluster Count (K)	$K = 5$	64.1	59.2	63.6	66.1
	$K = 10$	64.4	59.8	63.4	66.0
	$K = 20$	**64.8**	**61.0**	64.5	**66.9**
	$K = 40$	64.2	60.4	64.4	**66.9**
	$K = 80$	64.7	59.9	**64.6**	66.6
Seg. Threshold (τ_{abs})	$\tau = 0.5$	65.0	60.3	64.3	66.3
	$\tau = 0.6$	64.8	**61.0**	64.5	**66.9**
	$\tau = 0.7$	64.5	60.1	64.8	66.7
	$\tau = 0.8$	**65.5**	60.7	65.2	**66.9**
	$\tau = 0.9$	64.6	59.8	**65.5**	**66.9**

Table 12: Sensitivity of cluster count (K) and the absolute segmentation threshold (τ_{abs}) on Qwen3-VL-8B at 25% retention.

5 Conclusion

In this work, we try to address the computational cost of applying MLLMs to long-form video. We introduced a training-free token pruning framework centered on the hypothesis that a compact, low-dimensional representation can effectively guide a diversity-aware pruning strategy, serving as an efficient proxy for the original high-dimensional features. Our approach first projects high-dimensional tokens into a 3D perceptual space. This low-dimensional representation then guides a multi-stage pipeline of temporal segmentation, spatially-aware clustering, and stratified sampling.

Our experiments demonstrated that this method retains a high degree of the original model's performance across multiple benchmarks and MLLM architectures while pruning up to 75% of tokens. This validates our hypothesis that an efficient, low-dimensional perceptual projection can serve as an effective foundation for diversity-aware video token pruning.

References

1. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023)
2. Alvar, S.R., Singh, G., Akbari, M., Zhang, Y.: Divprune: Diversity-based visual token pruning for large multimodal models. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 9392–9401 (2025)
3. An, X., Xie, Y., Yang, K., Zhang, W., Zhao, X., Cheng, Z., Wang, Y., Xu, S., Chen, C., Wu, C., Tan, H., Li, C., Yang, J., Yu, J., Wang, X., Qin, B., Wang, Y., Yan, Z., Feng, Z., Liu, Z., Li, B., Deng, J.: Llava-onevision-1.5: Fully open framework for democratized multimodal training (2025), https://arxiv.org/abs/2509.23661
4. Arif, K.H.I., Yoon, J., Nikolopoulos, D.S., Vandierendonck, H., John, D., Ji, B.: Hired: Attention-guided token dropping for efficient inference of high-resolution vision-language models (2024), https://arxiv.org/abs/2408.10945
5. Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., Zhong, H., Zhu, Y., Yang, M., Li, Z., Wan, J., Wang, P., Ding, W., Fu, Z., Xu, Y., Ye, J., Zhang, X., Xie, T., Cheng, Z., Zhang, H., Yang, Z., Xu, H., Lin, J.: Qwen2.5-vl technical report. arXiv preprint arXiv:2502.13923 (2025)
6. Bolya, D., Fu, C.Y., Dai, X., Zhang, P., Feichtenhofer, C., Hoffman, J.: Token merging: Your ViT but faster. In: International Conference on Learning Representations (2023)
7. Cao, Q., Paranjape, B., Hajishirzi, H.: Pumer: Pruning and merging tokens for efficient vision language models (2023), https://arxiv.org/abs/2305.17530
8. Chen, L., Zhao, H., Liu, T., Bai, S., Lin, J., Zhou, C., Chang, B.: An image is worth 1/2 tokens after layer 2: Plug-and-play inference acceleration for large vision-language models (2024)
9. Comanici, G., Bieber, E., Schaekermann, M., Pasupat, I., Sachdeva, N., Dhillon, I., Blistein, M., Ram, O., Zhang, D., Rosen, E., et al.: Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. arXiv preprint arXiv:2507.06261 (2025)
10. Duan, H., Yang, J., Qiao, Y., Fang, X., Chen, L., Liu, Y., Dong, X., Zang, Y., Zhang, P., Wang, J., et al.: Vlmevalkit: An open-source toolkit for evaluating large multi-modality models. In: Proceedings of the 32nd ACM International Conference on Multimedia. pp. 11198–11201 (2024)
11. Fu, C., Dai, Y., Luo, Y., Li, L., Ren, S., Zhang, R., Wang, Z., Zhou, C., Shen, Y., Zhang, M., et al.: Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. arXiv preprint arXiv:2405.21075 (2024)
12. Gao, J., Sun, C., Yang, Z., Nevatia, R.: Tall: Temporal activity localization via language query. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 5267–5275 (2017)
13. Huang, X., Zhou, H., Han, K.: Prunevid: Visual token pruning for efficient video large language models. In: arXiv (2024)
14. Kwon, W., Li, Z., Zhuang, S., Sheng, Y., Zheng, L., Yu, C.H., Gonzalez, J.E., Zhang, H., Stoica, I.: Efficient memory management for large language model serving with pagedattention. In: Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles (2023)
15. Le Gall, D.: Mpeg: a video compression standard for multimedia applications. Commun. ACM **34**(4), 46–58 (Apr 1991). https://doi.org/10.1145/103085.103090, https://doi.org/10.1145/103085.103090

