Improve Temporal Reasoning in Multimodal Large Language Models via Video Contrastive Decoding

Daiqing Qi[1] **Dongliang Guo**[1] **Hanzhang Yuan**[1] **Handong Zhao**[2]

Mengxuan Hu[1] **Lehan Yang**[1] **Sheng Li**[1]

[1]University of Virginia [2]Adobe Research

Abstract

A major distinction between video and image understanding is that the former requires reasoning over time. Existing Video Large Language Models (VLLMs) demonstrate promising performance in general video understanding, such as brief captioning or object recognition within individual frames. However, they often struggle with temporal reasoning such as understanding continuous actions or tracking object transformations over time—which typically demands the integration of multiple frames in a temporally coherent manner. We first explore and explain such failures in Video LLMs from the perspective of *language and "image" priors.* While existing research has attempted to enhance the temporal understanding of VLLMs through various training strategies, the demand for expensive computational resources and training data often presents significant barriers. To this end, we further propose a simple yet novel idea for improving temporal reasoning in videos at no additional training cost. Specifically, to better capture the temporal structure across multiple frames—the key to effective temporal reasoning—we distort the temporal consistency in key frames *during the decoding phase*. Such corruption induces time-insensitive wrong responses from the model, which are then contrastively avoided when generating the final correct output. In this way, the model is encouraged to perform more temporally coherent reasoning. Our method yields consistent improvements across both temporal-specific and general video understanding benchmarks, demonstrating its effectiveness and generalizability.

1 Introduction

Benefiting from the significant advancements in Large Language Models in recent years, Video LLMs [14, 34, 35, 16, 30] have also experienced rapid development, exhibiting strong capabilities in general video understanding. A key distinction between video understanding and image understanding is that the former requires models to comprehend not only individual input frames but also the temporal relationships among them. Consequently, temporal perception is crucial for Video LLMs. However, recent studies [11, 21] have shown that even for simple and straightforward temporal reasoning questions that humans can easily answer, Video LLMs, such as LLaVA-Video [35], Video-LLaVA [14] and VILA [16, 22], often make mistakes that are clearly inconsistent with the ground truth. Recent studies [28, 8, 29, 11] have investigated the limitations of temporal reasoning in Video LLMs from a model-centric perspective. Given that a typical Video-LLM is composed of a vision encoder and an LLM backbone, a common analytical approach is to disentangle and examine the contributions of each component to temporal reasoning. While some works [28, 8, 29] attribute the temporal reasoning deficiencies of Video LLMs to ineffective video embeddings and accordingly focus on improving temporal information aggregation, others [11] leave aside the vision modules and

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

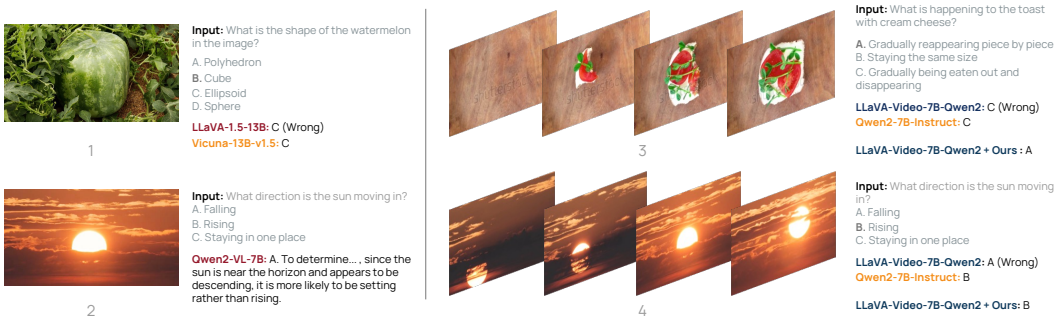

Figure 1: Example of how language and "image" priors affect Multimodal Large Language Models or Video Large Language Models in image or video understanding. (1), (3) show the negative effect of language priors and (2) reveals the how "image" priors from static visual content mislead the Video LLM's understanding of the video in (4).

embeddings, highlighting that the LLM itself exhibits weak sensitivity to the order of long textual sequences, as well as long video embeddings.

Existing works primarily focus on explaining the limitations of temporal perception from the perspective of the model itself. Consequently, the improvements derived from such analyses often require modifications to the model, such as refining video encoding [28, 8, 29] or improving the language counterpart [11]. However, these strategies typically involve modeling training (e.g., instruction tuning), which is computational- and data-intensive. Moreover, due to the increasing diversity of video LLM architectures nowadays, observations or findings from one certain architecture may not always generalize to others.

Towards this end, instead of digging into model architecture, we analyze model behavior through its responses to investigate under what circumstances video LLMs are more prone to making mistakes. We find that when video LLMs fail to understand temporal information in videos, they exhibit a behavior similar to what multimodal LLMs exhibit when hallucinate—namely, being influenced by language priors. More interestingly, due to the temporal dimension inherent in video inputs (compared to static image inputs), we observe that video LLMs also suffer from a negative influence of an *"image" prior*. For instance, the static visual features in individual frames—which do not carry temporal information, however, can mislead the LLM's perception of temporal dynamics. To avoid model-dependent modifications, we do not directly mitigating the negative effects at the model level. Instead, we take the opposite approach: we amplify these influencing factors to intentionally induce erroneous responses, and then use them as contrasting objectives in Contrastive Decoding [12]. This allows us to explicitly steer the decoding process away from such failure modes and toward generating the correct answers.

2 Uncovering Language and "Image" Prior in Temporal Reasoning

What Undermines Temporal Perception in Video LLMs? In contrast to existing works, we study the problem of limited temporal reasoning in Video-LLMs from a different perspective: instead of focusing on the model itself, we explore the conditions under which Video LLMs are more likely to struggle with temporal understanding. Although different models exhibit varying results on the same benchmark, it is observed that they tend to make mistakes more frequently on a specific subset of questions than on others. By analyzing the characteristics of these failure cases, key factors that undermine temporal reasoning can be uncovered.

Language prior typically refers to the prior knowledge in LLMs during (textual) pre-training or instruction-tuning, such as commonsense knowledge and reasoning in the context of multimodal understanding [17, 4]. However, it can sometimes hinder the model from grounding its answers in visual content, over-relying on the prior knowledge.

Fig. 1 (1) (top left) presents a typical example in image understanding with MLLMs. When asked about the shape of the watermelon in the image, LLaVA [19] chooses the incorrect option 'C', which aligns with the common sense that a watermelon is ellipsoidal. This strong language prior biases the

model toward selecting "C" instead of grounding to the visual input. When posing the same question without the image to its LLM backbone, Vicuna-13B-v1.5, the LLM responses with the same answer, revealing the influence of the LLM language prior on the MLLM's behavior. Similar observations are also mentioned in recent works [4] on MLLM hallucinations in image understanding. We are naturally curious whether the language prior has a similar influence on Video LLMs? If so, given the uniqueness of video inputs—such as *temporally ordered multi-frame sequences*—are there any distinct behaviors that Video LLMs may exhibit compared to MLLM with image inputs?

Language Prior in Video LLMs. We conducted an experiment to investigate whether language priors affect the temporal reasoning in Video LLMs. For this study, we selected two representative open-source Video LLMs—LLaVA-Video-7B-Qwen2 [35] and Video-LLaVA [13]—as test models, and used TempCompass [21] as the benchmark, which includes a variety of tasks closely related to temporal understanding, such as event ordering and attribute change. First, we performed standard inference with each Video LLM on TempCompass with both the video and textual input are provided. Then, we randomly sampled 200 questions where Video LLMs made mistakes, and conducted a blind evaluation on them, where the video input was removed and only the textual prompt was given to the LLM. The results from this blind evaluation can roughly reflect the influence of the language prior on inducing mistakes in Video LLMs.

Results show that, among the incorrect predictions made by LLaVA-Video-7B-Qwen2 and Video-LLaVA, **46.7%** and **38.9%** respectively matched the answers produced by blind LLM, significantly higher than random chance. In these cases, the Video LLMs failed to ground their responses in videos and instead relied on language priors, leading to incorrect answers.

