Visual Instruction Bottleneck Tuning

Changdae Oh **Jiatong Li** **Shawn Im** **Sharon Li**
Department of Computer Sciences, University of Wisconsin–Madison
{changdae,sharonli}@cs.wisc.edu

Abstract

Despite widespread adoption, multimodal large language models (MLLMs) suffer performance degradation when encountering unfamiliar queries under distribution shifts. Existing methods to improve MLLM generalization typically require either more instruction data or larger advanced model architectures, both of which incur non-trivial human labor or computational costs. In this work, we take an alternative approach to enhance the generalization and robustness of MLLMs under distribution shifts, from a representation learning perspective. Inspired by *information bottleneck (IB) principle*, we derive a variational lower bound of the IB for MLLMs and devise a practical implementation, ***Visual Instruction Bottleneck Tuning*** (`Vittle`). We then provide a theoretical justification of `Vittle` by revealing its connection to an information-theoretic robustness metric of MLLM. Empirical validation of multiple MLLMs on open-ended and closed-form question answering and object hallucination detection tasks over 45 datasets, including 30 shift scenarios, demonstrates that `Vittle` consistently improves the MLLM's robustness under shifts by pursuing the learning of a minimal sufficient representation.
Code: https://github.com/deeplearning-wisc/vittle

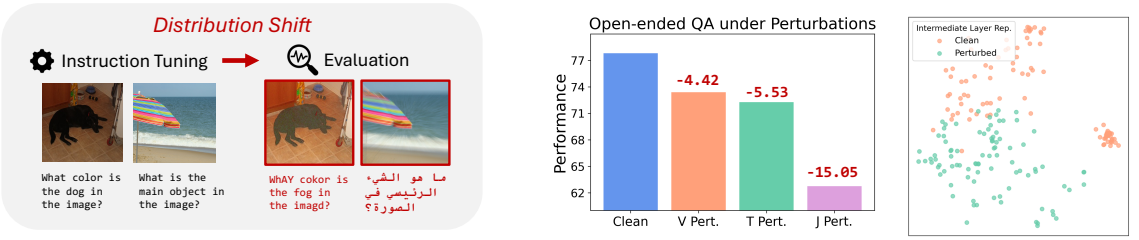
1 Introduction

In intensive races on the track of frontier-level AI models, we have observed unprecedented achievements through the form of a general-purpose chat assistant known as multimodal large language models (MLLMs) [1, 2, 3, 4] that combine a visual encoder with a large language model. Their universal yet flexible question-answering interface enables MLLMs to easily permeate our lives from general problem-solving [5, 6] to practical applications [7, 8, 9, 10]. While these models may achieve human-like or even surpass human-level performance on certain tasks, a critical gap remains in their robustness—particularly in handling input variations that humans process effortlessly.

Human intelligence thrives on the ability to distill a large amount of sensory and cognitive inputs into concise abstract representations, a process akin to *conceptual compression* [11, 12]. By prioritizing sparse salient features while discarding redundancy, humans can shape a robust prototypical representation of complex data instances that captures a proper level of **invariance to low-level superficial features** for generalization, yet maintains **sensitivity to high-level abstract features** for discrimination [13, 14, 15]. Unfortunately, there are consistent reports implying that the current MLLMs still lag far behind this desired trade-off between invariance and sensitivity [16, 17, 18, 19].

Specifically, MLLMs fail to produce relevant responses under query distribution shifts. That is, they are vulnerable to processing subtly perturbed samples and long-tailed samples [19]. This limitation partially stems from the difficulty of acquiring diverse high-quality multimodal instruction data at scale. When trained using standard maximum likelihood estimation on this relatively limited amount of instruction data, MLLM tends to fit to data-specific patterns and results in a brittle solution [20, 21, 22]. To enhance generalization, existing efforts typically fall into two categories (1) *data-centric* approaches, which collect more instruction data [23, 24, 25] and processes input in a finer granularity [26, 27], and (2) *model-centric* approaches, which scale up the underlying model

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



(a) MLLM encounters distribution shifts (b) Distribution shifts result in MLLM performance drops

Figure 1: **Illustration of distribution shifts for an MLLM (a) and performance degeneration and embedding shifts of the MLLM (b).** An MLLM (LLaVA-v1.5-7B) receives arbitrary queries that might be visually and/or textually perturbed by unexpected noise. These distribution shifts result in performance drops, as shown in the middle bar plot. A visualization of intermediate layer representations of the MLLM on LLaVA-Bench-COCO and its variants indicates that MLLM fails to learn a proper level of invariance to generalize multimodal queries in the representation space.

using more expressive or specialized backbones [28, 29, 30, 31]. However, both data scaling and model scaling are resource-intensive—requiring significant annotation or computational cost.

In this work, we propose a new approach from a *representation-centric* view to improve the robustness of MLLMs under distribution shifts. Rather than scaling data or model, we introduce a lightweight, theoretically grounded module that enhances the internal representations of MLLMs via the information bottleneck (IB) principle. While the IB framework has been explored in small-scale or classification settings [32, 33, 34, 35, 36], integrating it to autoregressive multimodal instruction tuning poses unique challenges due to the complexity of modeling mutual information across high-dimensional, sequential, and heterogeneous modalities. We overcome these barriers by formulating a novel variational lower bound of the IB objective specifically tailored to the multimodal and sequential nature of MLLMs. We further instantiate this formulation as a modular and scalable implementation—***Visual Instruction Bottleneck Tuning*** (`Vittle`), which inserts one simple bottleneck layer within the LLM backbone. `Vittle` pursues *minimal sufficient representations* [37] that try to preserve only response-relevant information while discarding non-essential residual features. To our knowledge, this is the first work to investigate the IB framework for end-to-end instruction tuning of multimodal LLMs, offering a model-agnostic pathway toward building more robust AI systems.

We conduct an extensive evaluation of `Vittle` across a wide spectrum of multimodal benchmarks to assess its robustness and generalization under distribution shift. Our experiments span 30 distribution shifts covering diverse forms of perturbation (in both vision and language) and long-tail distributions. Through these evaluations, we demonstrate that `Vittle` consistently improves robustness over standard instruction tuning baselines, without sacrificing performance on standard benchmarks and canonical tasks. Notably, we find that the bottlenecked representations induced by `Vittle` lead to enhanced invariance in the latent space, aligning semantically similar inputs more closely—even under input shifts—while reducing overfitting to modality-specific artifacts. We also show that `Vittle` is compatible with different MLLMs, offering robustness gains while maintaining similar inference-time cost. These results underscore the practical benefit and theoretical promise of information-regularized representation learning for robust multimodal instruction tuning.

Contributions: (1) We propose a new representation-centric framework for improving the robustness of MLLMs under distribution shifts, grounded in the information bottleneck principle. (2) We explore the IB-based end-to-end learning objective of an MLLM for the first time by inducing a new variational lower bound of IB for MLLM and devising a practical instantiation, `Vittle`, supported by theoretical analysis. (3) Through experiments on 30 diverse types of distribution shifts, we thoroughly validate the robustness of MLLMs on open-ended/closed-form QA and object hallucination detection tasks and show advantages of compressive representation induced by pursuing the IB principle.

2 Background, Related Work, and Motivation

Multimodal large language models (MLLMs). Recent advances in MLLMs integrate a pre-trained language model with a vision encoder through *visual instruction tuning* [38, 39]. To be specific, let $X = (X_v, X_t)$ denote a multimodal input query consisting of visual and textual input, e.g., an image and a corresponding instruction or a question given that image, and Y denote a desired

response given the input query. An MLLM f_θ with parameter θ is trained to produce the desired response given an input query with a conditional autoregressive language modeling objective, i.e., $\arg\min_\theta \mathbb{E}_{X,Y}[\sum_{m=1}^{M} \log f_\theta(Y_m|X_v, X_t, Y_{<m})]$ for a sequence of M-length responses, where the visual input X_v go through a visual encoder and projector modules to be converted as a sequence of tokens that have the same dimension as text embeddings and can be processed by an LLM backbone[1]. After being trained, these models process a wide array of multimodal instructions to solve arbitrary visual question answering tasks [40].

