Test-Time Consistency in Vision Language Models

Anonymous ACL submission

Abstract

Vision-Language Models (VLMs) have achieved impressive performance across a wide range of multimodal tasks, yet they often exhibit inconsistent behavior when faced with semantically equivalent inputs—undermining their reliability and robustness. Recent benchmarks, such as MM-R^3, highlight that even state-of-the-art VLMs can produce divergent predictions across semantically equivalent inputs, despite maintaining high average accuracy. Prior work addresses this issue by modifying model architectures or conducting large-scale fine-tuning on curated datasets. In contrast, we propose a simple and effective *test-time consistency framework* that enhances semantic consistency *without supervised re-training*. Our method is entirely *post-hoc*, model-agnostic, and applicable to any VLM with access to its weights. Given a single test point, we enforce consistent predictions via two complementary objectives: (i) a **Cross-Entropy Agreement Loss** that aligns predictive distributions across semantically equivalent inputs, and (ii) a **Pseudo-Label Consistency Loss** that draws outputs toward a self-averaged consensus. Our method is *plug-and-play*, and leverages information from a single test-input itself to improve consistency. Experiments on the MM-R^3 benchmark show that our framework yields substantial gains in consistency across state-of-the-art models, establishing a new direction for inference-time adaptation in multimodal learning.

1 Introduction

Vision-Language Models (VLMs) (Liu et al., 2024a,b; Wang et al., 2024; Hurst et al., 2024) have achieved impressive performance across a wide range of multimodal tasks, including visual question answering (Antol et al., 2015), captioning (Lin et al., 2014; Sharma et al., 2018; Chen et al., 2015), and reasoning (Johnson et al., 2017; Zellers et al., 2019). While existing evaluations predominantly focus on accuracy, a growing body of work highlights a critical shortcoming: *semantic inconsistency*. That is, VLMs producing divergent outputs when prompted with semantically equivalent inputs—undermining their reliability, interpretability, and deployment in high-stakes settings.

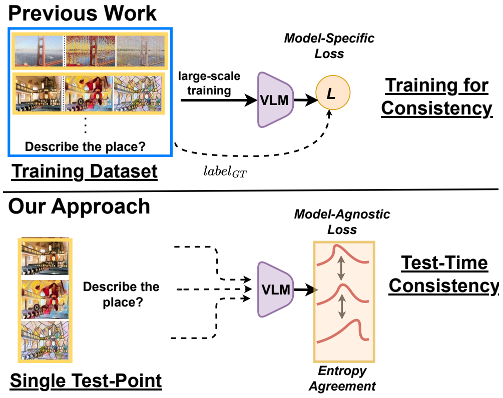

Figure 1: Comparison between **training-time** (top) and our **test-time** (bottom) consistency frameworks. While prior work (*e.g.*, (Chou et al., 2024)) needs large-scale supervised fine-tuning with curated dataset to enforce consistency, our method operates entirely post-hoc by adapting to a single test point with few gradient steps at test-time

Despite achieving strong accuracy on vision-language tasks, modern VLMs often yield inconsistent responses when presented with semantically equivalent variants of a test query. The MM-R^3 benchmark, recently proposed in Chou et al. (2024), highlights this issue by evaluating models under three types of controlled perturbations: *question rephrasing*, *image restyling*, and *context masking*. Results reveal that even state-of-the-art models show significant variance across these conditions, illustrating that high accuracy does not imply semantic consistency—a fundamental prerequisite for robust multimodal reasoning. While Chou et al. (2024) address consistency via adapter-based fine-tuning, their approach requires intrusive architectural modifications and access to a sizable training dataset.

In contrast, we study the problem of improving ***consistency at test time***—using only the test input itself, without any access to model internals, training data, or loss functions. Unlike prior methods that rely on supervised re-training, additional curated datasets, or adapter insertion (Chou et al., 2024), our framework makes *no assumptions about training* and operates entirely at inference. This is especially important in real-world where retraining is infeasible due to proprietary models, limited compute, or lack of access to original training data.

We propose a simple, general-purpose test-time consistency framework that can be applied to any probabilistic vision-language model (VLM) in a *plug-and-play* fashion. Our method leverages semantically equivalent variants of a given input and encourages agreement among the resulting predictions using two lightweight objectives: (1) a Cross-Entropy Agreement Loss, which penalizes divergence in output distributions, and (2) a Pseudo-Label Consistency Loss, which aligns predictions toward a consensus output. Crucially, our method requires no modifications to the model architecture and operates entirely at inference time. Intuitively, it encourages the model to generate invariant predictions across different linguistic and visual realizations of the same query, thereby promoting robustness and semantic stability.

Importantly, our method adapts model behavior at test time—even for a *single* input—by utilizing the information embedded in that input's semantic variations. This departs from training-centric paradigms and instead exploits the rich signal present in the test data itself, which prior work often overlooks. Because our framework requires no access to original training data, loss functions, or model internals, and avoids retraining or auxiliary supervision, it can be seamlessly integrated into any VLM pipeline regardless of architecture.

We evaluate our approach on the standard MM-R^3 benchmark and demonstrate substantial improvements in consistency across multiple open-source and proprietary VLMs. Our results show that even strong models benefit from targeted inference-time regularization, and we advocate for consistency—as well as accuracy—to be a central design goal in future multimodal learning systems.

Our work makes the following **contributions**:

- We address the underexplored problem of *test-time consistency* in VLMs by proposing a simple, model-agnostic framework that improves consistency in VLM response across semantically equivalent inputs. Our method operates entirely *post-hoc*—requiring only access to model weights and using information derived solely from a *single test input*. It requires no training dataset, no access to original loss functions or training procedures, and no supervised retraining—making it broadly applicable across models and practical deployment settings.
- Our framework leverages two complementary objectives: (1) a Cross-Entropy Agreement Loss to reduce divergence among predictions on perturbed inputs, and (2) a Pseudo-Label Consistency Loss to align predictions towards a consensus output. Unlike prior work focused on training-time consistency or fine-tuning, our method makes no assumptions about model training, and is *plug-and-play*, requiring only access to model outputs.
- We show that our framework significantly improves consistency across linguistic and visual perturbations in the MM-R^3 benchmark without retraining or architectural changes. Our results highlight that even a single test point contains valuable signal that can be used to adapt model behavior at inference, offering a new paradigm for robust multimodal reasoning.

2 Related Works

Consistency in Vision-Language Models. While existing evaluations of Vision-Language Models (VLMs) predominantly focus on accuracy, a growing body of work highlights a critical shortcoming: *semantic inconsistency* (Chou et al., 2024). That is, VLMs often produce divergent outputs when prompted with semantically equivalent inputs—undermining their reliability, interpretability, and applicability in high-stakes settings. The MM-R^3 benchmark (Chou et al., 2024) systematically investigates this issue, introducing a suite of perturbation-based evaluations across rephrased questions, stylized images, and masked contexts. Their results show that even state-of-the-art VLMs exhibit significant inconsistency across these settings, despite high accuracy—revealing a fundamental gap between correctness and stable reasoning.

While prior efforts to improve consistency (Chou et al., 2024) typically focus on modifying training objectives, leveraging larger models, or fine-tuning on curated data, these approaches are computationally intensive and often impractical.

In contrast, we address this challenge from a test-time perspective, proposing a lightweight, post-hoc framework that improves consistency without retraining or access to labels.