Fig. 1 (3) provides an illustrative example. When the video is not provided, the blind LLM (Qwen2-7B-Instruct) selects option C, which corresponds to the answer with a higher prior probability that aligns with commonsense expectations. When both the video and the question are provided to the Video LLM, the model still chooses option C. This indicates that, despite access to visual input, the model remains influenced by the language prior and overlooks the temporal information present in the video. However, we observed in the experiments that in a number cases, even when the blind LLM predicted the correct answer, **the Video LLM surprisingly made an entirely different and incorrect choice** contradicting both the language prior and the information presented in the video.

Image Prior in Video LLMs. Fig. 1 (4) illustrates an interesting case. When the image is not provided, the blind LLM makes the right choice, which can be viewed as a positive influence of the language prior. However, the Video LLM still chooses the incorrect answer, "falling", a bias in this case introduced by the visual content of the video itself. As shown in Fig. 1 (2), when we extract a single frame from the middle of the video and input it into an MLLM (Qwen2-VL-7B) along with the question, the model perceives the scene as sunset and chooses "falling" accordingly. This visual bias ultimately leads the Video LLM to overlook the temporal progression of the sun rising and to make the wrong prediction.

Briefly, for image understanding, the input consists of both textual and visual information, and language priors can influence how MLLMs interpret visual content. In video understanding, the model must understand textual, visual, and temporal information simultaneously. In this setting, beyond language priors, visual bias—specifically, biases introduced by static frames, which we term as *"image" priors*, can also negatively impact temporal perception. As demonstrated in Fig. 1 (2), certain frames (e.g., one that resembles a sunset) may dilute the model's ability to accurately perceive temporal dynamics.

3 Video Temporal Distortion

Our analysis demonstrates that language and "image" priors can impair the temporal perception of Video LLMs. However, the underlying mechanisms can be more complex than they appear, potentially influenced by factors from pre-training, instruction tuning strategies, to model architecture. Instead of intervening in model design, such as the architecture or training strategies, which is often compute- and data-intensive, we seek a post-hoc correction approach. Given that priors can impair the temporal reasoning of Video LLMs, is it possible to develop a model-agnostic, plug-in method that uses the model's own temporally insensitive errors as contrastive signals, explicitly guiding the model away from such biases and increasing the probability of generating correct predictions?

Contrastive Decoding (CD) [12] offers a promising solution. CD is a decoding approach that aims to find text which maximizes the gap between the log-probabilities of a good and a bad LLM response. It helps us better avoid selecting tokens with high probability in the bad response, since a higher probability in the bad response means a smaller gap with its corresponding probability in the good response. Intuitively, if we can induce temporally insensitive responses from Video LLMs, they can be explicitly avoided by contrastive decoding. Therefore, the key challenge that follows is to guide Video LLMs to consistently generate such bad responses, so that they can be used as contrasting objectives during the final decoding phase.

3.1 Contrastive Decoding in Video Large Language Models

Decoding in LLMs. Considering a Video LLM parametrized by θ. It takes as input a text query $\mathbf{x}$ and a video context $\mathbf{V}$, and generate a relevant response $\mathbf{y}$ to the text query. The response $\mathbf{y}$ is sampled auto-regressively from the probability distribution conditioned on the query $\mathbf{x}$ and the video context $\mathbf{V}$, formulated as:

$$\mathbf{y}_t \sim p_\theta(\mathbf{y}_t \mid \mathbf{V}, \mathbf{x}, \mathbf{y}_{<t}) \propto \exp \text{logit}_\theta(\mathbf{y}_t \mid \mathbf{V}, \mathbf{x}, \mathbf{y}_{<t}), \tag{1}$$

where $\mathbf{y}_t$ denotes the token at time step t, and $\mathbf{y}_{<t}$ represents generated tokens up to $(t-1)$.

Contrastive Decoding in Video LLMs. Specifically, given a text query $\mathbf{x}$ and a video input $\mathbf{V}$, the model generates two output distributions: one conditioned on the original $\mathbf{V}$ and the other on the distorted video input $\mathbf{V}'$, which is derived by applying pre-defined distortion (e.g., adding noise to visual features as the simplest case) to $\mathbf{V}$. Then, a new contrastive probability distribution is computed by leveraging the differences between two original distributions. The new contrastive distribution p_{vtd} is formulated as:

$$p_{\text{vtd}}(\mathbf{y} \mid \mathbf{V}, \mathbf{V}', \mathbf{x}) = \text{softmax}\left[(1+\alpha)\text{logit}_\theta(\mathbf{y} \mid \mathbf{V}, \mathbf{x}) - \alpha \text{logit}_\theta(\mathbf{y} \mid \mathbf{V}', \mathbf{x})\right], \tag{2}$$

where larger α indicate a stronger amplification of the differences ($\alpha = 0$ reduces to regular decoding). The process is shown in Fig. 3. Wrong option "B" is assigned possibilities in the original normal distribution $\text{logit}_\theta(\mathbf{y} \mid \mathbf{V}, \mathbf{x})$ due to language or image priors. However, as temporal information is removed in the distorted video input, which further amplifies the bias, option "B" receives significantly higher scores in $\text{logit}_\theta(\mathbf{y} \mid \mathbf{V}', \mathbf{x})$. Finally, the score of "B" is notably reduced in p_{vtd} after Eq. 3 is applied, leading to correct answer "A". Following Li et al. [12], we also apply an adaptive plausibility constraint on p_{vtd}, where CD is applied only to high-probability tokens whose probabilities exceed a fraction $\beta \in [0, 1]$ of the maximum token probability. Details are discussed in Appendix A.

3.2 Video Temporal Distortion

What Makes an Effective Temporal Distortion. Distortion strategies play a key role in elevating the probabilities of bad responses in $\text{logit}_\theta(\mathbf{y} \mid \mathbf{V}', \mathbf{x})$ while diminishing those of good responses. To consistently generate bad responses as contrasting objectives, we need *remove temporal clues in the video input while maintaining confounding priors* that induces incorrect answers neglecting temporal information. Such balance is critical. For example, in the case shown in Fig 1(4), masking the entire video unexpectedly leads to the correct answer. This is undesirable, because if the correct answer is used as a contrasting objective, its possibility is reduced with Eq. 3. What we need is an carefully distorted input, like in Fig 1(2), which lacks temporal clues and misleads the model.

Intuitive solutions include: ① adding noise to visual features ② randomly shuffling frame sequences ③ randomly dropping frames. Fig. 2 shows results of LLaVA-Video-7B [35] on EventHallusion [32], where the videos depict continuous actions that require strong temporal perception. We adopt three different strategies when applying the distortions for contrastive decoding: randomly apply distortion on 1. all frames, 2. first half frames and 3. last half frames. While the overall results improve, they are notably sensitive to different sampling preferences toward the beginning or the end of the video. It implies that designing adaptive distortion strategies with greater stability and adaptability is crucial. Towards this end, we propose a Video Temporal Distortion strategy that adaptively distorts frames.

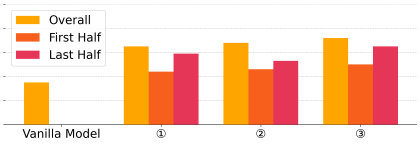

Figure 2: Distortion results.