Robustness problem in MLLMs. Despite their impressive performance on standard benchmarks and their growing deployment in real-world applications [8, 41, 42], MLLMs remain vulnerable to input perturbations [43, 44, 45]. For example, MLLMs undergo a systematic performance drop [19] when they encounter samples of superficial perturbations (e.g., varying brightness of image and typo in text) illustrated in Figure 1 (a). As shown in the bar plot of Figure 1 (b), LLaVA-v1.5-7B model undergoes severe performance degradation on LLaVA-Bench-COCO (LB-COCO; [38]) under the perturbations from visual input, textual input, and their joint (V, T, and J Pert.).

We posit that these vulnerabilities arise from the way MLLMs structure their internal representation space. In particular, inputs affected by perturbations are often embedded far from their intact (clean) counterparts, reflecting a distribution shift in the representation space that leads to poor generalization from an information-theoretic perspective [19]. The right side of Figure 1 (b) illustrates this phenomenon: using LLaVA-v1.5, we visualize representations of LB-COCO alongside its challenging variant, where the image and text inputs are perturbed. In this setting, semantically equivalent examples are mapped to distinct and distant regions in the latent space, *suggesting a lack of invariance to superficial input variations, which is crucial for robustness to distribution shifts*.

Motivated by this, our work aims to enhance the robustness of MLLMs by explicitly regularizing their internal representations, encouraging them to retain task-relevant information while discarding input-specific noise—thereby finding a good balance between invariance to low-level superficial features and sensitivity to high-level abstract features for better generalization.

Information bottleneck principle. The information bottleneck framework provides a principled approach to measure the quality of representations that are maximally predictive of a target variable while compressing redundant information from an input variable [46, 47]. Numerous works have explored the use of IB training objective [32], across computer vision [48, 49], natural language processing [35, 36], graph learning [34, 50], and time-series modeling [51]. These efforts are supported by theoretical insights suggesting that optimizing for the IB objective can reduce generalization error [33, 52]. However, most prior work focused on classification settings [35, 36] and/or relatively small-scale models [32, 53, 35]. Although a recent study explored IB for MLLMs [54], the authors adopted IB training on a lightweight projector module while keeping the LLM backbone frozen. In contrast, *our work is the first to investigate the IB framework for end-to-end training of large-scale autoregressive multimodal language models*. Beyond shallow adaptations, we directly modify the internal structure of the LLM to promote IB-consistent behavior throughout the training process. We focus specifically on instruction tuning for MLLMs—which have become increasingly central to modern AI ecosystems but remain largely unexplored from the perspective of IB-based learning.

3 Method

3.1 Preliminary: Information Bottleneck As a Learning Objective

Let X be a multimodal input query (e.g., image-text pair), Y the desired output, and $Z = f(X)$ an intermediate representation extracted by the MLLM encoder $f(\cdot)$. The Information Bottleneck principle aims to learn representations that are *maximally informative about the output* Y while being *minimally informative about the input* X. Formally, this is expressed as the optimization objective:

$$\max_f \text{IB}_f(X, Y) := \underbrace{I(Z, Y)}_{\text{acquiring information from desired output}} - \beta \underbrace{I(Z, X)}_{\text{compressing input-specific redundant information}}, \quad (1)$$

where $I(\cdot, \cdot)$ denotes mutual information and β is the trade-off coefficient. Minimizing $I(Z, X)$ encourages removing redundant or input-specific variations, while maximizing $I(Z, Y)$ ensures that the representation retains task-relevant signals necessary to predict the desired output.

[1]For simplicity, we will omit the visual encoder and projector in our learning objective at following sections.

In other words, the IB objective promotes representations that discard non-essential features tied to the input domain, while preserving those critical for solving the task. This property is desirable for robust instruction tuning, where user queries that have the same latent goal must be mapped to consistent responses under varied conditions (e.g., visual and textual perturbations). Despite its appeal, integrating the IB objective into MLLM training is ***highly non-trivial due to the intractability of mutual information estimation and the complexity of autoregressive and multimodal architectures***.

3.2 Variational Inference for Information Bottleneck in MLLMs

Directly optimizing the IB objective is generally intractable, as it involves mutual information terms over unknown data distributions. In this work, we introduce a tractable variational bound on the IB objective, specifically tailored to the autoregressive and multimodal structure of MLLMs. We outline the key steps below and provide full derivations in the Appendix D.

We begin with the mutual information term $I(Z, X)$. Given the sequential nature of MLLMs, we decompose both the input $X = (X_v, X_t)$ and the latent representation $Z = (Z_v, Z_t)$ into **v**isual and **t**extual components. We can then derive the following upper bound for $I(Z, X)$:

$$\begin{aligned} I(Z,X) &= \mathbb{E}_{x,z}[\log \frac{p(z|x)}{p(z)}] \leq \mathbb{E}_{x,z}[\log \frac{p(z|x)}{r(z)}] = \mathbb{E}_{x_v,x_t,z_v,z_t}[\log \frac{p(z_t|x_v,x_t)p(z_v|x_v)}{r(z_v)r(z_t)}] \\ &= \mathbb{E}_{x_v,x_t}[\mathbb{E}_{z_t|x_v,x_t}[\mathbb{E}_{z_v|x_v}[\log \frac{p(z_v|x_v)}{r(z_v)}]]] + \mathbb{E}_{x_v,x_t}[\mathbb{E}_{z_v|x_v}[\mathbb{E}_{z_t|x_v,x_t}[\log \frac{p(z_t|x_v,x_t)}{r(z_t)}]]] \\ &= \mathbb{E}_{x_v}[D_{\mathrm{KL}}(p(z_v|x_v)||r(z_v))] + \mathbb{E}_{x_v,x_t}[D_{\mathrm{KL}}(p(z_t|x_v,x_t)||r(z_t))], \end{aligned} \tag{2}$$

where the first inequality holds given the non-negativity of Kullback-Leibler divergence (KLD), $D_{\mathrm{KL}}(r(z)||p(z))$, and $p(z_v|x_v, x_t) = p(z_v|x_v)$ due to causal mask in MLLM. Here, we introduce $r(z) = r(z_v, z_t) = r(z_v)r(z_t)$ as a factorizable variational approximation of the true prior $p(z)$.

Next, for the output-relevant term $I(Z, Y)$, we have the lower bound as the same as Alemi et al. [32]:

$$I(Z,Y) = \mathbb{E}_{y,z}\left[\log \frac{p(y|z)}{p(y)}\right] \geq \mathbb{E}_{x,y,z}\left[\log q(y|z)\right] - \mathbb{E}_y[\log p(y)] \geq \mathbb{E}_{x,y}\left[\mathbb{E}_{z|x}\left[\log q(y|z)\right]\right], \tag{3}$$

where we replace the true posterior $p(y|z)$ with a variational approximation $q(y|z)$ that will be parameterized by a model component (will be elucidated in Section 3.3).