Test-Time Adaptation. Test-time adaptation methods have progressed from entropy-based confidence maximization to efficient modular tuning. MEMO (Zhang et al., 2022) improves robustness by enforcing confident and consistent predictions across augmented test-time views. Test-Time Prompt Tuning (Shu et al., 2022) adapts CLIP by optimizing prompts at inference to better match shifted distributions. MedAdapter (Shi et al., 2024) steers pretrained LLMs toward domain-specific tasks by updating small adapter modules without full retraining. LoRA-TTT (Kojima et al., 2025) further reduces adaptation cost by fine-tuning low-rank adapters at test time. Karmanov et al. (2024) proposed a lightweight VLM adaptation strategy that freezes the core model and updates only a small projection head via entropy minimization. While most of these works focus on improving accuracy via test-time optimization, our work targets a complementary and underexplored axis: *semantic consistency*. Unlike methods that require retraining, or architectural modifications, our approach is entirely post-hoc, model-agnostic, and leverages information from a single test input—making it lightweight, scalable, and broadly applicable.

Pseudo-Labeling and Self-Training. Pseudo-labeling has been widely used in semi-supervised learning (Yarowsky, 1995; Lee et al., 2013; Berthelot et al., 2019; Sohn et al., 2020; Zhang et al., 2021), often paired with augmentations or confidence thresholds. In vision-language models, it has been employed to generate pseudo-captions (Yang et al., 2022), region–phrase alignments (Chou et al., 2022), and visual-language prototypes (Ali et al., 2025). We adopt a test-time variant of pseudo-labeling, aggregating model outputs across perturbed inputs into a self-consistent consensus—encouraging stability without requiring external supervision or retraining.

Entropy-Based Adaptation. Entropy minimization has been a foundational strategy for improving robustness under distribution shift. Grandvalet and Bengio (Grandvalet and Bengio, 2004) introduced it as a regularization objective for unlabeled data, and TENT (Wang et al., 2021) applied it for test-time adaptation by optimizing batch norm parameters. MEMO (Zhang et al., 2022) improved on this by combining entropy minimization with multi-view consistency during inference. Extensions to large models include entropy-guided generation in LLMs (Kuhn et al., 2023; Farquhar et al., 2024) and efficient test-time tuning for vision-language models by updating only lightweight projection heads (Karmanov et al., 2024). Our method builds on this line of work by extending entropy-based objectives to open-ended multimodal settings—not to improve accuracy, but to enhance semantic consistency under perturbations, an underexplored yet practically important aspect of reliability and interpretability in multimodal reasoning.

3 Approach

3.1 Problem Setting

We follow the procedure and settings defined in the MM-R^3 benchmark to evaluate consistency under diverse semantic variations. Given a test input $\mathbf{x} = (I, Q)$, the benchmark provides K semantically equivalent variants (I_k, Q_k) constructed via:

- *Question Rephrasing*: Paraphrased variants of Q generated using a language model keeps I fixed.
- *Image Restyling*: Stylized versions of I using neural style transfer (*e.g.*, Mosaic, Candy, Undie, and Grayscale) with Q not altered.
- *Context Reasoning*: Variants of I with different occlusions applied to a specific object region, while keeping Q once again fixed.

Each perturbed pair (I_k, Q_k) is passed through the VLM to obtain response distributions:

$$\mathbf{p}_k = \texttt{VLM}(I_k, Q_k), \quad \text{for } k = 1, \ldots, K \tag{1}$$

3.2 Method

Overview. We propose a lightweight, test-time strategy to improve the semantic consistency of Vision-Language Models (VLMs) by encouraging agreement across semantically equivalent variants of a single test input. Our method operates entirely post-hoc and leverages only the information present in the given test example. It performs a small number of inference-time updates (typically 1–4 steps), requiring no access to training data, ground-truth labels, or model internals.

Our approach combines two complementary objectives: (1) a **Cross-Entropy Agreement Loss** that aligns token-level output distributions across perturbed inputs, and (2) a **Pseudo-Label Consistency Loss** that enforces convergence toward a stable, consensus output prediction. These objectives guide the model to become more consistent at

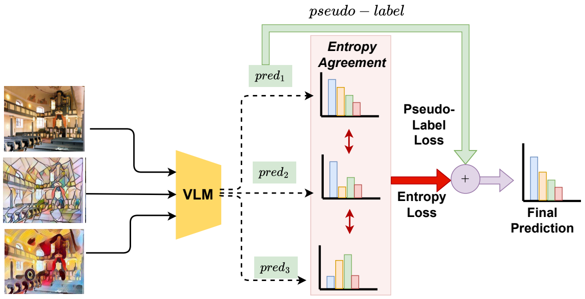

Figure 2: **Overview of our test-time consistency framework**. Given a test input with semantically equivalent input variants (*e.g.*, restyled images), we forward them through a pretrained VLM to obtain predictions. Two complementary objectives are used to improve consistency: (1) Cross-Entropy Agreement Loss, which aligns token-level output distributions across variants, and (2) Pseudo-Label Consistency Loss, which encourages agreement with a consensus pseudo-label. The model is updated with few (1-4) steps using gradients from these objectives, enabling consistent final predictions without access to training data or model internals.

inference, without altering its original architecture or parameters via supervised training.

3.3 Cross-Entropy Agreement Loss

To promote consistency across semantically equivalent input variants, we introduce a Cross-Entropy Agreement Loss that aligns their token-level output distributions. Given a test input, we generate VLM output for K perturbed variants and obtain token-level logits for each through a forward pass.

Let $\mathbf{z}_k^j \in \mathbb{R}^V$ denote the logits over the vocabulary V at output token position j of the VLM response for the k-th input variant. Let L_k be the total number of valid output tokens in response for that variant. We compute the average logits across the decoded sequence for each variant k:

$$\bar{\mathbf{z}}_k = \frac{1}{L_k} \sum_{j=1}^{L_k} \mathbf{z}_k^j \tag{2}$$

We then apply softmax to obtain the normalized token distribution:

$$\mathbf{p}_k = \mathrm{softmax}(\bar{\mathbf{z}}_k) \tag{3}$$

The agreement loss is defined as the average of all pairwise symmetric cross-entropies across the K output distributions:

$$\mathcal{L}_{\mathrm{CE}} = \frac{2}{K(K-1)} \sum_{i<j} \mathrm{CE}(\mathbf{p}_i, \mathbf{p}_j) + \mathrm{CE}(\mathbf{p}_j, \mathbf{p}_i) \tag{4}$$

This loss encourages alignment of the global output tokens across K input variants while ensuring the model's generation is *distributionally consistent*, even if wording or phrasing changes.

3.4 Pseudo-Label Consistency Loss

To complement distributional alignment, we introduce a Pseudo-Label Consistency Loss that enforces consistency at the output level by aligning each variant's predicted sequence to a common consensus output prediction.