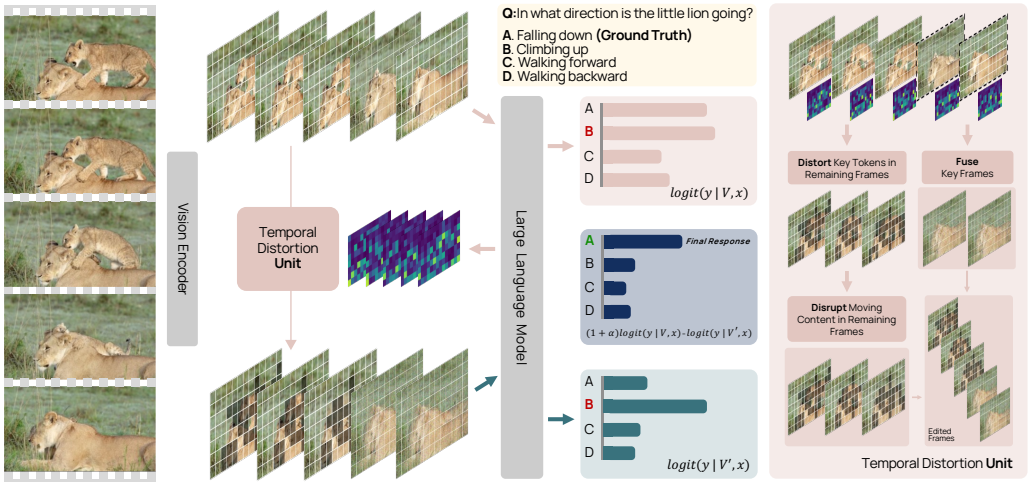

Figure 3: Overview. The original video embeddings $\mathbf{V}$ are first forwarded to the LLM with text query $\mathbf{x}$ to obtain $\text{logit}_\theta(\mathbf{y} \mid \mathbf{V}, \mathbf{x})$, where intermediate attention maps in LLM layers are retrieved and fed to Temporal Distortion Unit to guide the distortion of original video embeddings. Then distorted video $\mathbf{V}'$ is input to LLM to obtain $\text{logit}_\theta(\mathbf{y} \mid \mathbf{V}', \mathbf{x})$. Final token is generated from p_{vtd} in Eq. 3.

Temporal Distortion Criteria. We argue that, an ideal distortion should satisfy the following criteria: **(1)** it removes temporal information. **(2)** it includes priors that are likely to mislead the model, such as salient static visual features. **(3)** The distortion applied to the video should be adaptive towards different video inputs than random to ensure stability. Our solution is illustrated in Fig. 3. Given original video embeddings $\mathbf{V}$, and text query $\mathbf{x}$, we obtain distorted video embeddings $\mathbf{V}'$ with our Temporal Distortion Unit guided by attention maps from intermediate LLM layers to adaptively distort important frames with rich temporal information, while keeping less relevant frames that can potentially provide misleading context information or priors.

Framework. Fig. 3 (Right) shows the distortion pipeline. Given a set of K original video frame embeddings $\{\mathbf{v}_i\}_{i=1}^{K}$, and corresponding attention maps $\{\mathbf{A}_i\}_{i=1}^{L}$ from L intermediate LLM layers, We first compute the importance of each image token. To mitigate the bias of relying solely on the attention maps from the final layer, we compute *token importance* at each layer and aggregate it with momentum. Specifically, given attention map $\mathbf{A}_l$ at layer l, token importance matrix is computed as:

$$\mathbf{S}_l = \frac{1}{h}\sum_{i=1}^{h} \mathbf{A}_l^{(i,:,:)}[-1], \mathbf{A} \in \mathbb{R}^{(h,n,n)}$$

where h, n denote the value of attention heads and unmasked input tokens so far, respectively. Consequently, the importance of an image token $\mathbf{v}_j$ indexed at j is $\mathbf{S}_l[j]$ from layer l. To obtain the final token importance score, we apply momentum-based accumulation over all layers. Specifically, we iteratively update the aggregated importance map $\tilde{S}_l$ as follows:

$$\tilde{\mathbf{S}}_l = w_m \cdot \tilde{\mathbf{S}}_{l-1} + (1 - w_m) \cdot \mathbf{S}_l,$$

where $w_m \in [0, 1)$ is the momentum coefficient that controls the contribution of previous layers. This update emphasizes recent layer information while retaining long-range contributions from earlier layers. Then, the *frame importance* is computed as the sum of the importance scores of all image tokens within the frame.

Key Frame Fusion. First, we select the top-w_{fdr} (Frame Distortion Ratio) most important frames and remove the their temporal information by substituting each of them with the results of mean pooling over the selected set. In this way, temporal clues are removed while coarse-grained image context is retained, which can induce biased temporal insensitive inaccurate response. We further add a small amount of Gaussian noise with weight w_{fpr} (Frame Perturbation Ratio) to the pooled embeddings. Compared to directly dropping selected frames, mean pooling retains more confounding visual context, and is more robust than shuffling, yielding better results in practice consistently.

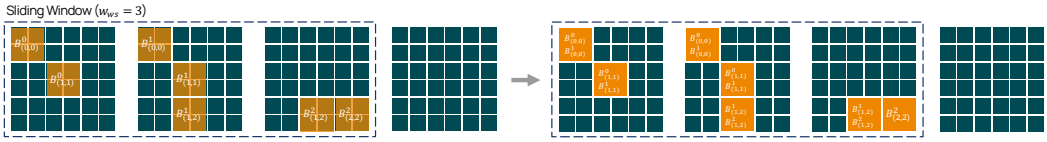

Figure 4: Disrupt moving content in remaining frames within a sliding window. **Left:** Marked dynamic blocks. **Right:** Fusion results of marked dynamics.

Distorting Key Tokens in Remaining Frames. Although the remaining frames are less important compared to the top-w_{fdr} most important ones, their large quantity still preserves a certain degree of temporal information. If frame-level fusion is applied to these frames in the same manner as it is to key frames, it can lead to excessive information loss across the video and damage the misleading image priors, making the model unable to consistently produce the most probable mistakes. As a result, the contrasting objective becomes overly random and loses its guiding effect in contrastive decoding. Therefore, we design a more fine-grained distortion strategy specifically for this subset of frames. First, for each remaining frame, we mask the top-w_{tdr} (Token Distortion Ratio) most important image tokens, thereby masking regions that potentially contain temporal cues.

Disrupting Moving Content in Remaining Frames. In the previous step, we performed fine-grained token-level masking on each of the remaining frames based on token importance. However, since attention maps sometimes do not accurately reflect the actual importance of each token, we instead leverage visual similarity between frames in this step to "blur" the moving content within the frames (if any), thereby removing potential temporal changes while preserving the overall image context.

Fig 4 visualizes this step. A sliding window (non-overlapping) of size w_{ws} is adopted to process the entire video sequence, where we independently process the frames within a window each time. A token is considered *dynamic* if its corresponding visual content changes significantly across the window. However, due to the high spatial resolution and token granularity of popular vision encoders like CLIP [27], direct per-token comparisons are neither robust nor efficient. To address it, we first downsample each frame from size (H, W) to $(\frac{H}{w_{\text{bs}}}, \frac{W}{w_{\text{bs}}})$ by applying mean pooling over non-overlapping patches, as illustrated in Fig. 4. Each resulting region is termed as *Block*.

Downsampling. Note that in the downsampling stage, i.e., when we downsample each frame from size (H, W) to $(\frac{H}{w_{\text{bs}}}, \frac{W}{w_{\text{bs}}})$, *we do not really reduce token numbers*. As illustrated in Fig. 4 (left), we actually replace all image tokens within one "downsampled" region with the mean of the tokens included. For example, the value of each of the four token within $\mathbf{B}^0_{(0,0)}$ is the mean of the four tokens.