Finally, combining the lower bound of $I(Z, Y)$ and the upper bound of $I(Z, X)$ yields a variational lower bound for the IB objective as follows,

$$\begin{aligned} \mathrm{IB}(X,Y) \geq\ & \mathbb{E}_{x,y}\left[\mathbb{E}_{z|x}[\log q(y|z)]\right] \\ & - \beta\left(\mathbb{E}_{x_v}\left[D_{\mathrm{KL}}(p(z_v|x_v)||r(z_v))\right] + \mathbb{E}_{x_v,x_t}\left[D_{\mathrm{KL}}(p(z_t|x_v,x_t)||r(z_t))\right]\right), \end{aligned} \tag{4}$$

In the next section, we elaborate on how we can implement this variational lower bound for MLLM instruction tuning in practice.

3.3 `Vittle`: A Practical Implementation of Visual Instruction Bottleneck Tuning

By using a Monte Carlo approximation of expectations over data, Eq. (4) can be written as follows,

$$\mathcal{L}_\beta = \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{z|x^i}[\log q(y^i|z)] - \beta\left(D_{\mathrm{KL}}(p(z_v|x_v^i)||r(z_v)) + D_{\mathrm{KL}}(p(z_t|x_v^i,x_t^i)||r(z_t))\right), \tag{5}$$

where, $x^i = (x_v^i, x_v^t)$ denotes the i-th sample query from an instruction tuning dataset. To compute this empirical estimate of the IB lower bound, we need to model the posterior distributions, $p(z_v|x_v)$ and $p(z_t|x_v, x_t)$, and prior distributions $r(z_v)$ and $r(z_t)$, of the MLLM's inner representation Z. Although in principle these distributions can take arbitrary forms, multivariate Gaussian distributions with simplified covariance matrices have been widely adopted in variational inference and probabilistic embedding literature [55, 56, 57, 32, 58, 59, 60, 61] due to their mathematical tractability and empirical effectiveness. By following this common standard, we set the posteriors and priors as Gaussian with diagonal covariance, and will elucidate how exactly they are defined below.

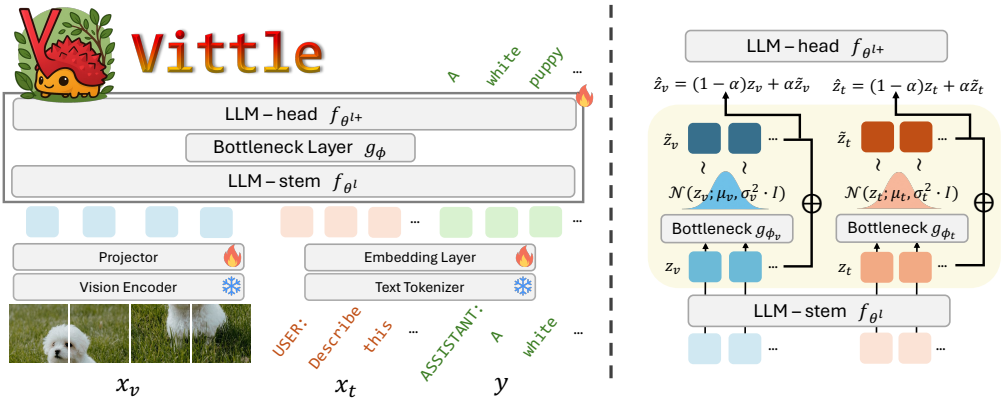


Figure 2: `Vittle` **architecture**. We insert a learnable bottleneck layer $g_\phi = \{g_{\phi_v}, g_{\phi_t}\}$ on top of l blocks of LLM backbone (i.e., LLM-stem f_{θ^l}) to estimate posterior distributions of token embeddings. After obtaining a sample per token $\{\tilde{z}_v, \tilde{z}_t\}$ from posteriors, we interpolate it with a pre-bottlenecked token representation $\{z_v, z_t\}$ and pass it through the remaining LLM blocks (i.e., LLM-head $f_{\theta^{l+}}$).

Posterior distributions. As illustrated in Figure 2, we parameterize the posteriors $p(z_v|x_v)$ and $p(z_t|x_v, x_t)$ using simple feed-forward blocks. Specifically, they are non-linear mappings, $g_{\phi_\cdot} : \mathbb{R}^d \to \mathbb{R}^{2d}$ implemented by multi-layer perceptron (MLP) for each modalities, which map each d-dimensional token embedding to the posterior Gaussian parameter vectors $\mu \in \mathbb{R}^d$ and $\sigma^2 \in \mathbb{R}^d_+$. Given an intermediate l-th layer output representation $(z_v, z_t) = f_{\theta^l}(x_v, x_t)$, we define:

$$p(z_v|x_v) := \mathcal{N}(z_v; \mu_v, \sigma_v^2 \cdot I), \quad p(z_t|x_v, x_t) := \mathcal{N}(z_t; \mu_t, \sigma_t^2 \cdot I),$$

where $[\mu_v, \sigma_v^2] = g_{\phi_v}(f_{\theta^l}(x_v))$ and $[\mu_t, \sigma_t^2] = g_{\phi_t}(f_{\theta^l}(x_v, x_t))$, with the mean and variance parameters are bipartited along the output dimensions of the MLP. These MLPs are applied position-wise in the same manner as Transformer's feed-forward layers [62], producing token-wise variational posteriors. Now, we can sample from the posterior distributions of MLLM representation by $\tilde{z}_v \sim p(z_v|x_v)$ and $\tilde{z}_t \sim p(z_t|x_v, x_t)$. Then, to strike a balance between invariance and sensitivity, we interpolate the original representation z (pre-bottleneck) with the post-bottleneck counterpart $\tilde{z}$ as $\hat{z} = (1-\alpha)z + \alpha\tilde{z}$[2]. These representations are fed into the remaining layers to compute the predictive distribution over outputs, i.e., $q(y|z) := f_{\theta^{l+}}(y|\hat{z}_v, \hat{z}_t)$. While direct sampling introduces non-differentiability, we can enable the gradient flow using the reparameterization trick [56] to sample $\tilde{z}$ via $\tilde{z} = \mu + \sigma \odot \epsilon$ with $\epsilon \sim \mathcal{N}(\mathbf{0}, I)$ where μ and σ are the outputs of the bottleneck MLP module given input x.

Prior distributions. We consider two instantiations of the prior distribution for both Z_v and Z_t: (1) a *fixed* standard[3] Gaussian $\mathcal{N}(\mathbf{0}, I)$, which is input-independent and enforces strong isotropy, and (2) a *learnable* Gaussian $\mathcal{N}(\mu_\psi, \sigma_\psi^2 \cdot I)$, where μ_ψ and σ_ψ^2 are two learnable vectors shared across samples. Each prior affects the formation of representations differently—the fixed prior imposes stronger regularization and robustness, while the learnable prior introduces additional flexibility by allowing the model to adapt to the instruction tuning distribution. We name the former `Vittle` (F) and the latter `Vittle` (L), and validate them altogether for all the evaluations in Section 4.

Overall objective and implementation. The first term of $\mathcal{L}_\beta$(Eq. (5)) can be easily computed through the standard cross-entropy, and our Gaussian instantiation of posteriors and priors allows us to derive closed-form expressions of KLD terms that can be computed from simple arithmetic between μ and σ^2 parameters (See Appendix A.2). We set $\beta = \frac{0.1}{d}$ where d is the hidden dimension of the MLLM, to normalize the KL regularization terms relative to the size of the latent dimension. The interpolation coefficient α in $\hat{z} = (1-\alpha)z + \alpha\tilde{z}$ increases progressively following a cosine schedule up to 0.5. During inference, we consistently use an averaged representation $\hat{z} = (z + \tilde{z})/2$ with a deterministic posterior representation $\tilde{z} = \mu$ rather than using a random sample. The target layer to apply the bottleneck module can differ between visual and textual tokens, but we set $l = 24$ for both modalities among 32 layers in a 7B-size LLM, i.e., top 25% layer, by default for simplicity (See Appendix B.1 for the ablation study). Figure 2 depicts the architecture overview.