Let $\{\mathbf{y}_1, \ldots, \mathbf{y}_K\}$ be the decoded textual outputs from the K semantically equivalent input variants, generated using greedy decoding. To compute a consensus label, we define a string similarity function $\mathrm{sim}(\cdot, \cdot)$ based on normalized Levenshtein distance (*e.g.*, token set ratio). We cluster the K output responses by assigning two responses $\mathbf{y}_i$ and $\mathbf{y}_j$ to the same cluster if

$$\mathrm{sim}(\mathbf{y}_i, \mathbf{y}_j) \geq \tau, \tag{5}$$

where $\tau \in [0, 1]$ is a fixed similarity threshold (*i.e.*, $\tau = 0.85$). Among all clusters, we identify the largest one, and from within it, select the most frequent response as the pseudo-label:

$$\hat{\mathbf{y}}_{\mathrm{pseudo}} = \mathrm{mode}\left(\mathcal{C}_{\mathrm{max}}\right), \tag{6}$$

where $\mathcal{C}_{\mathrm{max}}$ is the largest similarity-based cluster.

We then tokenize $\hat{\mathbf{y}}_{\mathrm{pseudo}}$ and use it as the supervision target for all K variants. Let $\mathbf{p}_k$ denote the token-level predicted distribution from variant k (*i.e.*, the model's output logits after softmax). The *Pseudo-Label Consistency Loss* is defined as:

$$\mathcal{L}_{\mathrm{PL}} = \frac{1}{K} \sum_{k=1}^{K} \mathrm{CE}(\hat{\mathbf{y}}_{\mathrm{pseudo}}, \mathbf{p}_k), \tag{7}$$

where $\mathrm{CE}(\cdot, \cdot)$ denotes the cross-entropy loss between pseudo-label tokens and predicted distribution. This loss encourages all variants to converge to the dominant semantic response, enhancing answer-level consistency of perturbed inputs.

Complementarity of Losses. The Cross-Entropy Agreement Loss encourages *token-level alignment* by smoothing output distributions across input variants, while the Pseudo-Label Consistency Loss enforces *prediction-level convergence* by aligning decoded outputs with a dominant consensus response. Together, these losses regularize both the internal generation process and final output, yielding improved semantic consistency at test time using only the information in single-test point without modifying the underlying model.

3.5 Final Objective and Inference

Given a test input with K semantically equivalent variants (I_k, Q_k) (*e.g.*, via question rephrasing, image restyling, or context masking), we adapt the model using gradients from two complementary objectives: the Cross-Entropy Agreement Loss $\mathcal{L}_{\text{CE}}$ and the Pseudo-Label Consistency Loss $\mathcal{L}_{\text{PL}}$.

The total loss at each update step is computed as a weighted sum:

$$\mathcal{L}_{\text{total}} = \alpha \cdot \mathcal{L}_{\text{CE}} + \beta \cdot \mathcal{L}_{\text{PL}}, \tag{8}$$

where α and β are hyperparameters balancing distributional agreement and semantic convergence. We optimize this objective for a small number of gradient-based updates—typically between 1 and 4—using only the current test example, without access to any labeled data or training corpus.

Adaptive Step Selection Different test inputs may benefit from different numbers of adaptation steps—while some improve with a few updates, others may degrade due to over-adaptation. To address this, we introduce an adaptive mechanism that dynamically selects the optimal number of steps for each test point.

After each update step $t \in \{0, 1, \ldots, T\}$, we decode the model's output responses for the K input variants. To assess internal consistency, we compute the average pairwise token-set similarity—based on normalized Levenshtein distance—among the K decoded answers and the previously generated pseudo-label (used in the Pseudo-Label Consistency Loss). The similarity score for step t is given by:

$$\text{score}_t = \frac{1}{K(K-1)} \sum_{i<j} \text{sim}(a_i^t, a_j^t), \tag{9}$$

where a_i^t and a_j^t denote either one of the K decoded answers or the shared pseudo-label. The step t^* with the highest score is selected as the final output, reflecting the most consistent model behavior during adaptation.

$$t^* = \arg\max_t \text{score}_t. \tag{10}$$

This selection mechanism is fully unsupervised and relies solely on model outputs *without using ground-truth annotations*. It enables per-instance, test-time adaptation that is both robust and efficient, ensuring that predictions remain semantically consistent while avoiding over-updating.

Method Variants. We report results for two variants of our method. In the first, we use a fixed number of adaptation steps ($T = 2$) for all samples, which we refer to as **Test-time (constant T)**. In the second, we use the adaptive step selection mechanism described above to dynamically choose the optimal number of updates per input. We refer to this variant as **Test-time (adapt. T)**. This comparison allows us to assess the trade-offs between simplicity and input-specific adaptivity.

4 Experiments

Dataset. We evaluate our method on the standard MM-R^3 dataset (Chou et al., 2024), which consists of three test-time consistency tasks: **question rephrasing**, **image restyling**, and **context reasoning**. The *question rephrasing* task assesses whether VLMs produce consistent answers to semantically equivalent questions phrased differently. The *image restyling* task evaluates consistency under visual domain shifts by presenting stylized versions of the image. The *context reasoning* task tests the model's ability to reason under partial occlusion. Our evaluations are conducted on MM-R^3 test set.

Models. We evaluate our method on widely used state-of-the-art open-source Vision-Language Models (VLMs). Specifically, LLaVA 1.5M (Liu et al., 2024a) (`llava-v1.5-7b` version), LLaVA-Next (Liu et al., 2024b) (`llava-v1.6-mistral-7b` checkpoint), and Qwen2-VL (Wang et al., 2024) (`Qwen2-VL-7B-Instruct` variant). These are all strong VLM models, with Qwen2-VL broadly considered the strongest among the three,

Our choice of these models is motivated by the fact that these models are widely used as foundations for downstream applications and frequently serve as initialization points for developing more advanced VLMs. Since our method involves modifying model parameters at test time, we restrict our evaluation to open-source models and exclude proprietary systems such as GPT-4V or Gemini. Evaluating on these representative models enables us to assess the generality, practical utility, and broader impact of test-time consistency improvements across VLMs.

Implementation Details. Please refer to the Appendix A.3.

Evaluation Metrics. Since VLM responses are open-ended and linguistically diverse, we adopt evaluation metrics similar to those introduced in MM-R^3 (Chou et al., 2024), in order to capture both correctness and consistency. We briefly introduce the core evaluation metrics used to assess correctness and consistency; full metric definitions and implementation details are provided in Appendix.

Table 1: **Overall results.** We highlight our approach in orange color and the overall results in gray color. The best-performing method is in bold for each models.

	Models	Acc	$\mathbf{S_{GT}}$	Con	$\mathbf{S_C}$	O_{all}
Question Rephrasing	LLaVa 1.5M	36.18	62.96	48.55	64.10	52.73
	+ Constant T	38.00	65.05	77.67	84.65	63.03
	+ Adapt. T	39.58	65.10	79.11	86.10	**64.08**
	LLaVa-Next	42.89	64.89	49.18	65.69	55.61
	+ Constant T	44.48	68.74	83.39	88.47	68.25
	+ Adapt. T	44.74	68.67	85.18	89.92	**68.83**
	Qwen2-VL	66.72	79.69	65.78	76.16	72.07
	+ Constant T	70.79	82.66	90.44	93.52	83.66
	+ Adapt. T	72.14	83.27	93.6	95.64	**85.33**
Image Restyling	LLaVa 1.5M	9.61	34.85	18.96	56.91	28.03
	+ Constant T	12.09	35.62	20.14	59.01	29.77
	+ Adapt. T	17.94	40.15	33.90	64.46	**36.52**
	LLaVa-Next	17.57	41.47	55.34	71.36	40.27
	+ Constant T	18.99	42.49	88.25	91.25	45.80
	+ Adapt. T	18.71	42.52	91.85	93.16	**46.00**
	Qwen2-VL	21.13	39.25	61.67	75.85	41.96
	+ Constant T	22.60	42.32	98.30	98.97	48.85
	+ Adapt. T	22.58	46.40	99.14	99.45	**51.20**
Context Reasoning	LLaVa 1.5M	16.11	42.69	65.64	75.08	41.47
	+ Constant T	22.88	49.49	88.89	93.45	51.81
	+ Adapt. T	31.04	55.14	72.11	81.90	**55.26**
	LLaVa-Next	30.24	27.43	32.11	58.44	35.23
	+ Constant T	32.50	50.84	89.91	90.16	56.97
	+ Adapt. T	32.29	53.85	95.24	96.66	**59.45**
	Qwen2-VL	29.09	40.03	34.58	53.70	38.77
	+ Constant T	29.60	50.11	91.17	91.75	55.52
	+ Adapt. T	30.42	53.00	99.53	99.66	**58.80**