For a given frame t and block at position (i, j), we denote it as $\mathbf{B}^t_{(i,j)}$. We define its *similarity score* as the average cosine similarity with all blocks at the same position in the other frames within the window:

$$\text{sim}(\mathbf{B}^t_{(i,j)}) = \frac{1}{w_{\text{ws}}-1} \sum_{\substack{t'=1 \\ t' \neq t}}^{w_{\text{ws}}} \cos\left(\mathbf{B}^t_{(i,j)}, \mathbf{B}^{t'}_{(i,j)}\right).$$

Blocks with lower similarity scores are considered more dynamic, as they indicate greater temporal change across frames. With similarity scores for all blocks, we select the top-w_{cfr} (Content Fusion Ratio) blocks with the lowest similarity scores across the entire window and label them as *Dynamic Blocks*. For each block $\mathbf{B}^t_{(i,j)}$, if it is not dynamic, we retain its original (pre-downsampled) token values. If it is dynamic and the same position across other frames in the window also contains dynamic blocks, we apply mean pooling across those positions to overwrite $B^t_{(i,j)}$ as follows.

$$\hat{\mathbf{B}}^t_{(i,j)} = \frac{1}{|\mathcal{D}_{(i,j)}|} \sum_{t' \in \mathcal{D}_{(i,j)}} \mathbf{B}^{t'}_{(i,j)},$$

where $\mathcal{D}_{(i,j)}$ is the set of frames within the window where block (i, j) is marked as dynamic. Otherwise it remains unchanged. This process disrupts temporal cues by smoothing moving content while preserving static visual context details. Finally, the concatenation of the distorted key frames and the remaining frames constitutes the temporally distorted video representation $\mathbf{V}'$, which participates in contrastive decoding as defined in Eq. 3. Technical details are available in Appendix B.

Metrics	VILA	PLLaVA	Video-ChatGPT	Video-Chat2	LLaMA-VID	ShareGPT4-Video	LLaVA-Video	+VCD	+SID	+TCD	+Ours
Match Rate	100.0	100.0	100.0	83.13	100.0	89.4	100.0	100.0	100.0	100.0	**100.0**
Entire	53.5	45.6	14.9	16.7	30.7	11.4	52.1	53.4	53.3	53.6	**55.3**
Interleave	62.2	68.9	79.8	58.6	98.9	93.7	60.3	65.7	65.4	66.8	**75.6**
Misleading	83.3	81.4	21.6	22.6	43.1	6.8	82.5	82.8	82.9	83.3	**84.2**
Overall	65.0	65.5	47.2	37.9	66.0	49.1	63.5	66.5	66.4	67.2	**72.1**

Table 1: EventHallusion evaluation results. Our method notably outperforms baselines on all tasks. Best results are shown in **bold**.

4 Experiments

Benchmarks. We first evaluate models on temporally-oriented benchmarks, including TempCompass [21] and EventHallusion [32]. The former targets temporal reasoning, while the latter assesses understanding of continuous actions. We then perform a comprehensive evaluation on general video understanding benchmarks, including VideoMME [5] and MLVU [36].

Models. We choose representative and widely-used video LLMs, Video-LLaVA [15] and LLaVA-Video-7B-Qwen2 [35], as backbone. Based on them, we apply our temporal distortion mechanism with contrastive decoding. In our experiments, we compare our method with popular Video LLMs as well as alternative contrastive decoding strategies, including VCD [9], SID [7] and TCD [32].

Configuration. By default, we adopt 8 frm for Video-LLaVA [15] and 32 frm for LLaVA-Video-7B-Qwen2 [35]. We run all inferences on NVIDIA A6000 GPUs and A100 GPUs. **Detailed experiment configuration and hyperparameter settings are available in Appendix C.**

4.1 Results

Through the experiments, we fist aim to explore the following questions: (1) Can our video temporal distortion effectively induce poor responses, thereby serving as a contrasting signal to improve temporal reasoning via contrastive decoding? (2) If so, can enhanced temporal perception also benefit general video understanding tasks? (3) Does strengthening temporal perception introduce a potential conflict with spatial understanding capabilities?

Temporal Reasoning. Tab. 2 shows results on TempCompass [21]. Across all four tasks and five temporally-oriented categories, our method consistently improves model performance when applied to Video-LLaVA and LLaVA-Video-7B-Qwen2. Compared to existing contrastive decoding methods such as VCD, our approach also demonstrate clear advantages. Similarly, on Tab. 1, our method yields notably better results on all tasks over baselines.

Holistic Video Understanding. Our method shows clear advantages on temporally-oriented benchmarks, which primarily feature videos depicting short-term, continuous changes of an event or object. In contrast, comprehensive video understanding benchmarks involve much longer videos and broader reasoning tasks across multiple aspects of video content. We evaluate LLaVA-Video-7B-Qwen2, the more powerful Video LMM, on Video-MME [5] and MLVU [36] for holistic video understanding. Results are shown on Tab. 3 and Tab. 4. Our method also demonstrates clear advantages over baselines, especially on temporally-oriented tasks, such as Temporal Perception and Reasoning in Video-MME [5] and Action Count (AC) and Action Order (AO) in MLVU [36].

4.2 Analysis

Ablations. We also explore the contribution of each component in our method, as well as how hyperparameters affect model performance. Tab. 5 demonstrates the effectiveness of each module in Temporal Distortion Unit, and the importance of sampling key frames with attention guidance. In Fig. 5, we show how each hyperparameter in Contrastive Decoding (CD) and Temporal Distortion Unit (TDU) influences model performance on TempCompass with LLaVA-Video-7B-Qwen2 [35].

From Fig. 5, we observe that a moderate level of distortion is crucial for effective contrastive decoding. In the ablation study of parameters most closely related to the degree of distortion—such as Frame Distortion Ratio, Token Distortion Ratio, Content Fusion Ratio, and again Frame Distortion

	Model	Intern-VL2 [2]	LLaVA [10] OneVision	Long VA [33]	VILA [16]	VID-LLaVA	+VCD	+SID	+TCD	+Ours	LLaVA-Video	+VCD	+SID	+TCD	+Ours
Multi-Choice QA	Action	93.8	96.5	92.3	92.9	76.0	76.9	77.0	77.2	**78.4** +2.4	95.6	96.4	96.4	96.3	**96.7** +1.1
	Direction	43.9	40.6	36.7	33.7	35.2	35.8	35.8	35.9	**36.7** +1.5	40.3	41.3	41.6	41.6	**43.6** +3.3
	Speed	51.1	45.4	43.2	44.1	35.6	37.0	37.1	37.4	**38.6** +3.0	50.5	49.2	49.6	49.9	**51.7** +1.2
	Event Order	67.2	69.5	54.3	50.0	37.7	39.0	39.1	39.4	**40.4** +2.7	71.2	71.8	71.2	69.9	**72.5** +1.3
	Attr. Change	59.9	56.9	52.4	60.0	40.9	42.0	42.2	42.1	**43.8** +2.9	71.5	72.5	72.7	72.8	**75.3** +3.8
	Average	65.5	64.8	56.1	56.4	45.5	46.5	46.6	46.8	**48.1** +2.6	65.8	66.1	66.2	66.1	**68.0** +2.2
Yes/No QA	Action	84.8	86.0	86.2	84.8	74.3	75.2	75.4	75.4	**76.1** +1.8	87.7	88.4	88.7	88.9	**90.4** +2.7
	Direction	53.2	55.3	50.4	52.2	51.8	52.5	52.4	52.7	**53.8** +2.0	54.3	54.9	54.6	55.1	**56.7** +2.4
	Speed	61.3	57.4	53.1	54.3	50.2	51.2	51.2	51.4	**53.3** +3.1	58.1	58.9	59.8	59.4	**62.1** +4.0
	Event Order	70.7	76.2	61.8	61.2	49.2	49.2	49.3	49.2	**49.7** +0.5	67.2	66.4	66.5	66.5	**67.4** +0.2
	Attr. Change	63.0	59.1	54.5	61.3	51.1	52.0	51.9	52.1	**53.2** +2.1	63.6	63.7	63.8	63.7	**65.8** +2.2
	Average	68.2	69.7	62.1	63.6	56.3	57.1	57.1	57.2	**58.3** +2.0	66.8	67.4	67.5	67.4	**69.4** +2.6
Matching	Action	96.6	96.0	94.6	95.0	87.9	88.0	88.0	88.2	**89.1** +1.2	96.0	96.2	96.4	96.0	**96.6** +0.6
	Direction	59.9	56.9	54.4	58.7	53.8	53.8	53.9	54.1	**55.2** +1.4	59.3	59.6	59.7	59.9	**60.9** +1.6
	Speed	67.0	61.9	53.3	60.5	58.4	58.6	58.9	58.8	**59.9** +1.5	61.6	61.8	61.9	61.9	**62.3** +0.7
	Event Order	84.0	81.3	64.3	66.0	59.0	61.3	61.9	62.2	**65.0** +6.0	71.7	75.4	75.8	76.0	**80.7** +9.0
	Attr. Change	77.1	73.8	62.5	65.3	58.3	59.8	59.7	59.4	**63.5** +5.2	71.5	73.9	74.0	74.2	**77.1** +5.6
	Average	77.1	73.8	65.7	68.9	63.3	64.1	64.3	64.3	**66.4** +3.1	71.8	73.1	73.2	73.2	**75.2** +3.4
Generation	Action	84.6	79.3	75.8	74.7	50.8	51.4	51.5	51.7	**53.3** +2.5	85.5	86.1	85.9	85.7	**87.9** +2.4
	Direction	38.8	30.7	35.3	36.2	28.7	29.2	29.0	29.3	**30.3** +1.6	39.8	40.4	40.6	40.7	**41.6** +1.8
	Speed	31.2	25.3	32.2	31.7	23.2	24.0	24.3	24.1	**25.5** +2.3	32.0	33.1	32.7	33.0	**34.1** +2.1
	Event Order	60.8	56.8	35.3	46.7	38.2	38.3	38.5	38.6	**39.9** +1.7	61.7	62.1	61.9	62.2	**63.1** +1.4
	Attr. Change	60.2	57.4	45.8	47.1	33.6	33.9	34.1	33.9	**35.6** +2.0	61.3	62.2	62.3	61.9	**63.5** +2.2
	Average	52.1	47.6	44.7	47.1	34.8	35.3	35.4	35.4	**36.9** +2.1	55.8	56.5	56.4	56.5	**57.8** +2.0
Avg. Category	Action	84.8	86.0	86.4	85.9	71.4	72.8	72.9	73.1	**74.2** +2.8	91.2	91.7	91.8	91.7	**93.0** +1.8
	Direction	53.2	55.3	44.2	45.3	42.4	42.8	42.7	43.0	**44.0** +1.6	48.4	49.1	49.1	49.2	**50.7** +2.3
	Speed	61.3	57.4	45.8	47.7	41.9	42.7	42.8	42.9	**44.3** +2.4	50.5	50.7	50.9	50.9	**52.6** +2.1
	Event Order	70.7	76.2	53.0	55.6	45.7	46.9	47.2	47.3	**48.8** +3.1	68.0	68.9	68.8	68.7	**70.9** +2.9
	Attr. Change	63.0	59.1	53.3	58.0	45.7	46.9	46.9	46.8	**49.0** +3.3	67.0	68.1	68.2	68.1	**70.4** +3.4
	Overall	66.0	64.2	56.9	58.8	49.8	50.4	50.5	50.6	**52.1** +2.3	65.0	65.7	65.8	65.7	**67.5** +2.5