[2]This interpolation yields an asymmetric posterior specification for $I(Z, X)$ and $I(Z, Y)$ (Appendix D).

[3]An alternative is to use the representation statistics (mean and variance) from a pre-instruction-tune model to host informative priors similar to `mixout` [63], which may relax the need for interpolation $(1-\alpha)z + \alpha\tilde{z}$.

3.4 Theoretical Justification

The learning objective of `Vittle` has an attractive theoretical interpretation that can support the improvement in robustness of `Vittle`. In this section, we first introduce a recently proposed information-theoretic measure of MLLM's robustness under distribution shifts, ***effective mutual information difference***, EMID [19], and show how `Vittle` can contribute to reducing EMID.

Definition 3.1 (**EMID**). *Let $P_\Theta : \mathcal{X} \to \mathcal{Y}$ be an MLLM with parameters Θ that produces an output response Y_Θ given an input instruction X. For joint distributions P_{XY} and Q_{XY}, effective mutual information difference of P_Θ over distributions P and Q is defined as below,*

$$\textit{EMID}(P_{XY}, Q_{XY}; P_\Theta) := [I(P_{XY_\Theta}) - I(P_{XY})] - [I(Q_{XY_\Theta}) - I(Q_{XY})]. \tag{6}$$

where I denotes the mutual information that measures the relevance between input query and response. A higher value of EMID indicates that MLLM P_Θ undergoes query-response relavance degeneration in the distribution Q (e.g., test-time) compared to P (e.g., train-time), so we want to achieve a lower value of it to ensure robustness. We now derive an upper bound for the EMID (See Appendix E).

Proposition 3.2 (**EMID upper bound**). *Let P_Θ be an MLLM that maps $X = \{X_v, X_t\}$ to $Z = \{Z_v, Z_t\}$, and then subsequently maps Z to Y_Θ. Given joint distributions $P_{XY} = P_X \times P_{Y|X}$ and $Q_{XY} = Q_X \times Q_{Y|X}$ (resp. P_{ZY} and Q_{ZY}), by assuming consistent conditionals over $Z_v|Z_t$, $Z_t|Z_v$, and $Y|X$ between P and Q, we have an upper bound for EMID$(P_{XY}, Q_{XY}; P_\Theta)$ as below,*

$$\hat{H}\big(D_{\text{JS}}^{\frac{1}{2}}(P_{Z_v}||Q_{Z_v}) + D_{\text{JS}}^{\frac{1}{2}}(P_{Z_t}||Q_{Z_t}) + \sqrt{\Delta_{X|Z}}\big) + |H(P_{Y_\Theta}) - H(P_Y)| + |H(Q_{Y_\Theta}) - H(Q_Y)|, \tag{7}$$

where H and $D_{\text{JS}}^{\frac{1}{2}}$ indicate the entropy and square root of Jensen-Shannon divergence (JSD), respectively, $\Delta_{X|Z} := \mathbb{E}_{z\sim P}[D_{\text{KL}}(P_{X|z}||M_{X|z})] + \mathbb{E}_{z\sim Q}[D_{\text{KL}}(Q_{X|z}||M_{X|z})]$ with a mixture distribution $M = \frac{P+Q}{2}$, and $\hat{H} := \max_{x\in\mathcal{X}}[H(Q_{Y|x}) + H(P_{Y_\Theta})]$. As we are interested in the terms related to Θ being optimized, the terms $H(P_Y)$, $H(Q_Y)$, and $\max H(Q_{Y|X})$, can be ignored from Eq. 7 which are fixed across model parameters. We can also ignore $\sqrt{\Delta_{X|Z}}$ term because it cannot directly affect Y_Θ given the conditional independence $X \perp Y_\Theta | Z$. That is, the upper bound of EMID is boiled down to the multiplication and summation between the output entropy and the representational divergence.

> **Implication.** `Vittle` maximizes the variational lower bound of IB, which consists of (1) minimizing a standard negative log-likelihood term representing an expected risk, and (2) minimizing KLD terms to enforce posterior distributions close to prior distributions. By (1), MLLM P_Θ seeks a solution Θ that minimizes the expected risk and reduces its output entropy $H(P_{Y_\Theta})$ and probably $H(Q_{Y_\Theta})$ as well [64, 65, 66]. By (2), it may reduce JSD between representation distributions P_Z and Q_Z by promoting all posterior samples to be laid near the pre-defined priors [67, 68, 69]. In summary, reduced entropy and JSD terms induce a lower EMID, which means that `Vittle` may achieve better robustness to distribution shifts than the standard training method that neglects the divergence in representation space.

We show that `Vittle` indeed reduces empirical JSD and EMID under distribution shifts in Table 4, and demonstrate in Section 4.2 that `Vittle`'s nice theoretical property is translated into consistent robustness gains under 30 distribution shift scenarios while maintaining in-distribution performance.

4 Experiment

4.1 Setup

Model and implementation detail. We adopt LLaVA-v1.5 [70] as our main baseline MLLM, where we set CLIP ViT-L/14-336px [71] as a vision encoder, Vicuna-v1.5-7B [72] as an LLM, and a two-layer MLP as a projector. We follow the standard two-stage training of LLaVA [38], and replicate stage-1 for image-text alignment with the same configuration and dataset (`LLaVA-pretrain-558k`) of LLaVA-v1.5 [70]. Then, on the `LLaVA-mix-665k`, we apply `Vittle` method by inserting the posterior MLP blocks and training the whole model with our IB objective. To validate the scalability and broad applicability, we also consider LLaVA-v1.5-13B, Prism-7B [73], LLaVA-Mini-Vicuna-7B [74], and LLaVA-Llama3-8B-Instruct [75]. Refer to Appendix A for additional details and Appendix B for the results on LLaVA-v1.5-13B and Prism-7B, respectively.

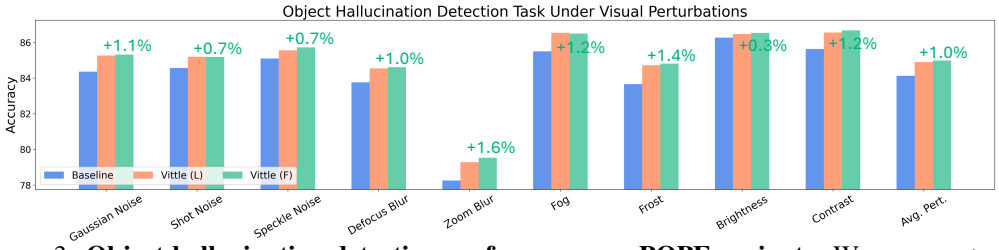


Figure 3: **Object hallucination detection performance on POPE variants**. We enumerate the hallucination detection accuracy of each method on nine versions of perturbed samples, and observe consistent gains by `Vittle` (highlighted by green numbers of relative improvement from baseline).