- **Accuracy (Acc):** Measures correctness using fuzzy string matching, accounting for minor lexical variations. A similarity threshold of 85 is used to determine a match.
- **Similarity with Ground Truth ($S_{\mathbf{GT}}$):** Computes semantic similarity between the model's response and the reference answer using BERT sentence embeddings, offering a more flexible alternative to exact match.
- **Consistency Accuracy (Con):** Evaluates semantic agreement across responses to semantically equivalent inputs. Responses are considered consistent if their pairwise similarity exceeds a threshold of 0.7.
- **Consistency Similarity ($S_{\mathbf{C}}$):** Computes the average pairwise similarity across all response variations, providing a smoother measure of output invariance.
- **Overall Score (O_{all}):** The harmonic mean of correctness and consistency metrics.

$$H_{mean}(mean(\mathbf{Acc}, \mathbf{S_{GT}}), mean(\mathbf{Con}, \mathbf{S_C})). \quad (11)$$

We use the harmonic mean to emphasize models that are balanced in both accuracy and consistency, as it penalizes performance when either component is low. This provides a unified measure of overall model quality.

4.1 Main Results

Overview. Table 1 presents the performance of our test-time consistency framework across three tasks in the MM-R^3 benchmark. We report results for each base model with two variants: **Test-time (constant T)** and **Test-time (adaptive T)**. Across all tasks, our method consistently improves semantic consistency and overall performance, with the adaptive variant yielding the best results.

Question Rephrasing. In the rephrasing task, our adaptive test-time method yields substantial gains in consistency and overall score across all three models while preserving accuracy. For instance, on LLaVA-1.5M, O_{all} improves from 52.73 (base) to 64.08, with consistency rising from 48.55 to 79.11 and **Acc** increasing from 36.18 to 39.58. LLaVA-Next and Qwen2-VL also show notable gains, with the adaptive variant achieving the best O_{all} for each model: 68.83 and 85.33, respectively. These results validate the ability of our method to enforce semantic invariance across linguistic perturbations without reducing accuracy.

Image Restyling. This task poses a significant domain shift challenge due to stylized visual inputs. Our method leads to especially large improvements in consistency for all models. On LLaVA-Next, consistency improves from 55.34 (base) to 91.85 (Test-time) and further to 91.85 (Adaptive), with O_{all} reaching 46.00. Qwen2-VL sees the highest performance overall, with the adaptive variant achieving O_{all} = 51.20 and nearly perfect consistency (99.14). These results demonstrate the robustness of our framework under visual perturbations.

Context Reasoning. Our approach also improves model behavior in the context reasoning task, which requires stable answers under partial information. Our method delivers both higher consistency and improved accuracy. Specifically, LLaVA-1.5M shows a dramatic gain in O_{all} from 41.47 to 55.26 (adaptive), while LLaVA-Next reaches the highest score of 59.45. Interestingly, even though Qwen2-VL starts from a stronger baseline, our method boosts its O_{all} to 58.80 and consistency to 99.53. These results suggest that test-time consistency not only stabilizes outputs but also improves factual grounding under ambiguity.

4.2 Comparison with Fine-Tuning

To contextualize the effectiveness of our approach, we compare it against the fully fine-tuned model from MM-R^3 (Chou et al., 2024), which retrains LLaVA-1.5M through large-scale supervised training using task-specific data from the curated MM-

Table 2: **Comparison of our approach with supervised fine-tuned model on LLaVa 1.5M model.**

Models	Acc	S_{GT}	Con	S_C	O_{all}
Question Rephrasing					
LLaVa 1.5M	36.18	62.96	48.55	64.1	52.73
+ Finetuning (Chou et al., 2024)	42.55	69.03	63.79	75.83	62.02
+ Adapt. T	39.58	65.10	79.11	86.10	**64.08**
Image Restyling					
LLaVa 1.5M	9.61	34.85	18.96	56.91	28.03
+ Finetuning (Chou et al., 2024)	25.45	50.67	50.94	66.06	**46.11**
+ Adapt. T	17.94	40.15	33.90	64.46	36.52
Context Reasoning					
LLaVa 1.5M	16.11	42.69	65.64	75.08	41.47
+ Finetuning (Chou et al., 2024)	63.93	76.62	75.00	83.91	**74.58**
+ Adapt. T	31.04	55.14	72.11	81.9	55.26

Table 3: **Ablation Studies on contribution of different loss functions we use in our approach**

	$\mathcal{L}_{CE}$	$\mathcal{L}_{PL}$	Acc	S_{GT}	Con	S_C	O_{all}
Question Rephrasing			61.44	69.71	52.29	66.86	62.43
	✓		59.48	71.70	52.94	66.36	62.48
		✓	66.67	76.21	85.62	88.90	78.56
	✓	✓	66.01	77.18	90.20	93.10	**80.39**
Image Restyling			14.16	38.36	54.33	70.77	36.99
	✓		16.12	39.86	61.86	74.10	39.65
		✓	17.93	40.73	83.97	89.86	43.86
	✓	✓	19.25	40.35	84.94	90.40	**44.48**
Context Reasoning			32.68	27.23	31.37	55.81	35.51
	✓		28.10	52.19	55.56	71.80	49.25
		✓	32.77	56.13	97.14	97.1	60.99
	✓	✓	33.33	55.76	98.69	99.26	**61.44**

Table 4: **Hyper-parameter search on LLaVa-Next.**

	α	β	Acc	S_{GT}	Con	S_C	O_{all}
Question Rephrasing	0.1	1	66.01	76.38	86.27	89.81	78.73
	0.5	1	66.01	77.18	90.20	93.10	**80.39**
	1	1	64.71	75.63	83.00	87.82	77.04
	1	0.5	64.71	75.68	83.01	87.88	77.07
	1	0.1	65.36	75.61	79.08	84.83	75.79
Image Restyling	0.1	1	17.91	40.23	86.22	91.01	43.78
	0.5	1	19.25	40.35	84.94	90.4	**44.48**
	1	1	17.93	40.28	85.24	90.13	43.70
	1	0.5	17.93	40.28	85.26	90.47	43.73
	1	0.1	17.84	40.60	82.37	87.19	43.46
Context Reasoning	0.1	1	32.68	55.32	97.39	98.11	60.68
	0.5	1	33.33	55.76	98.69	99.26	**61.44**
	1	1	33.33	55.69	97.39	98.30	61.19
	1	0.1	32.68	55.17	96.08	97.25	60.4
	1	0.5	32.68	55.21	94.77	96.30	60.20

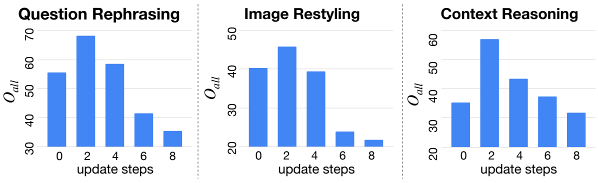

Figure 3: We show effect of different number of update steps for each task.