Table 2: Video temporal understanding evaluation on TempCompass. Best results are shown in **bold**.

Method	Temporal Perception	Temporal Reasoning	Short	Medium	Long	Overall
Video-LLaVA-7B [15]	-	-	45.3	38.0	36.2	39.9
LLaVA-NeXT-Video-7B-DPO [18]	40.0	29.4	48.9	42.0	35.6	42.1
Llama-3-VILA1.5-8B [16]	50.9	41.2	56.1	42.1	39.6	45.9
VILA1.5-40B [16]	60.0	40.7	72.0	61.2	53.8	62.3
InternVL-Chat-V1.5-20B [3]	45.5	33.3	60.2	46.4	45.6	50.7
LongVA-7B [33]	58.2	37.3	61.1	50.4	46.2	52.6
LLaVA-Video-7B-Qwen2 [35]	61.1	54.3	73.5	54.8	51.1	59.8
+ VCD [9]	75.4	55.3	73.9	55.2	50.8	60.0
+ SID [7]	75.3	55.4	74.0	55.1	50.6	59.9
+ TCD [32]	75.9	55.8	73.6	55.1	50.9	59.8
+ Ours	**84.1**	**57.8**	**75.3**	**57.2**	**52.2**	**61.6**

Table 3: Video-MME evaluation results. Our method enhances LLaVA-Video accuracy across various video durations, even outperforming VILA1.5-40B in temporal reasoning. Best results are shown in **bold**.

Ratio—we find that setting the values too low results in limited improvements, while excessively high values, i.e., severe distortion, lead to a relative decline in performance. This aligns with our earlier analysis: overly severe distortion tends to randomize the model's responses, thereby undermining its role as a negative response to guide the generation in contrastive decoding. Only appropriately calibrated distortion can effectively induce negative responses, thereby enhancing performance via contrastive decoding.

Is There a Trade-off Between Temporal and Spatial Understanding? Intuitively, when humans watch a video and focus on temporal changes in the scene, their attention to static details tends to decline. Interestingly, we observe a similar phenomenon in Video LLMs. On the MLVU benchmark, while our performance improves notably on temporally related tasks such as AC and AO, it drops on

Model	AC*	ER	Needle QA	AO*	Plot QA	AR	TR	Overall
Video-ChatGPT-7B [23]	31.1	42.0	40.3	25.1	29.9	24.0	26.9	31.3
Video-LLaVA-7B [31]	35.9	45.2	53.2	20.1	48.4	57.0	71.6	47.3
MA-LMM-7B [6]	24.3	38.9	43.1	25.1	35.8	35.5	51.9	36.4
Llama-3-VILA1.5-8B [16]	0.0	24.7	32.4	6.6	20.0	27.0	46.2	22.4
VILA1.5-40B [16]	11.7	35.8	38.3	34.3	62.0	56.4	84.7	46.2
InternVL-Chat-V1.5-20B [3]	13.3	24.5	40.0	14.3	42.0	51.3	80.2	37.9
LongVA-7B [33]	25.2	48.6	70.4	41.7	68.1	58.5	82.2	56.4
LLaVA-Video-7B-Qwen2 [35]	41.8	68.5	76.3	57.9	75.1	**67.8**	**84.5**	67.4
+ VCD [9]	42.8	68.5	77.0	60.1	75.1	65.1	82.8	67.3
+ SID [7]	42.9	68.5	77.2	60.1	75.1	65.2	82.7	67.4
+ TCD [32]	42.3	68.6	76.7	60.2	75.2	65.0	83.0	67.3
+ **Ours**	**44.1**	**68.8**	**78.5**	**62.6**	**75.8**	65.7	83.8	**68.5**

Table 4: MLVU evaluation results. Our method achieves the best overall performance, with notable gains in temporal-related aspects. TR: Topic Reasoning, AR: Anomaly Recognition, ER: Ego Reasoning, AO: Action Order, AC: Action Count. * denotes temporal-related dimensions. Best results are in **bold**.

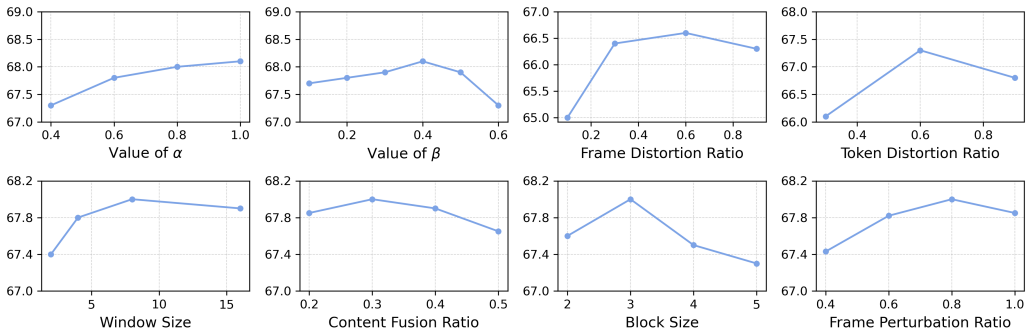

Figure 5: Sensitivity to Hyperparameter Settings on TempCompass.