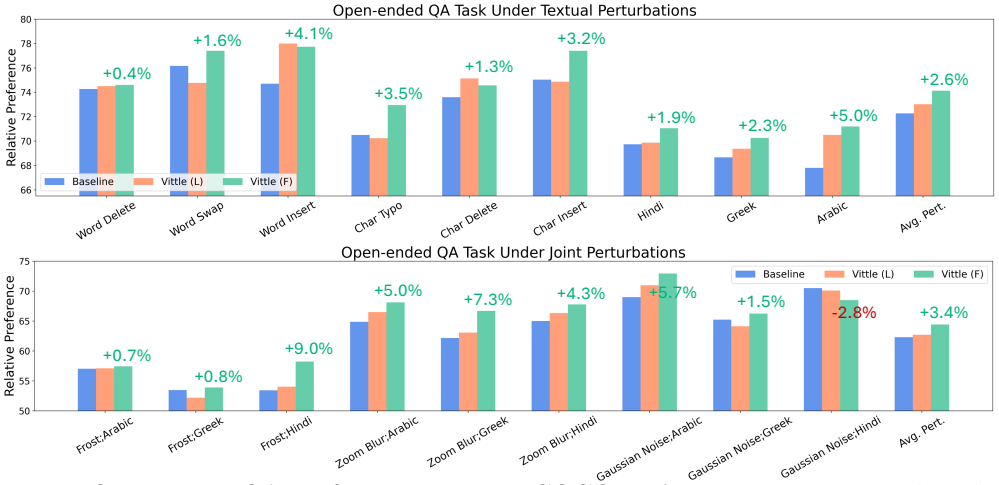


Figure 4: **Open-ended QA performance on LB-COCO variants**. We enumerate the relative preference score of responses from each model on 18 version of perturbed samples, and observe consistent gains by `Vittle` (especially for the `Vittle` (F)) on most of the textual (top), and joint (bottom) perturbations (results on visual perturbations are deferred to Appendix B).

Task, metric, and datasets. We evaluate instruction-tuned MLLMs with three representative tasks: (1) *open-ended question answering*, (2) *object hallucination detection*, and (3) *closed-form question answering*. All are formatted as a question answering (QA) with a single image input, where we use the average relative preference score measured by GPT-4o LLM judge [76] with three repeated runs for open-ended QA, while using exact matching accuracy for hallucination detection and closed-form QA. For open-ended QA tasks, we adopt four datasets: LB-COCO [38] as a *clean and typical* dataset, and LLaVA-Bench in-the-wild (LB-Wild), LLaVA-Bench-Wilder (LB-Wilder), and WildVision-Bench (WV-Bench) as *long-tail* datasets. Then, we apply 27 types of image and text perturbations on LB-COCO samples[4] to yield 28 variants of *perturbed* LB-COCO (one of clean and nine of visual, textual, and joint perturbations, respectively). For object hallucination detection tasks, we adopt POPE [77] as a *clean and typical* dataset. Then, we generate nine variants of *perturbed* POPE with visual perturbations. Here, we consider the LB-COCO and POPE as in-distribution (ID) datasets because they are generated from MS-COCO samples that construct majorities of the instruction tuning set of modern MLLMs, including LLaVA. For closed-form QA, we adopt four representative datasets: ScienceQA [78], MMMU [79], MME [5], and MMStar [80]. In summary, we experiment with **45 datasets** (31 of open-ended, 10 of object hallucination detection, and 4 of closed-form tasks).

4.2 Results

Vittle improves robustness under input perturbations. We first evaluate `Vittle` on object hallucination detection tasks with nine variants of POPE perturbed by visual corruptions in Figure 3. Although MLLMs trained with a standard objective and `Vittle` similarly suffer from perturbations, two instantiations of `Vittle` consistently outperform the standard objective. Interestingly, `Vittle` outperforms the baseline even in clean POPE (See Appendix B).

[4]See Appendix A.3 for a comprehensive summary of all perturbations and their generation processes.

We speculate that Vittle's information control prevents the reliance on a partial feature of a single modality [81], e.g., language prior [82], which is a common source of hallucination. Next, we present the validation on the open-ended QA task with 18 types of input perturbations, which are applied to visual and textual input independently or simultaneously in Figure 4. As we can see, Vittle greatly enhances performance in various perturbation datasets highlighted by green numbers that indicate the relative improvements of Vittle (F) compared to the baseline. Among the two variants of Vittle, Vittle (F) showcases better generalization under perturbations than Vittle (L), suggesting the benefits of a conservative zero-centered isotropic prior distribution to address a variety of subtle input perturbations. Next, we further explore Vittle's robustness by evaluating varying perturbation severity. To be specific, we generate perturbations on three different degrees that determine how significantly the image or text would be changed. In Figure 5, we see that Vittle achieves better performance in general, where the margin becomes larger under severe perturbations. In summary, we observe a consistent gain by Vittle on the perturbed input setting across two tasks, which indicates that Vittle enhances the robustness to distribution shifts by pursuing the minimal sufficiency of data representation.

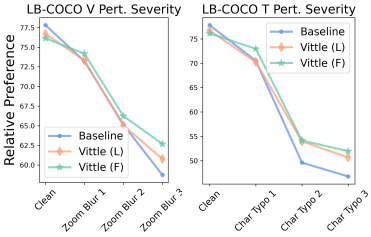


Figure 5: **Evaluation under varying perturbation severity.** Vittle achieves better performance, especially on severe perturbations.

Vittle improves generalization to long-tail distributions. Not only subtle perturbations on input, but long-tail samples are also commonly encountered in many MLLM applications. In Table 1, we validate Vittle on three long-tail QA tasks constructed with real-world user queries. We see that Vittle also excels in generalizing long-tailed samples compared to the baseline. Interestingly, Vittle (L)–learnable prior–exhibits better performance compared with Vittle (F). We speculate that a learnable prior IB guides the model to learn a better sensitivity for high-level abstractions as well as an invariance to low-level noise by allowing additional flexibility to shape data-driven priors, yielding superior performance on tasks that require in-depth understanding of irregular queries.

Table 1: **Performance comparison on long-tail open-ended QA tasks** those contain queries that are quite different from typical training samples in terms of visual content and textual semantics.

Method	LB-Wild	LB-Wilder	WV-Bench
Baseline	51.6	156.9	60.0
Vittle (L)	**54.6**	**168.8**	**60.4**
Vittle (F)	52.2	166.1	59.7

Table 2: **Performance comparison on general benchmark datasets**. These four multi-choice QA datasets require a higher level of multimodal understanding across multiple domains.

Method	SciQA	MMMU	MME	MMStar	Avg.
Baseline	64.6	**35.6**	69.7	**33.7**	50.9
Vittle (L)	64.7	35.3	**70.5**	**33.7**	**51.1**
Vittle (F)	**65.4**	34.5	70.1	33.5	50.9

Vittle preserves competitive performance on general benchmarks. Although the main focus of Vittle is to improve the model's robustness under distribution shifts, securing the rich multimodal understanding capability and knowledge to diverse disciplines is also crucial as an essence of MLLM. To validate this, we evaluate each method on four representative closed-form knowledge-intensive QA benchmark datasets covering various fields. In Table 2, we observe that Vittle shows competitive performance with the standard approach, which implies that Vittle can also be used as a general-purpose learning objective.

Table 3: **Comparison with weight-space compression methods.** We compare Vittle with the LoRA and weight decay (WD) methods on LB-COCO and its perturbed variants.

Method	Clean	V Pert.	T Pert.	J Pert.
Baseline	77.8	73.4	72.2	62.3
LoRA	73.4	70.4	62.7	39.7
WD	74.1	72.1	73.0	59.5
Vittle (L)	76.7	73.9	73.0	62.7
Vittle (F)	76.1	**74.2**	**74.1**	**64.4**

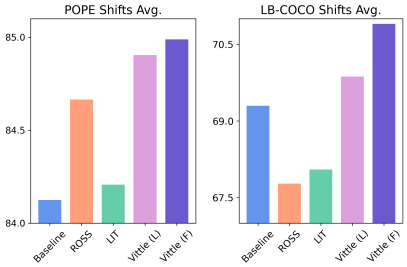


Figure 6: **Comparison with other objective functions.** We report the average performance for all perturbations in POPE and LB-COCO.