R^3 training set. In contrast, our method adapts the model using only a single test point and two test-time gradient steps, without access to labeled data, training code, or model internals.

Table 2 presents the results on three MM-R^3 tasks. Despite being significantly lighter in terms of computational cost and supervision, our method achieves competitive—and in some cases superior—performance compared to full fine-tuning. Specifically, on the *Question Rephrasing* task, it achieves an O_{all} score of 64.08, outperforming the fine-tuned model (62.02) by a notable margin.

On *Context Reasoning*, although full fine-tuning achieves the highest score (74.58), our method still improves substantially over the base model (55.26 vs. 41.47), again without any retraining. For *Image Restyling*, our method narrows the gap considerably (36.52 vs. 46.11), demonstrating strong robustness to visual perturbations even without additional training data. Notably, our method underperforms on these tasks in overall score because full fine-tuning jointly learns the novel task (unsupported by the base VLM) through curated training dataset and improves consistency. It can be seen that the performance of our approach on consistency (i.e Con) is nearly equivalent to that of full-finetuning, while on accuracy the improvement drops. This is not surprising as (Chou et al., 2024) learns from voluminous training data, which our model is not designed to do being a test-time approach.

4.3 Ablation Study

All experiments in the ablation studies are performed on the LLaVA-Next model, unless specified otherwise.

4.3.1 Contribution of each component in our test-time consistency framework

To understand the contribution of each component in our test-time consistency framework, we perform an ablation study by selectively enabling the Cross-Entropy Agreement Loss ($\mathcal{L}_{\mathrm{CE}}$) and the Pseudo-Label Consistency Loss ($\mathcal{L}_{\mathrm{PL}}$). Table 3 reports results across all three MM-R^3 tasks.

Complementary Benefits. We observe that both losses independently contribute to improving consistency and overall performance. Applying only $\mathcal{L}_{\mathrm{CE}}$ improves consistency over the base model in all tasks, while $\mathcal{L}_{\mathrm{PL}}$ alone often yields stronger gains in Acc.

Best Results with Combined Loss. The full method—using both $\mathcal{L}_{\text{CE}}$ and $\mathcal{L}_{\text{PL}}$—achieves the highest overall performance across all tasks. For instance, in the question rephrasing task, the combination yields $O_{all} = 80.39$ and consistency of 90.20, outperforming both individual losses. Similar trends are observed in image restyling and context reasoning, where the joint objective achieves the best O_{all} scores of 44.48 and 61.44, respectively. These results show the complementary roles of two losses: $\mathcal{L}_{\text{CE}}$ promotes token-level alignment of outputs across input perturbations, while $\mathcal{L}_{\text{PL}}$ anchors model predictions to a consensus output.

4.3.2 Ablation on number of updated steps.

Figure 3 shows the impact of varying the number of gradient update steps (T) in our **Test-time (constant T)** variant, where a fixed number of updates is applied to all test inputs. We observe that performance improves initially but degrades beyond a certain point, revealing a trade-off between effective adaptation and overfitting. Across all three tasks, setting $T = 2$ yields the best score (O_{all}).

The performance drop beyond $T = 2$ is most pronounced in the *Question Rephrasing* and *Context Reasoning* tasks, likely due to over-adaptation and overfitting on linguistic variations or ambiguous inputs. In contrast, the *Image Restyling* task is relatively robust to the number of updates, suggesting greater stability under visual perturbations.

This ablation is specific to the **Test-time (constant T)** setup. Our alternative variant, **Test-time (adapt. T)**, automatically selects the optimal number of updates per instance using the adaptive step selection mechanism described in Section 3.5. As such, it does not require manual tuning or per-task sensitivity analysis. Together, these two variants allow us to assess the trade-offs between simplicity and input-specific adaptability.

4.3.3 Ablation on Loss Weighting Coefficients

We ablate the loss weighting coefficients α and β in our total loss $\mathcal{L}_{\text{total}} = \alpha \cdot \mathcal{L}_{\text{CE}} + \beta \cdot \mathcal{L}_{\text{PL}}$, using LLaVA-Next across the MM-R^3 tasks. Results in Table 4 show that our method is robust to a range of settings, but performance is highest when both objectives are appropriately balanced.

The best results are obtained with $\alpha = 0.5$ and $\beta = 1$, yielding top O_{all} scores across all tasks: 80.39 (Rephrasing), 44.48 (Restyling), and 61.44 (Reasoning). Performance drops slightly when either loss dominates—for example, using $\beta = 0.1$ reduces consistency and overall score.

Table 5: **Results on original OKVQA dataset task.**

Acc	LLaVA 1.5M	LLaVA-Next	Qwen2-VL
Original	55.09	54.69	54.13
+ Constant T	53.98	56.10	58.61

4.3.4 Preservation of Base Model Capabilities

To ensure that our test-time consistency framework does not degrade the model's original capabilities, we evaluate performance on the unperturbed OKVQA dataset (Marino et al., 2019) before and after adaptation. For this experiment, we generate three semantically equivalent rephrasings of each original question using GPT-4V. These rephrasings are used during adaptation, while the final evaluation is performed on the original (unmodified) question from OKVQA dataset.

Results are shown in Table 5. Both LLaVA-Next and Qwen2-VL improve in accuracy on original unperturbed input after test-time adaptation—rising from 54.69 to 56.10 and from 54.13 to 58.61, respectively. This indicates that our method not only preserves but can even enhance model performance on standard benchmarks. LLaVA 1.5M shows a minor drop (55.09 → 53.98), suggesting slightly higher sensitivity in smaller models. Overall, these results show that our approach does not degrade on the original task distribution, and instead enables consistency improvements.

4.3.5 Ablation on Decoding Temperature.

The ablation studies on different decoding temperatures, $\tau = 0, 0.5, 1$ are shown in Appendix A.1.

4.3.6 Qualitative Results.

We show qualitative results for three tasks in the Appendix (see Appendix A.4 for more details).

5 Conclusion.

We present a simple yet effective *test-time consistency* framework for vision–language models that requires no access to curated training data, model internals, or supervised fine-tuning. By leveraging semantically equivalent variants of each input and enforcing agreement through two lightweight losses, our method seamlessly adapts VLMs at inference-time using inherent information in single test-input. Experiments on the MM-R^3 benchmark show that our approach significantly improves consistency while preserving or enhancing accuracy. We advocate for consistency as a core evaluation criterion for building reliable, real-world VLM systems in future work.