AR and TR. We further conduct an experiment on TempCompass, where we progressively increase the Token Distortion Ratio and the Frame Distortion Ratio, respectively. As shown in the line plot, the accuracy of the sub-task "Attribute Change" keeps increasing, indicating an improvement in temporal perception. However, the overall accuracy (represented by the bar chart) begins to decline when the distortion ratio is approximately 0.6, suggesting that other sub-tasks are negatively affected.

Method	TempCompass	EventHallusion
Vanilla	67.5	72.1
- Attention Guidance	65.7	67.1
- Key Frame Fusion	66.1	66.3
- Key Token Distortion	66.5	68.3
- Moving Content Disruption	66.9	69.7

Table 5: Ablation Study Results on Temporal Benchmarks: TempCompass and EventHallusion.

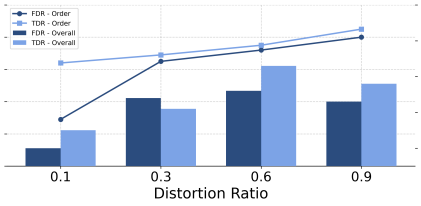

Figure 6: Influence of Distortion

5 Related Work

Video Large Language Models. Multimodal Large Language Models (MLLM) [20, 25, 26] are evolving rapidly, advancing image-text dialogue through fine-tuning pre-trained Large Language Models (LLM) with image features from additional visual encoders. Video LLMs extend MLLMs from image to video understanding by involving encoded video frames during training, such as Video-LLaVA [14], VILA-series [22, 1, 16] and LLaVA-NeXT-video [34]. To include stronger temporal information in video representations, recent works [31, 24] design additional temporal-aware encoders or customize their own training data with time-stamp annotations. Similarly, Li et al. [11] also improve

temporal reasoning with additional data but sourced from text. In contrast to existing works, our method enhances temporal reasoning in videos by incorporating a new temporal-aware decoding strategy to avoid time-insensitive responses at no additional training cost.

Temporal Understanding in Video LLMs. Temporal understanding is fundamental in Video LLMs. While existing popular benchmarks such as Video-MME [5] and MLVU [36] focus on the general evaluation of Video LLMs across diverse video categories and durations, TempCompass [21] specifically focuses on the temporal reasoning ability of Video LLMs with a variety of temporally focused tasks. EventHallusion [32] investigates Video LLMs' capability to understand continuous events. Our method not only achieves notable improvements on temporal understanding tasks, but also demonstrates promising results on general video benchmarks.

Contrastive Decoding. Contrastive Decoding (CD) [12] is a search-based LLM decoding approach that improves text generation by explicitly avoiding poor responses during decoding. Recent works [9, 7] explore its application in reducing hallucination in MLLMs by deliberately introducing distorted image inputs to elicit poor responses, which are then used to guide the model away from such errors. TCD [32] inherits similar idea to avoid video event hallucination by randomly dropping frames. Different from existing works, our work particularly focuses on improving temporal reasoning in Video LLMs. Extended discussions are available in Appendix E.

6 Limitations

When performing video distortion, our Temporal Distortion Unit relies solely on signals from the model itself—specifically, the attention maps extracted from the intermediate LLM layers—as guidance to estimate the importance of each visual token and each video frame. Compared to treating all frames equally and applying uniform random sampling, our approach represents a significant improvement. However, it is still not perfect. Attention maps do not always accurately reflect the true importance of each visual token, and relying on them often yields only coarse-grained results. To more precisely assess the importance of visual representations, future work may explore more accurate and robust methods beyond attention-based guidance.

Moreover, our current study is limited to Video LLMs, with distortion applied only to the visual representations. In practice, many videos come with accompanying subtitles, and models often take both video and subtitle inputs. An interesting future direction would be to distort both modalities—applying not only visual distortion but also video-aware distortion to subtitles. This would be challenging and different from the purely text-based distortion strategies employed in existing works on contrastive decoding for LLMs.

7 Conclusion

In this work, we investigated the challenges of temporal reasoning in Video Large Language Models (Video LLMs) and identified two key factors contributing to their failures—language prior and image prior. Building on these insights, we proposed video contrastive decoding with temporal distortion, a simple yet effective method that enhances temporal coherence without requiring additional training or computational overhead. By intentionally introducing temporal distortions in key frames and contrastively optimizing against such failures, our method encourages models to maintain temporal consistency and avoid time-insensitive predictions. Extensive experiments demonstrate that our approach significantly improves both temporal-specific and general video understanding benchmarks, showing strong effectiveness, generalizability, and scalability for improving temporal reasoning in multimodal large language models.

Acknowledgment

The work is supported in part by the U.S. Office of Naval Research Award under Grant Number N00014-24-1-2668, the National Science Foundation under Grants IIS-2316306 and CNS-2330215, the National Institutes of Health (NIH) under Grant R01EB293388, and gifts from Adobe Research.

References

[1] Yukang Chen, Fuzhao Xue, Dacheng Li, Qinghao Hu, Ligeng Zhu, Xiuyu Li, Yunhao Fang, Haotian Tang, Shang Yang, Zhijian Liu, Ethan He, Hongxu Yin, Pavlo Molchanov, Jan Kautz, Linxi Fan, Yuke Zhu, Yao Lu, and Song Han. Longvila: Scaling long-context visual language models for long videos, 2024.

[2] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24185–24198, 2024.

[3] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. *arXiv preprint arXiv:2312.14238*, 2023.

[4] Alessandro Favero, Luca Zancato, Matthew Trager, Siddharth Choudhary, Pramuditha Perera, Alessandro Achille, Ashwin Swaminathan, and Stefano Soatto. Multi-modal hallucination control by visual information grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. arXiv:2403.14003.

[5] Chaoyou Fu, Yuhan Dai, Yondong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, et al. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. *arXiv preprint arXiv:2405.21075*, 2024.

[6] Bo He, Hengduo Li, Young Kyun Jang, Menglin Jia, Xuefei Cao, Ashish Shah, Abhinav Shrivastava, and Ser-Nam Lim. Ma-lmm: Memory-augmented large multimodal model for long-term video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13504–13514, 2024.

[7] Fushuo Huo, Wenchao Xu, Zhong Zhang, Haozhao Wang, Zhicheng Chen, and Peilin Zhao. Self-introspective decoding: Alleviating hallucinations for large vision-language models. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2025. Published as a conference paper at ICLR 2025.

[8] Peng Jin, Ryuichi Takanobu, Caiwan Zhang, Xiaochun Cao, and Li Yuan. Chat-univi: Unified visual representation empowers large language models with image and video understanding. *arXiv preprint arXiv:2311.08046*, 2023.

[9] Sicong Leng, Hang Zhang, Guanzheng Chen, Xin Li, Shijian Lu, Chunyan Miao, and Lidong Bing. Mitigating object hallucinations in large vision-language models through visual contrastive decoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[10] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024.

[11] Lei Li, Yuanxin Liu, Linli Yao, Peiyuan Zhang, Chenxin An, Lean Wang, Xu Sun, Lingpeng Kong, and Qi Liu. Temporal reasoning transfer from text to video. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2025.

[12] Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori Hashimoto, Luke Zettlemoyer, and Mike Lewis. Contrastive decoding: Open-ended text generation as optimization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, Toronto, Canada, 2023. Main conference long paper.

[13] Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning united visual representation by alignment before projection. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5971–5984, Miami, Florida, USA, November 2024. Association for Computational Linguistics.

[14] Bin Lin, Bin Zhu, Yang Ye, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning united visual representation by alignment before projection. *arXiv preprint arXiv:2311.10122*, 2023.

[15] Bin Lin, Bin Zhu, Yang Ye, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning united visual representation by alignment before projection. *arXiv preprint arXiv:2311.10122*, 2023.

[16] Ji Lin, Hongxu Yin, Wei Ping, Yao Lu, Pavlo Molchanov, Andrew Tao, Huizi Mao, Jan Kautz, Mohammad Shoeybi, and Song Han. Vila: On pre-training for visual language models. 2023. Preprint.