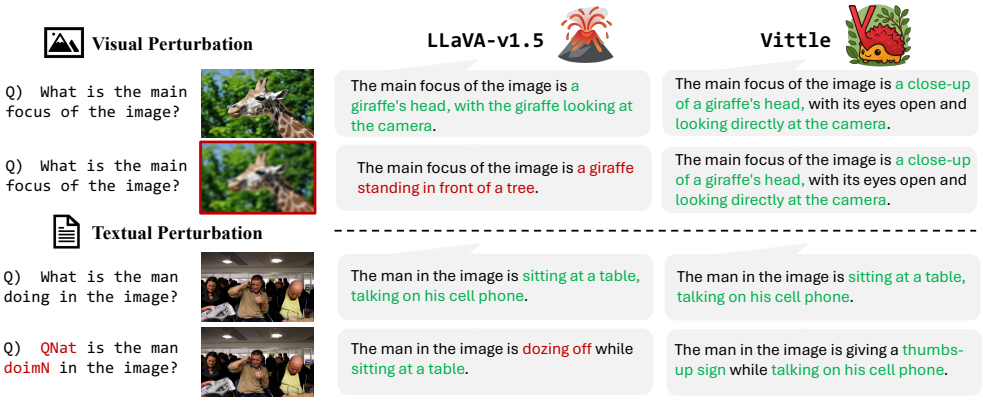


Figure 7: **Case study on LB-COCO under perturbations.** Although LLaVA-v1.5 produces a reasonable response for clean samples, the response and its quality vary under perturbations. Meanwhile, `Vittle` maintains the consistency for the responses.

Comparison with alternative learning approaches. Note that the regularization forced by `Vittle` works on the representation space to penalize the amount of information encoded in the data representations. One of the natural alternatives is to regularize the model weight directly. In Table 3, we compare `Vittle` with LoRA [83] and the weight decay method (WD) as instantiations of weight-space regularization, and the results suggest that explicit regularization on weight-space does not ensure a good balance between adaptability on in-distribution and robustness to distribution shifts. The other line of alternatives is ***information maximization*** during visual instruction tuning [84, 85], which is the exact opposite of `Vittle`'s design principle. We compare two recent methods on this line, ROSS [84] and LIT [85], with `Vittle` on LB-COCO and POPE under perturbations. As shown in Figure 6, although these approaches are effective in improving object hallucination detection performances, they fail to achieve competitive performance on the open-ended QA task (See Appendix B for more results). This implies the non-trivial challenge of devising a general learning objective for MLLMs that can consistently improve robustness across diverse tasks, where we can see the promise of `Vittle` towards broadly applicable robust instruction tuning.

Qualitative analysis. Figure 7 shows responses of clean queries and their visually or textually perturbed counterparts. Although the query before and after each perturbation conveys the same meaning and intention, LLaVA-v1.5 reveals volatility in its responses, whereas `Vittle` shows stable behavior by providing consistent responses, i.e., generating exactly the same response in the case of visual perturbation while keep focusing on the same object in the case of textual perturbation.

Representation analysis. We next see how `Vittle` shapes the representation space and how it affects robustness. In Table 4, we measure the average value of empirical JSD and EMID discussed in Section 3.4 over 27 perturbed variants of LB-COCO. Both JSD and EMID are computed between two distributions, clean and one of its perturbed versions, and then averaged over 27 clean-perturbed pairs (See Appendix A.4 for details). As our hypothesis, `Vittle` reduces distributional gaps, e.g., achieving smaller JSDs, between clean and perturbed samples in its representation space, thereby achieving a smaller EMID value that indicates better robustness. In Figure 8 (top), we further show PCA visualizations (in the same axis scale) for representations of LLaVA and `Vittle` on clean and image-text perturbed LB-COCO. We see that `Vittle` embeds the clean and semantically equivalent perturbed samples more closely. Moreover, the bottom panel shows that `Vittle` induces smaller cosine distances between clean and perturbed pairs in terms of the histogram and the average value in parentheses. These results indicate that our learning objective is indeed effective in structuring a better representation space that drives robustness.

Table 4: **JSD and EMID evaluation on 27 LB-COCO variants.**

Method	JSD (↓)	EMID (↓)
Baseline	0.068	0.026
`Vittle` (L)	0.048	0.021
`Vittle` (F)	0.047	0.025

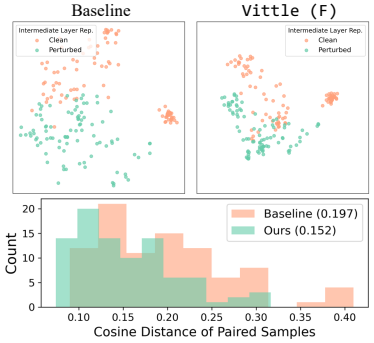


Figure 8: **PCA and pair-wise cosine distance of representations.**

Cost analysis. `Vittle` introduces a lightweight bottleneck layer inside of LLM that slightly increases the total number of trainable parameters (by 1.5%). One may thus wonder how `Vittle`'s training and inference time is compared with a bottleneck-free baseline. In Table 5, we show the wall-clock training (per iter and total), and inference time per sample. Although `Vittle` increases the training time up to 20% compared with baseline, *its inference time is almost identical to the original model*, which is a reasonable amount of cost overhead given significant gains in terms of robustness. We also compare the peak memory in Table 14, showing a negligible amount of memory overhead.

Table 5: **Runtime comparison.**

Method	Tr./it. (s)	Tr. (h)	Te. (s)
Baseline	7.363	11.06	0.1048
`Vittle` (F)	9.482	13.36	0.1072

Varying MLLM specification. As `Vittle` is a model-agnostic learning framework without architecture-specific constraint, we now explore `Vittle` training on two different MLLM specifications in Table 6: (1) LLaVA-Mini that has the same vision encoder and LLM as LLaVA-v1.5 but has different architectural design by incorporating visual token compressor and pre-modality fusion layer; and (2) LLaVA-Llama3-8B-Instruct (denoted in LLaVA++) that we just replace the LLM backbone with Llama3-8B-Instruct. As we can see, `Vittle` consistently outperforms the standard LLaVA training in various MLLM specifications, demonstrating its generality across different model architectures (See Appendix B.5 for more results).

Table 6: `Vittle` **with different model backbones.**

Backbone	Method	POPE	POPE Shifts Avg.
LLaVA-Mini [74]	Baseline	79.37	77.39
	`Vittle` (F)	**81.07**	**78.32**
LLaVA++ [75]	Baseline	84.60	80.54
	`Vittle` (F)	**85.87**	**84.08**

5 Conclusion

This work provided the first investigation on the promise of information bottleneck in the context of MLLM instruction tuning to ensure the robustness of MLLM under distribution shifts. We proposed a new theoretically-grounded visual instruction tuning method, `Vittle`. It injects a bottleneck layer inside the LLM to induce posterior samples of internal representations that encode useful information to produce valid responses while discarding other residual information from input queries. With negligible additional cost, `Vittle` is easily optimized with a variational lower bound of IB and shows consistent gains in robustness in 30 types of distribution shifts while also achieving competitive performance on standard benchmarks, indicating that `Vittle` promotes a good balance between invariance and sensitivity during representation learning.

Limitation and future work. One possible concern with `Vittle` is its reliance on the quality of Y, i.e., a gold response to given instruction, which is usually generated by another LLM. As disclosed by Yeh et al. [86], existing datasets for supervised fine-tuning are quite noisy, and we cannot ensure the advantage of IB on this noisy annotation setup. Moreover, IB alone does not guarantee the generalization of multi-modal counterfactual samples with conflicting language prior [87, 82] or samples from completely different domains [88, 89, 90], and may require additional annotations. Besides, we observed that `Vittle` slightly hurts the optical character recognition capability of MLLM, indicating the importance of further exploration to preserve the fine-grained recognition capability while pursuing robustness to distribution shifts. Investigating the potential of noisy annotation, counterfactual/domain generalization, and fine-grained recognition setups can be interesting future research problems. Meanwhile, the noise-robust representation space achieved by IB training can be helpful for representation steering methods [91, 92] that are also worth exploring for future work.