Limitations

Our analysis is limited by the scope of the MM-R^3 dataset and its predefined perturbations, which may not fully capture the diversity of real-world consistency challenges. While our method improves consistency without access to training data or model internals, it requires multiple forward and backward passes per test input, which increases inference-time latency. However, it remains significantly more efficient and scalable overall compared to full fine-tuning, as it avoids large-scale training and need for supervision. Additionally, since adaptation is performed locally on a single test point, it may not correct broader model deficiencies or systematic biases. Finally, because our approach updates model parameters during inference, it may not be suitable for deployment in strictly frozen or closed-source model environments.

References

Eman Ali, Sathira Silva, and Muhammad Haris Khan. 2025. Dpa: Dual prototypes alignment for unsupervised adaptation of vision-language models. In *WACV*.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. 2015. VQA: Visual Question Answering. In *ICCV*.

David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. 2019. Mixmatch: A holistic approach to semi-supervised learning. *NeurIPS*.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada. Association for Computational Linguistics.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. 2015. Microsoft coco captions: Data collection and evaluation server. *ArXiv*.

Shih-Han Chou, Shivam Chandhok, James J Little, and Leonid Sigal. 2024. Mm-r^3: On (in-) consistency of multi-modal large language models (mllms). *ArXiv*.

Shih-Han Chou, Zicong Fan, James J Little, and Leonid Sigal. 2022. Semi-supervised grounding alignment for multi-modal feature learning. In *CRV*.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. 2024. Detecting hallucinations in large language models using semantic entropy. *Nature*.

Yves Grandvalet and Yoshua Bengio. 2004. Semi-supervised learning by entropy minimization. *NeurIPS*.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *ArXiv*.

Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. 2017. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *CVPR*.

Adilbek Karmanov, Dayan Guan, Shijian Lu, Abdulmotaleb El Saddik, and Eric Xing. 2024. Efficient test-time adaptation of vision-language models. In *CVPR*.

Yuto Kojima, Jiarui Xu, Xueyan Zou, and Xiaolong Wang. 2025. Lora-ttt: Low-rank test-time training for vision-language models. *ArXiv*.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *ArXiv*.

Dong-Hyun Lee and 1 others. 2013. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *ECCV*.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024a. Improved baselines with visual instruction tuning. In *CVPR*.

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

Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. 2019. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *CVPR*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL*.

Wenqi Shi, Ran Xu, Yuchen Zhuang, Yue Yu, Haotian Sun, Hang Wu, Carl Yang, and May Dongmei Wang. 2024. MedAdapter: Efficient test-time adaptation of large language models towards medical reasoning. In *EMNLP*.

Manli Shu, Weili Nie, De-An Huang, Zhiding Yu, Tom Goldstein, Anima Anandkumar, and Chaowei Xiao. 2022. Test-time prompt tuning for zero-shot generalization in vision-language models. *NeurIPS*.

Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. 2020. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *NeurIPS*.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. 2021. Tent: Fully test-time adaptation by entropy minimization. In *ICLR*.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, and 1 others. 2024. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *ArXiv*.

Yang Yang, Chaoyue Wang, Risheng Liu, Lin Zhang, Xiaojie Guo, and Dacheng Tao. 2022. Self-augmented unpaired image dehazing via density and depth decomposition. In *CVPR*.

David Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *ACL*.

Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. From recognition to cognition: Visual commonsense reasoning. In *CVPR*.

Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. 2021. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *NeurIPS*.

Marvin Zhang, Sergey Levine, and Chelsea Finn. 2022. Memo: Test time robustness via adaptation and augmentation. *NeurIPS*.

A Appendix

A.1 Ablation on Decoding Temperature

Table 6: **Different temperature on LLaVa-Next.**

Temp	Acc	$\mathbf{S_{GT}}$	Con	$\mathbf{S_C}$	O_{all}
Question Rephrasing					
LLaVa-NEXT, $\tau = 0$	42.89	64.89	49.18	65.69	55.61
+ Constant T	44.48	68.74	83.39	88.47	**68.25**
LLaVa-NEXT, $\tau = 0.5$	42.06	65.29	52.02	66.52	56.33
+ Constant T	44.48	68.74	83.39	88.47	**68.25**
LLaVa-NEXT, $\tau = 1$	42.06	65.29	52.02	66.52	56.33
+ Constant T	44.48	68.74	83.39	88.47	**68.25**
Image Restyling					
LLaVa-NEXT, $\tau = 0$	17.57	41.47	55.34	71.36	40.27
+ Constant T	18.99	42.49	88.25	91.25	**45.80**
LLaVa-NEXT, $\tau = 0.5$	17.57	41.47	55.34	71.36	40.27
+ Constant T	17.64	40.64	82.80	76.64	**42.68**
LLaVa-NEXT, $\tau = 1$	17.57	41.47	55.34	71.36	40.27
+ Constant T	17.64	40.64	82.80	76.64	**42.68**
Context Reasoning					
LLaVa-NEXT, $\tau = 0$	30.24	27.43	32.11	58.44	35.23
+ Constant T	32.50	50.84	89.91	90.16	**56.97**
LLaVa-NEXT, $\tau = 0.5$	30.07	51.99	52.09	66.68	48.53
+ Constant T	32.31	53.84	93.4	95.31	**59.15**
LLaVa-NEXT, $\tau = 1$	30.07	51.99	52.09	66.68	48.53
+ Constant T	32.31	53.84	93.4	95.31	**59.15**

We conduct an ablation to assess the impact of decoding temperature τ on our test-time consistency framework using LLaVA-NEXT across three perturbation types: *Question Rephrasing*, *Image Restyling*, and *Context Reasoning* (Table 6).

Across all perturbations, our method improves consistency and overall robustness regardless of the temperature setting. Notably:

- **Question Rephrasing:** Our test-time strategy consistently boosts performance to a peak $O_{all} =$ **68.25** for all values of τ, indicating stable performance across decoding scales and strong resilience to linguistic variations.
- **Image Restyling:** While baseline performance is lower due to visual perturbations, our method still yields significant improvements. The best result is observed at $\tau = 0$, where O_{all} improves from 40.27 to **45.80**, a gain of 5.5 points.
- **Context Reasoning:** This task benefits most from our consistency framework. The best performance, $O_{all} =$ **59.15**, is achieved at both $\tau = 0.5$ and $\tau = 1$, indicating that our method improves reasoning-heavy tasks.

These results demonstrate that our approach is robust to temperature variation and consistently enhances consistency and semantic alignment across all perturbation categories.

A.2 Evaluation Metrics

To systematically assess the performance of VLMs, we use four distinct evaluation metrics, on similar lines as previous work (Chou et al., 2024), each capturing different aspects of model performance.

Accuracy (Acc). To evaluate accuracy we assess the responses from VLMs based on an fuzzy string matching with the ground truth annotations, accounting for minor lexical variations. A similarity threshold of 85 is used to determine a match. The accuracy score is then calculated as the average of correct responses across the benchmark test-set.

Similarity with GT ($\mathbf{S_{GT}}$). Given the limitations of exact match criteria—which may penalize semantically correct responses for minor lexical differences—we employ a semantic similarity metric to better evaluate alignment between model outputs and ground truth. For example, terms like *person*, *man*, and *woman* are semantically related but would be treated as mismatches under strict accuracy metrics. To address this, we use BERT-based Sentence Similarity (Reimers and Gurevych, 2019), which leverages contextual language model encodings to assess the semantic alignment between predictions and reference answers. This metric rewards semantic correctness over surface-form similarity. Final scores are computed as the average similarity across the dataset.