[17] Zhiqiu Lin, Xinyue Chen, Deepak Pathak, Pengchuan Zhang, and Deva Ramanan. Revisiting the role of language priors in vision-language models. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, 2024. arXiv:2306.01879.

[18] Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge. 2024. Preprint.

[19] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning, 2023.

[20] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024.

[21] Yuanxin Liu, Shicheng Li, Yi Liu, Yuxiang Wang, Shuhuai Ren, Lei Li, Sishuo Chen, Xu Sun, and Lu Hou. Tempcompass: Do video llms really understand videos? *arXiv preprint arXiv:2403.00476*, 2024.

[22] Zhijian Liu, Ligeng Zhu, Baifeng Shi, Zhuoyang Zhang, Yuming Lou, Shang Yang, Haocheng Xi, Shiyi Cao, Yuxian Gu, Dacheng Li, Xiuyu Li, Yunhao Fang, Yukang Chen, Cheng-Yu Hsieh, De-An Huang, An-Chieh Cheng, Vishwesh Nath, Jinyi Hu, Sifei Liu, Ranjay Krishna, Daguang Xu, Xiaolong Wang, Pavlo Molchanov, Jan Kautz, Hongxu Yin, Song Han, and Yao Lu. Nvila: Efficient frontier visual language models, 2024.

[23] Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. Video-chatgpt: Towards detailed video understanding via large vision and language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL 2024)*, 2024.

[24] Ming Nie, Dan Ding, Chunwei Wang, Yuanfan Guo, Jianhua Han, Hang Xu, and Li Zhang. Slowfocus: Enhancing fine-grained temporal understanding in video llm. In *Proceedings of the 38th Conference on Neural Information Processing Systems (NeurIPS)*, 2024. `https://github.com/fudan-zvg/SlowFocus`.

[25] Daiqing Qi, Handong Zhao, Jing Shi, Simon Jenni, Yifei Fan, Franck Dernoncourt, Scott Cohen, and Sheng Li. The photographer's eye: Teaching multimodal large language models to see, and critique like photographers. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pages 24807–24816, June 2025.

[26] Daiqing Qi, Handong Zhao, Zijun Wei, and Sheng Li. Tag-grounded visual instruction tuning with retrieval augmentation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 2008–2026, 2024.

[27] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

[28] Shuhuai Ren, Linli Yao, Shicheng Li, Xu Sun, and Lu Hou. Timechat: A time-sensitive multimodal large language model for long video understanding. *arXiv preprint arXiv:2312.02051*, 2023.

[29] Reuben Tan, Ximeng Sun, Ping Hu, Jui-hsien Wang, Hanieh Deilamsalehy, Bryan A. Plummer, Bryan Russell, and Kate Saenko. Koala: Key frame-conditioned long video-llm. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13581–13591, 2024.

[30] Yolo Yunlong Tang, Jing Bi, Pinxin Liu, Zhenyu Pan, Zhangyun Tan, Qianxiang Shen, Jiani Liu, Hang Hua, Junjia Guo, Yunzhong Xiao, Chao Huang, Zhiyuan Wang, Susan Liang, Xinyi Liu, Yizhi Song, Yuhe Nie, Jia-Xing Zhong, Bozheng Li, Daiqing Qi, Ziyun Zeng, Ali Vosoughi, Luchuan Song, Zeliang Zhang, Daiki Shimada, Han Liu, Jiebo Luo, and Chenliang Xu. Video-lmm post-training: A deep dive into video reasoning with large multimodal models, 2025.

[31] Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint arXiv:2306.02858*, 2023.

[32] Jiacheng Zhang, Yang Jiao, Shaoxiang Chen, Jingjing Chen, and Yu-Gang Jiang. Eventhallusion: Diagnosing event hallucinations in video llms. In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence (AAAI)*. Association for the Advancement of Artificial Intelligence, 2025. To appear.

[33] Peiyuan Zhang, Kaichen Zhang, Bo Li, Guangtao Zeng, Jingkang Yang, Yuanhan Zhang, Ziyue Wang, Haoran Tan, Chunyuan Li, and Ziwei Liu. Long context transfer from language to vision. *arXiv preprint arXiv:2406.16852*, 2024.

[34] Yuanhan Zhang, Bo Li, haotian Liu, Yong jae Lee, Liangke Gui, Di Fu, Jiashi Feng, Ziwei Liu, and Chunyuan Li. Llava-next: A strong zero-shot video understanding model, April 2024.

[35] Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. Video instruction tuning with synthetic data. *arXiv preprint arXiv:2410.02713*, 2024.

[36] Junjie Zhou, Yan Shu, Bo Zhao, Boya Wu, et al. Mlvu: A comprehensive benchmark for multi-task long video understanding. *arXiv preprint arXiv:2406.04264*, 2024.

A Adaptive Plausibility Constraint

Contrastive Decoding in Video Large Language Models. Given a text query $\mathbf{x}$ and a video input $\mathbf{V}$, the model generates two output distributions: one conditioned on the original $\mathbf{V}$ and the other on the distorted video input $\mathbf{V}'$, which is derived by applying pre-defined distortion (e.g., adding noise to visual features as the simplest case) to $\mathbf{V}$. Then, a new contrastive probability distribution is computed by leveraging the differences between two original distributions. The new contrastive distribution p_{vtd} is formulated as:

$$p_{\text{vtd}}(\mathbf{y} \mid \mathbf{V}, \mathbf{V}', \mathbf{x}) = \text{softmax}\left[(1+\alpha)\text{logit}_\theta(\mathbf{y} \mid \mathbf{V}, \mathbf{x}) - \alpha\text{logit}_\theta(\mathbf{y} \mid \mathbf{V}', \mathbf{x})\right], \tag{3}$$

where larger α indicate a stronger amplification of the differences ($\alpha = 0$ reduces to regular decoding).

Adaptive Plausibility Constraint. Eq. 3 rewards texts favored by the response with original video inputs and penalizes texts favored by the response with distorted video inputs. However, the response with distorted video inputs is not always mistaken. Although video inputs are distorted, they may still preserve useful information, which can lead to correct answers. Therefore, penalizing all texts from response with distorted video inputs indiscriminately would penalize these correct answers, and conversely reward implausible answers. To tackle this issue, we follow Li et al. [12] to introduce the plausibility constraint.

Adaptive plausibility constraint is contingent upon the confidence level associated with the output distribution with original video inputs:

$$\mathcal{V}_{\text{head}}(\mathbf{y}_{<t}) = \left\{\mathbf{y}_t \in \mathcal{V} : p_\theta(\mathbf{y}_t \mid \mathbf{V}, \mathbf{x}, \mathbf{y}_{<t}) \geq \beta \max_{\mathbf{w}} p_\theta(\mathbf{w} \mid \mathbf{V}, \mathbf{x}, \mathbf{y}_{<t})\right\} \tag{4}$$

$$p_{\text{vtd}}(\mathbf{y}_t \mid \mathbf{V}, \mathbf{V}', \mathbf{x}) = 0, \quad \text{if } \mathbf{y}_t \notin \mathcal{V}_{\text{head}}(\mathbf{y}_{<t}) \tag{5}$$

where $\mathcal{V}$ is the output vocabulary of LVLMs and β is a hyperparameter in $[0, 1]$ for controlling the truncation of the next token distribution. Larger β indicates more aggressive truncation, keeping only high-probability tokens.