Acknowledgments and Disclosure of Funding

The authors thank Seongheon Park and Sean Xuefeng Du for their valuable suggestions and discussions that shaped the draft, and thank all the NeurIPS 2025 PCs, SACs, ACs, and anonymous reviewers. In addition, Changdae Oh particularly thanks Sanghyuk Chun for his sharp feedback on the manuscript. Research is supported in part by the AFOSR Young Investigator Program under award number FA9550-23-1-0184, National Science Foundation (NSF) Award No. IIS-2237037 and IIS-2331669, Office of Naval Research under grant number N00014-23-1-2643, Open Philanthropy, Schmidt Sciences Foundation, and Alfred P. Sloan Fellowship.

References

[1] OpenAI. Gpt-5 system card. Technical report, OpenAI, August 2025.

[2] Anthropic. Claude sonnet 4.5 system card. Technical report, Anthropic, September 2025.

[3] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

[4] xAI. Grok 4 model card. Technical report, xAI, August 2025.

[5] Zijing Liang, Yanjie Xu, Yifan Hong, Penghui Shang, Qi Wang, Qiang Fu, and Ke Liu. A survey of multimodel large language models. In *Proceedings of the 3rd International Conference on Computer, Artificial Intelligence and Control Engineering*, pages 405–409, 2024.

[6] Hao Yang, Yanyan Zhao, Yang Wu, Shilong Wang, Tian Zheng, Hongbo Zhang, Zongyang Ma, Wanxiang Che, and Bing Qin. Large language models meet text-centric multimodal sentiment analysis: A survey. *arXiv preprint arXiv:2406.08068*, 2024.

[7] Rawan AlSaad, Alaa Abd-Alrazaq, Sabri Boughorbel, Arfan Ahmed, Max-Antoine Renault, Rafat Damseh, and Javaid Sheikh. Multimodal large language models in health care: Applications, challenges, and future outlook. *J. Med. Internet Res.*, 26:e59505, 2024.

[8] Yiwei Li, Huaqin Zhao, Hanqi Jiang, Yi Pan, Zhengliang Liu, Zihao Wu, Peng Shu, Jie Tian, Tianze Yang, Shaochen Xu, et al. Large language models for manufacturing. *arXiv preprint arXiv:2410.21418*, 2024.

[9] Davide Caffagni, Federico Cocchi, Luca Barsellotti, Nicholas Moratelli, Sara Sarto, Lorenzo Baraldi, Marcella Cornia, and Rita Cucchiara. The revolution of multimodal large language models: A survey. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 13590–13618, 2024.

[10] Can Cui, Yunsheng Ma, Xu Cao, Wenqian Ye, Yang Zhou, Kaizhao Liang, Jintai Chen, Juanwu Lu, Zichong Yang, Kuei-Da Liao, Tianren Gao, Erlong Li, Kun Tang, Zhipeng Cao, Tong Zhou, Ao Liu, Xinrui Yan, Shuqi Mei, Jianguo Cao, Ziran Wang, and Chao Zheng. A survey on multimodal large language models for autonomous driving. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops*, pages 958–979, January 2024.

[11] Mark Turner. The art of compression. *The artful mind: Cognitive science and the riddle of human creativity*, pages 93–114, 2006.

[12] Eddie Gray and David Tall. Abstraction as a natural process of mental compression. *Mathematics Education Research Journal*, 19(2):23–40, 2007.

[13] George A Miller. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological review*, 63(2):81, 1956.

[14] Eleanor Rosch. Cognitive representations of semantic categories. *Journal of experimental psychology: General*, 104(3):192, 1975.

[15] Li Zhaoping. Vision: looking and seeing through our brain's information bottleneck. *arXiv preprint arXiv:2503.18804*, 2025.

[16] Xingxuan Zhang, Jiansheng Li, Wenjing Chu, Junjia Hai, Renzhe Xu, Yuqing Yang, Shikai Guan, Jiazheng Xu, and Peng Cui. On the out-of-distribution generalization of multimodal large language models. *arXiv preprint arXiv:2402.06599*, 2024.

[17] Zhongyi Han, Guanglin Zhou, Rundong He, Jindong Wang, Tailin Wu, Yilong Yin, Salman Khan, Lina Yao, Tongliang Liu, and Kun Zhang. How well does gpt-4v (ision) adapt to distribution shifts? a preliminary investigation. In *ICLR 2024 Workshop on Mathematical and Empirical Understanding of Foundation Models*, 2024.

[18] Moon Ye-Bin, Nam Hyeon-Woo, Wonseok Choi, and Tae-Hyun Oh. Beaf: Observing before-after changes to evaluate hallucination in vision-language models. In *European Conference on Computer Vision*, pages 232–248. Springer, 2025.

[19] Changdae Oh, Zhen Fang, Shawn Im, Xuefeng Du, and Yixuan Li. Understanding multimodal llms under distribution shifts: An information-theoretic approach. In *International Conference on Machine Learning*, 2025.

[20] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

[21] Wenqian Ye, Guangtao Zheng, Yunsheng Ma, Xu Cao, Bolin Lai, James Matthew Rehg, and Aidong Zhang. MM-spubench: Towards better understanding of spurious biases in multimodal LLMs. In *Workshop on Responsibly Building the Next Generation of Multimodal Foundational Models*, 2024.

[22] Yiming Liang, Tianyu Zheng, Xinrun Du, Ge Zhang, Jiaheng Liu, Xingwei Qu, Wenqiang Zu, Xingrun Xing, Chujie Zheng, Lei Ma, et al. Aligning instruction tuning with pre-training. *arXiv preprint arXiv:2501.09368*, 2025.

[23] Bo Zhao, Boya Wu, Muyang He, and Tiejun Huang. Svit: Scaling up visual instruction tuning. *arXiv preprint arXiv:2307.04087*, 2023.

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

[25] Shuhao Gu, Jialing Zhang, Siyuan Zhou, Kevin Yu, Zhaohu Xing, Liangdong Wang, Zhou Cao, Jintao Jia, Zhuoyi Zhang, Yixuan Wang, et al. Infinity-mm: Scaling multimodal performance with large-scale and high-quality instruction data. *arXiv preprint arXiv:2410.18558*, 2024.

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

[27] Yunhang Shen, Chaoyou Fu, Shaoqi Dong, Xiong Wang, Peixian Chen, Mengdan Zhang, Haoyu Cao, Ke Li, Xiawu Zheng, Yan Zhang, et al. Long-vita: Scaling large multi-modal models to 1 million tokens with leading short-context accuray. *arXiv preprint arXiv:2502.05177*, 2025.

[28] Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*, 2024.

[29] Peter Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Adithya Jairam Vedagiri IYER, Sai Charitha Akula, Shusheng Yang, Jihan Yang, Manoj Middepogu, Ziteng Wang, et al. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. *Advances in Neural Information Processing Systems*, 37:87310–87356, 2024.

[30] Min Shi, Fuxiao Liu, Shihao Wang, Shijia Liao, Subhashree Radhakrishnan, Yilin Zhao, De-An Huang, Hongxu Yin, Karan Sapra, Yaser Yacoob, Humphrey Shi, Bryan Catanzaro, Andrew Tao, Jan Kautz, Zhiding Yu, and Guilin Liu. Eagle: Exploring the design space for multimodal LLMs with mixture of encoders. In *The Thirteenth International Conference on Learning Representations*, 2025.