16. Li, B., Zhang, Y., Guo, D., Zhang, R., Li, F., Zhang, H., Zhang, K., Li, Y., Liu, Z., Li, C.: Llava-onevision: Easy visual task transfer (2024), https://arxiv.org/abs/2408.03326
17. Li, B., Wang, R., Wang, G., Ge, Y., Ge, Y., Shan, Y.: Seed-bench: Benchmarking multimodal llms with generative comprehension. arXiv preprint arXiv:2307.16125 (2023)
18. Li, D., Yang, Z., Lu, S.: Todre: Visual token pruning via diversity and task awareness for efficient large vision-language models (2025), https://arxiv.org/abs/2505.18757
19. Li, K., Wang, Y., He, Y., Li, Y., Wang, Y., Liu, Y., Wang, Z., Xu, J., Chen, G., Luo, P., Wang, L., Qiao, Y.: Mvbench: A comprehensive multi-modal video understanding benchmark. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 22195–22206 (June 2024)
20. Liang, T., Glossner, J., Wang, L., Shi, S., Zhang, X.: Pruning and quantization for deep neural network acceleration: A survey (2021), https://arxiv.org/abs/2101.09671
21. Lin, Z., Lin, M., Lin, L., Ji, R.: Boosting multimodal large language models with visual tokens withdrawal for rapid inference. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 5334–5342 (2025)
22. Mangalam, K., Akshulakov, R., Malik, J.: Egoschema: A diagnostic benchmark for very long-form video language understanding (2023), https://arxiv.org/abs/2308.09126
23. Pan, Y., Wang, Z., Xie, Q., Wen, Y., Zhang, Y., Zhang, G., Hu, H., Pan, Z., Huang, Y., Gan, Z., Lin, Y., Ping, A., Peng, T., Liu, J.: Mt-video-bench: A holistic video understanding benchmark for evaluating multimodal llms in multi-turn dialogues (2025), https://arxiv.org/abs/2510.17722
24. Qu, X., Aponte, D., Banbury, C., Robinson, D.P., Ding, T., Koishida, K., Zharkov, I., Chen, T.: Automatic joint structured pruning and quantization for efficient neural network training and compression. arXiv preprint arXiv:2502.16638 (2025)
25. Shang, Y., Cai, M., Xu, B., Lee, Y.J., Yan, Y.: Llava-prumerge: Adaptive token reduction for efficient large multimodal models. In: ICCV (2025)
26. Sun, Y., Li, Z., Zhang, Y., Pan, T., Dong, B., Guo, Y., Wang, J.: Efficient attention mechanisms for large language models: A survey (2025), https://arxiv.org/abs/2507.19595
27. Team, Q.: Qwen3 technical report (2025), https://arxiv.org/abs/2505.09388
28. Wang, W., Gao, Z., Gu, L., Pu, H., Cui, L., Wei, X., Liu, Z., Jing, L., Ye, S., Shao, J., et al.: Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. arXiv preprint arXiv:2508.18265 (2025)
29. Wu, H., Li, D., Chen, B., Li, J.: Longvideobench: A benchmark for long-context interleaved video-language understanding (2024), https://arxiv.org/abs/2407.15754
30. Xiao, J., Shang, X., Yao, A., Chua, T.S.: Next-qa: Next phase of question-answering to explaining temporal actions. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 9777–9786 (June 2021)
31. Xing, L., Huang, Q., Dong, X., Lu, J., Zhang, P., Zang, Y., Cao, Y., He, C., Wang, J., Wu, F., Lin, D.: Pyramiddrop: Accelerating your large vision-language models via pyramid visual redundancy reduction (2025), https://arxiv.org/abs/2410.17247
32. Yang, J., Yang, S., Gupta, A., Han, R., Fei-Fei, L., Xie, S.: Thinking in Space: How Multimodal Large Language Models See, Remember and Recall Spaces. In: CVPR (2025)

33. Zhang, K., Li, B., Zhang, P., Pu, F., Cahyono, J.A., Hu, K., Liu, S., Zhang, Y., Yang, J., Li, C., Liu, Z.: Lmms-eval: Reality check on the evaluation of large multimodal models (2024), https://arxiv.org/abs/2407.12772
34. Zhang, Q., Liu, M., Li, L., Lu, M., Zhang, Y., Pan, J., She, Q., Zhang, S.: Beyond attention or similarity: Maximizing conditional diversity for token pruning in mllms. arXiv preprint arXiv:2506.10967 (2025)
35. Zhang, Y., Fan, C.K., Ma, J., Zheng, W., Huang, T., Cheng, K., Gudovskiy, D., Okuno, T., Nakata, Y., Keutzer, K., et al.: Sparsevlm: Visual token sparsification for efficient vision-language model inference. In: International Conference on Machine Learning (2025)
36. Zhou, Z., Ning, X., Hong, K., Fu, T., Xu, J., Li, S., Lou, Y., Wang, L., Yuan, Z., Li, X., Yan, S., Dai, G., Zhang, X.P., Dong, Y., Wang, Y.: A survey on efficient inference for large language models (2024), https://arxiv.org/abs/2404.14294