Combining the video contrastive decoding and the adaptive plausibility constraint, we obtain the full formulation:

$$\begin{gathered}\mathbf{y}_t \sim \text{softmax}\left[(1+\alpha)\,\text{logit}_\theta(\mathbf{y}_t \mid \mathbf{V}, \mathbf{x}, \mathbf{y}_{<t}) - \alpha\,\text{logit}_\theta(\mathbf{y}_t \mid \mathbf{V}', \mathbf{x}, \mathbf{y}_{<t})\right] \\ \text{subject to } \mathbf{y}_t \in \mathcal{V}_{\text{head}}(\mathbf{y}_{<t})\end{gathered} \tag{6}$$

B Technical Details of Video Temporal Distortion

B.1 Disrupting Moving Content in Remaining Frames

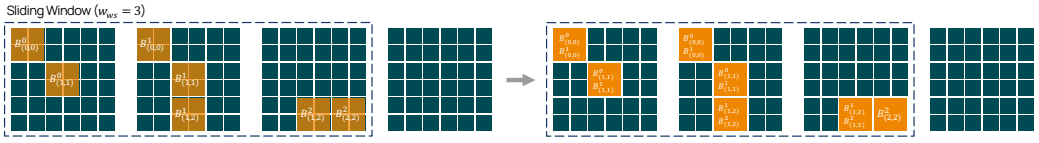

Figure 7: Disrupt moving content in remaining frames within a sliding window. **Left:** Marked dynamic blocks. **Right:** Fusion results of marked dynamics.

Downsampling. Note that in the downsampling stage, i.e., when we downsample each frame from size (H, W) to $(\frac{H}{w_{\text{bs}}}, \frac{W}{w_{\text{bs}}})$, *we do not really reduce token numbers*. As illustrated in Fig. 7 (left), we actually replace all image tokens within one "downsampled" region with the mean of the tokens included. For example, the value of each of the four token within $\mathbf{B}^0_{(0,0)}$ is the mean of the four tokens.

C Experiments

C.1 Experimental Configuration

In **TempCompass** [21], we use the following hyperparameters: $\alpha = 1$, $\beta = 0.2$, $w_{\text{fdr}} = 0.2$, $w_{\text{tdr}} = 0.4$, $w_{\text{ws}} = 8$, $w_{\text{cfr}} = 0.3$, $w_{\text{bs}} = 3$, $w_{\text{fpr}} = 0.5$, $w_{\text{momentum}} = 0.8$. In **EventHallusion** [32], we use the following hyperparameters: $\alpha = 1$, $\beta = 0.2$, $w_{\text{fdr}} = 0.5$, $w_{\text{tdr}} = 0.5$, $w_{\text{ws}} = 8$, $w_{\text{cfr}} = 0.3$, $w_{\text{bs}} = 3$, $w_{\text{fpr}} = 0.8$, $w_{\text{momentum}} = 0.8$. In **Video-MME** [5] and **MLVU** [36], we use the following hyperparameters: $\alpha = 1$, $\beta = 0.2$, $w_{\text{fdr}} = 0.6$, $w_{\text{tdr}} = 0.8$, $w_{\text{ws}} = 8$, $w_{\text{cfr}} = 0.3$, $w_{\text{bs}} = 3$, $w_{\text{fpr}} = 0.8$, $w_{\text{momentum}} = 0.5$.

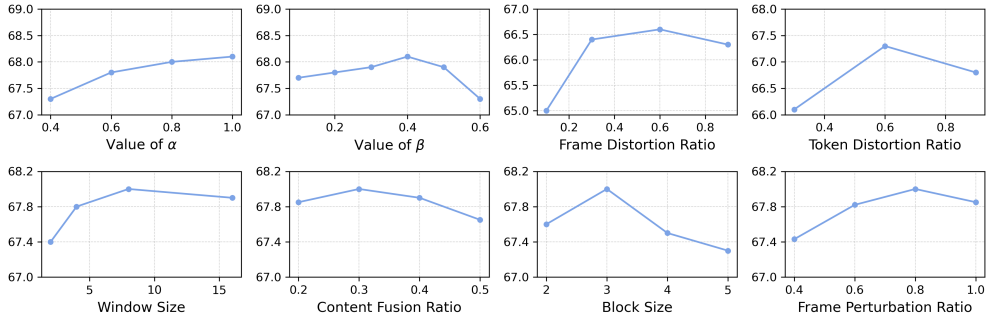

Figure 8: Sensitivity to Hyperparameter Settings on TempCompass [21].

C.2 Analysis

From Fig. 8, we observe that a moderate level of distortion is crucial for effective contrastive decoding. In the ablation study of parameters most closely related to the degree of distortion—such as Frame Distortion Ratio, Token Distortion Ratio, Content Fusion Ratio, and again Frame Distortion Ratio—we find that setting the values too low results in limited improvements, while excessively high values, i.e., severe distortion, lead to a relative decline in performance. This aligns with our earlier analysis: overly severe distortion tends to randomize the model's responses, thereby undermining its role as a negative response to guide the generation in contrastive decoding. Only appropriately calibrated distortion can effectively induce negative responses, thereby enhancing performance via contrastive decoding.

D Extended Discussion

We are among the first to explore video temporal understanding from the perspective of language and image priors, and to enhance it using contrastive decoding with video temporal distortion.

We are among the first to explore video temporal understanding from the perspective of language and image priors, and to enhance it using contrastive decoding with video temporal distortion.

Recent works [9, 7] have applied contrastive decoding to mitigate hallucinations in image understanding with MLLMs. For example, VCD [9] introduces random noise to distort the original image, while SID [7] prunes important tokens based on attention guidance. TCD [32] alleviates event hallucination in videos by randomly dropping frames.

SID [7] adopts a similar strategy to estimate token importance and removes the most important tokens—this is conceptually similar to the second step of our video temporal distortion. However, there are notable differences in how attention maps are utilized and how the pruning is applied. Specifically, SID uses attention maps from the k-th layer to assess token importance and then prunes the most important tokens starting from the $(k+1)$-th layer.

In contrast, our approach aggregates attention maps from all layers, from shallow to deep, to compute a more accurate importance score. Furthermore, while SID performs pruning from intermediate layers, we input the distorted video representations directly at the first layer, ensuring that the dropped

information is effectively masked from the very beginning. This design allows our method to better mask information that should be dropped.

E Efficiency

Due to the nature of Contrastive Decoding (which requires two forward passes), CD-based methods (e.g., VCD and ours) are inevitably slower than the vanilla model in generation speed.

In practical scenarios, it's difficult to optimize performance, time efficiency, and memory efficiency all at once. Different applications prioritize different aspects. CD-based methods are less time-efficient, but they offer better performance without a significant increase in memory usage. This makes them well-suited for applications where content quality is critical, such as education and medical assistance, where accuracy really matters and errors come at a high cost.

Time-Efficient Implementation. In practice, we can optimize the code to reduce the time by nearly half with no impact on the results, making it as efficient as the vanilla Video LLM.

The implementation is simple: we parallelize the two forward passes during next-token generation. Previously, it is implemented in the `def sample()` function as follows:

```
outputs = self.forward(...)       # inference with raw video
outputs_cd = self.forward(...)    # inference with distorted video

# calculate final logits with outputs and outputs_cd:
...
```

Now, with just a few extra lines, we use `torch.cuda.Stream()` to run both forward passes in parallel:

```
stream_main = torch.cuda.Stream()
stream_cd = torch.cuda.Stream()
outputs_holder = {}
outputs_cd_holder = {}

# submit main inference
with torch.cuda.stream(stream_main):
    outputs_holder['main'] = self.forward(...)

# submit contrastive inference
with torch.cuda.stream(stream_cd):
    outputs_cd_holder['cd'] = self.forward(...)

torch.cuda.synchronize()

outputs = outputs_holder['main']
outputs_cd = outputs_cd_holder['cd']

# calculate final logits with outputs and outputs_cd:
...
```

With this implementation, our method can be as fast as the vanilla Video LLM with the same average GPU memory usage. Note that the peak GPU memory usage will be larger.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: they are in the abstract and introduction

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: In Appendix

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA] .

   Justification: no theoretical result

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: included in appendix

   Guidelines:

   - The answer NA means that the paper does not include experiments.

- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [NA] .

   Justification: publish when accepted

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: in paper and appendix

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [NA] .

   Justification: doesnt apply

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: in paper and appendix

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.

- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: we followed it

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: no societal impact of the work performed.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: paper poses no such risks

    Guidelines:

    - The answer NA means that the paper poses no such risks.

- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: yes

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: no new assets

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: the paper does not involve crowdsourcing

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: the paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: the core method development in this research does not705 involve LLMs as any important, original, or non-standard components.706

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.