Consistency Accuracy (Con). This metric quantifies the proportion of responses that exhibit a predefined level of semantic consistency. We compute pairwise similarity scores between outputs using the same semantic similarity metric as in S_{GT}, and consider a pair consistent if its similarity exceeds a threshold of 0.7—motivated by observations from the Semantic Textual Similarity benchmark (Cer et al., 2017). A response is deemed consistent if it meets this threshold with its paired counterpart.

The final score is calculated as the average proportion of consistent pairs across the dataset, providing an aggregate measure of the model's semantic stability across perturbed inputs.

Consistency Similarity ($\mathbf{S_C}$). Similar to the Consistency Accuracy metric, this measure computes pairwise semantic similarity scores between responses to assess consistency. However, instead of applying a threshold, we take the average of these similarity scores across the dataset. This provides a more *continuous* assessment of the model's coherence, capturing fine-grained variations in semantic consistency across perturbed inputs.

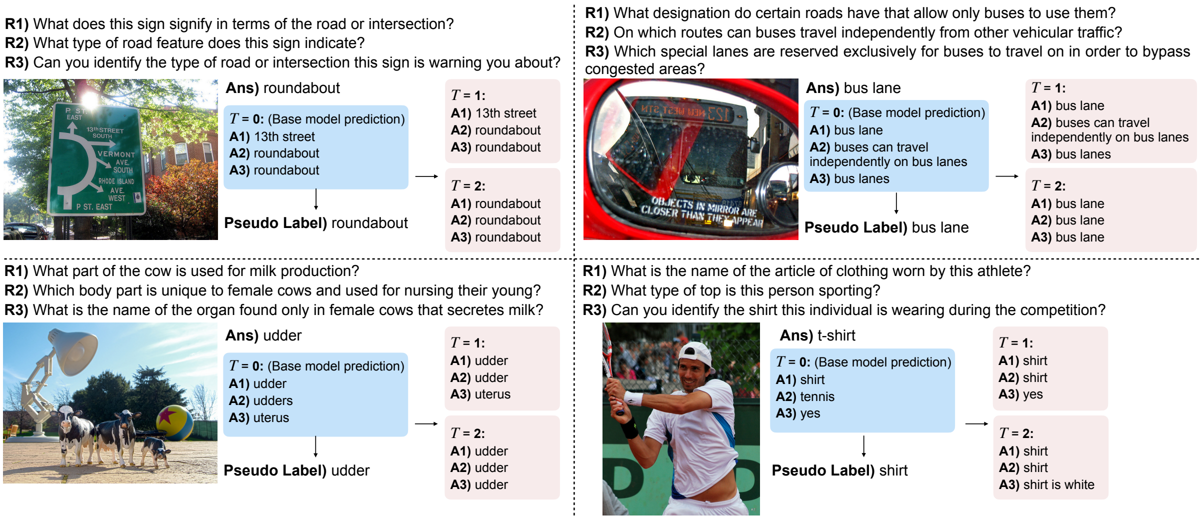
R1) What does this sign signify in terms of the road or intersection?
R2) What type of road feature does this sign indicate?
R3) Can you identify the type of road or intersection this sign is warning you about?
Ans) roundabout
T = 0: (Base model prediction)
A1) 13th street
A2) roundabout
A3) roundabout
Pseudo Label) roundabout
T = 1:
A1) 13th street
A2) roundabout
A3) roundabout
T = 2:
A1) roundabout
A2) roundabout
A3) roundabout
R1) What designation do certain roads have that allow only buses to use them?
R2) On which routes can buses travel independently from other vehicular traffic?
R3) Which special lanes are reserved exclusively for buses to travel on in order to bypass congested areas?
Ans) bus lane
T = 0: (Base model prediction)
A1) bus lane
A2) buses can travel independently on bus lanes
A3) bus lanes
Pseudo Label) bus lane
T = 1:
A1) bus lane
A2) buses can travel independently on bus lanes
A3) bus lanes
T = 2:
A1) bus lane
A2) bus lane
A3) bus lane
R1) What part of the cow is used for milk production?
R2) Which body part is unique to female cows and used for nursing their young?
R3) What is the name of the organ found only in female cows that secretes milk?
Ans) udder
T = 0: (Base model prediction)
A1) udder
A2) udders
A3) uterus
Pseudo Label) udder
T = 1:
A1) udder
A2) udder
A3) uterus
T = 2:
A1) udder
A2) udder
A3) udder
R1) What is the name of the article of clothing worn by this athlete?
R2) What type of top is this person sporting?
R3) Can you identify the shirt this individual is wearing during the competition?
Ans) t-shirt
T = 0: (Base model prediction)
A1) shirt
A2) tennis
A3) yes
Pseudo Label) shirt
T = 1:
A1) shirt
A2) shirt
A3) yes
T = 2:
A1) shirt
A2) shirt
A3) shirt is white

Figure 4: Qualitative results on the question rephrasing task.

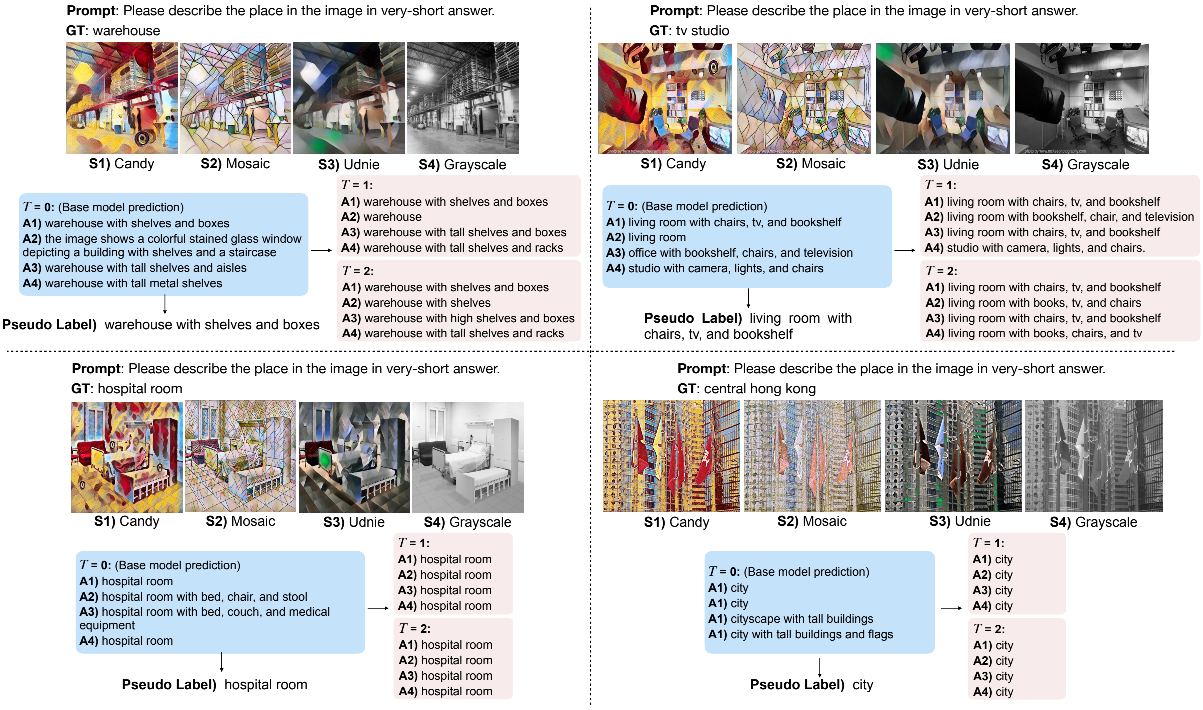
Prompt: Please describe the place in the image in very-short answer.
GT: warehouse
S1) Candy
S2) Mosaic
S3) Udnie
S4) Grayscale
T = 0: (Base model prediction)
A1) warehouse with shelves and boxes
A2) the image shows a colorful stained glass window depicting a building with shelves and a staircase
A3) warehouse with tall shelves and aisles
A4) warehouse with tall metal shelves
Pseudo Label) warehouse with shelves and boxes
T = 1:
A1) warehouse with shelves and boxes
A2) warehouse
A3) warehouse with tall shelves and boxes
A4) warehouse with tall shelves and racks
T = 2:
A1) warehouse with shelves and boxes
A2) warehouse with shelves
A3) warehouse with high shelves and boxes
A4) warehouse with tall shelves and racks
Prompt: Please describe the place in the image in very-short answer.
GT: tv studio
S1) Candy
S2) Mosaic
S3) Udnie
S4) Grayscale
T = 0: (Base model prediction)
A1) living room with chairs, tv, and bookshelf
A2) living room
A3) office with bookshelf, chairs, and television
A4) studio with camera, lights, and chairs
Pseudo Label) living room with chairs, tv, and bookshelf
T = 1:
A1) living room with chairs, tv, and bookshelf
A2) living room with bookshelf, chair, and television
A3) living room with chairs, tv, and bookshelf
A4) studio with camera, lights, and chairs.
T = 2:
A1) living room with chairs, tv, and bookshelf
A2) living room with books, tv, and chairs
A3) living room with chairs, tv, and bookshelf
A4) living room with books, chairs, and tv
Prompt: Please describe the place in the image in very-short answer.
GT: hospital room
S1) Candy
S2) Mosaic
S3) Udnie
S4) Grayscale
T = 0: (Base model prediction)
A1) hospital room
A2) hospital room with bed, chair, and stool
A3) hospital room with bed, couch, and medical equipment
A4) hospital room
Pseudo Label) hospital room
T = 1:
A1) hospital room
A2) hospital room
A3) hospital room
A4) hospital room
T = 2:
A1) hospital room
A2) hospital room
A3) hospital room
A4) hospital room
Prompt: Please describe the place in the image in very-short answer.
GT: central hong kong
S1) Candy
S2) Mosaic
S3) Udnie
S4) Grayscale
T = 0: (Base model prediction)
A1) city
A1) city
A1) cityscape with tall buildings
A1) city with tall buildings and flags
Pseudo Label) city
T = 1:
A1) city
A2) city
A3) city
A4) city
T = 2:
A1) city
A2) city
A3) city
A4) city

Figure 5: Qualitative results on the image restyling task.

Overall Performance (O_{all}). We report overall model performance using the harmonic mean (H_{mean}) of correctness and consistency scores. Specifically, we first compute the average of $\mathbf{Acc}$ and $\mathbf{S_{GT}}$ to assess correctness, and the average of $\mathbf{Con}$ and $\mathbf{S_C}$ to assess consistency. These two averages are then combined using the harmonic mean:

$$H_{mean}(mean(\mathbf{Acc}, \mathbf{S_{GT}}), mean(\mathbf{Con}, \mathbf{S_C})). \quad (12)$$

We use the harmonic mean to balance correctness and consistency, as it penalizes models that perform well on only one aspect, thereby encouraging robust performance across both dimensions.

A.3 Implementation Details.

We use the pre-trained VLMs as the base models and only fine-tune the language modelling head (LM-head) layer. We set updated steps $T = 2$ for the test-time experiments and maximum updated steps to $T = 4$ for the adaptive test-time experiments. The learning rate is set to $5e^{-4}$. All experiments are conducted on NVIDIA A40 with batch size 1 on all three models.

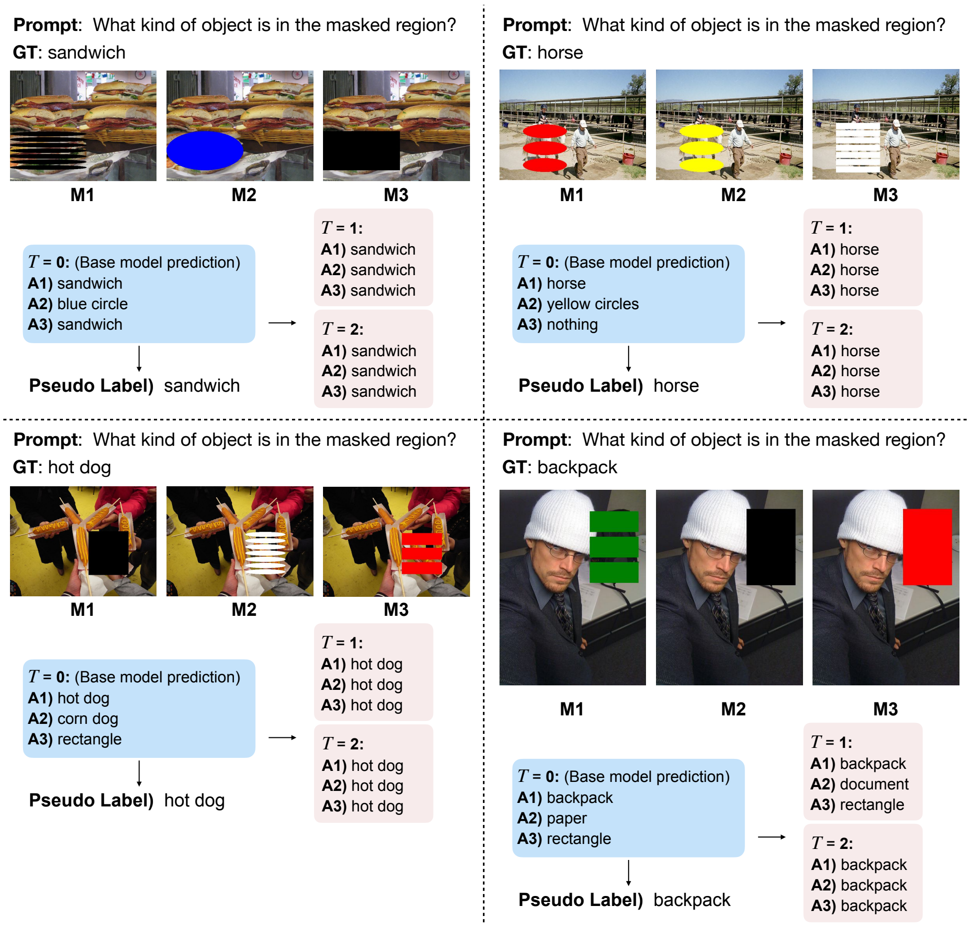

Figure 6: Qualitative results on the context reasoning task.

A.4 Qualitative Results

We show qualitative results for the question rephrasing task in Figure 4, image restyling in Figure 5, and context reasoning in Figure 6. Across all three tasks, even when the base model predictions are inconsistent, our method is able to further improve consistency and thus overall score (as also supported by quantitative results in Main manuscript).