[31] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[32] Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. Deep variational information bottleneck. In *International Conference on Learning Representations*, 2017.

[33] Matias Vera, Pablo Piantanida, and Leonardo Rey Vega. The role of the information bottleneck in representation learning. In *2018 IEEE International Symposium on Information Theory (ISIT)*, pages 1580–1584, 2018.

[34] Tailin Wu, Hongyu Ren, Pan Li, and Jure Leskovec. Graph information bottleneck. *Advances in Neural Information Processing Systems*, 33:20437–20448, 2020.

[35] Rabeeh Karimi Mahabadi, Yonatan Belinkov, and James Henderson. Variational information bottleneck for effective low-resource fine-tuning. *arXiv preprint arXiv:2106.05469*, 2021.

[36] Yawei Li, David Rügamer, Bernd Bischl, and Mina Rezaei. Calibrating LLMs with information-theoretic evidential deep learning. In *The Thirteenth International Conference on Learning Representations*, 2025.

[37] Thomas M Cover. *Elements of information theory*. John Wiley & Sons, 1999.

[38] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.

[39] Wenliang Dai, Junnan Li, DONGXU LI, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Advances in Neural Information Processing Systems*, 36:49250–49267, 2023.

[40] Tony Lee, Haoqin Tu, Chi Heem Wong, Wenhao Zheng, Yiyang Zhou, Yifan Mai, Josselin Roberts, Michihiro Yasunaga, Huaxiu Yao, Cihang Xie, et al. Vhelm: A holistic evaluation of vision language models. *Advances in Neural Information Processing Systems*, 37:140632–140666, 2024.

[41] Xiang Li, Congcong Wen, Yuan Hu, Zhenghang Yuan, and Xiao Xiang Zhu. Vision-language models in remote sensing: Current progress and future trends. *IEEE Geoscience and Remote Sensing Magazine*, 2024.

[42] Mubashar Raza, Zarmina Jahangir, Muhammad Bilal Riaz, Muhammad Jasim Saeed, and Muhammad Awais Sattar. Industrial applications of large language models. *Scientific Reports*, 15(1):13755, 2025.

[43] Jielin Qiu, Yi Zhu, Xingjian Shi, Florian Wenzel, Zhiqiang Tang, Ding Zhao, Bo Li, and Mu Li. Benchmarking robustness of multimodal image-text models under distribution shift. *Journal of Data-centric Machine Learning Research*, 2024.

[44] Xuanming Cui, Alejandro Aparcedo, Young Kyun Jang, and Ser-Nam Lim. On the robustness of large multimodal models against image adversarial attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24625–24634, 2024.

[45] Aayush Atul Verma, Amir Saeidi, Shamanthak Hegde, Ajay Therala, Fenil Denish Bardoliya, Nagaraju Machavarapu, Shri Ajay Kumar Ravindhiran, Srija Malyala, Agneet Chatterjee, Yezhou Yang, et al. Evaluating multimodal large language models across distribution shifts and augmentations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5314–5324, 2024.

[46] Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

[47] Naftali Tishby and Noga Zaslavsky. Deep learning and the information bottleneck principle. In *2015 ieee information theory workshop (itw)*, pages 1–5. Ieee, 2015.

[48] Yawei Luo, Ping Liu, Tao Guan, Junqing Yu, and Yi Yang. Significance-aware information bottleneck for domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6778–6787, 2019.

[49] Marco Federici, Anjan Dutta, Patrick Forré, Nate Kushman, and Zeynep Akata. Learning robust representations via multi-view information bottleneck. In *International Conference on Learning Representations*, 2020.

[50] Siqi Miao, Mia Liu, and Pan Li. Interpretable and generalizable graph learning via stochastic attention mechanism. In *International Conference on Machine Learning*, pages 15524–15543. PMLR, 2022.

[51] Zichuan Liu, Tianchun Wang, Jimeng Shi, Xu Zheng, Zhuomin Chen, Lei Song, Wenqian Dong, Jayantha Obeysekera, Farhad Shirani, and Dongsheng Luo. Timex++: Learning time-series explanations with information bottleneck. In *International Conference on Machine Learning*, pages 32062–32082. PMLR, 2024.

[52] Kenji Kawaguchi, Zhun Deng, Xu Ji, and Jiaoyang Huang. How does information bottleneck help deep learning? In *International Conference on Machine Learning*, pages 16049–16096. PMLR, 2023.

[53] Zifeng Wang, Tong Jian, Aria Masoomi, Stratis Ioannidis, and Jennifer Dy. Revisiting hilbert-schmidt information bottleneck for adversarial robustness. *Advances in Neural Information Processing Systems*, 34:586–597, 2021.

[54] Jiaqi Bai, Hongcheng Guo, Zhongyuan Peng, Jian Yang, Zhoujun Li, Mohan Li, and Zhihong Tian. Mitigating hallucinations in large vision-language models by adaptively constraining information flow. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 23442–23450, 2025.

[55] Alex Graves. Practical variational inference for neural networks. *Advances in neural information processing systems*, 24, 2011.

[56] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations (ICLR)*, 2014.

[57] David M. Blei, Alp Kucukelbir, and Jon D. McAuliffe and. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017.

[58] Seong Joon Oh, Andrew C. Gallagher, Kevin P. Murphy, Florian Schroff, Jiyan Pan, and Joseph Roth. Modeling uncertainty with hedged instance embeddings. In *International Conference on Learning Representations*, 2019.

[59] Sanghyuk Chun, Seong Joon Oh, Rafael Sampaio De Rezende, Yannis Kalantidis, and Diane Larlus. Probabilistic embeddings for cross-modal retrieval. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8415–8424, 2021.

[60] Sanghyuk Chun. Improved probabilistic image-text representations. In *The Twelfth International Conference on Learning Representations*, 2024.

[61] Sanghyuk Chun, Wonjae Kim, Song Park, and Sangdoo Yun. Probabilistic language-image pre-training. In *The Thirteenth International Conference on Learning Representations*, 2025.

[62] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[63] Cheolhyoung Lee, Kyunghyun Cho, and Wanmo Kang. Mixout: Effective regularization to finetune large-scale pretrained language models. In *International Conference on Learning Representations*, 2020.

[64] Bingbing Wen, Chenjun Xu, Robert Wolfe, Lucy Lu Wang, Bill Howe, et al. Mitigating overconfidence in large language models: A behavioral lens on confidence estimation and calibration. In *NeurIPS 2024 Workshop on Behavioral Machine Learning*, 2024.

[65] Haoyan Yang, Yixuan Wang, Xingyin Xu, Hanyuan Zhang, and Yirong Bian. Can we trust llms? mitigate overconfidence bias in llms through knowledge transfer. *arXiv preprint arXiv:2405.16856*, 2024.

[66] Tobias Groot and Matias Valdenegro-Toro. Overconfidence is key: Verbalized uncertainty evaluation in large language and vision-language models. In *Proceedings of the 4th Workshop on Trustworthy Natural Language Processing (TrustNLP 2024)*, pages 145–171, 2024.

[67] Alexander Alemi, Ben Poole, Ian Fischer, Joshua Dillon, Rif A Saurous, and Kevin Murphy. Fixing a broken elbo. In *International conference on machine learning*, pages 159–168. PMLR, 2018.

[68] Hao Fu, Chunyuan Li, Xiaodong Liu, Jianfeng Gao, Asli Celikyilmaz, and Lawrence Carin. Cyclical annealing schedule: A simple approach to mitigating kl vanishing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 240–250, 2019.

[69] Yixin Wang, David Blei, and John P Cunningham. Posterior collapse and latent variable non-identifiability. *Advances in neural information processing systems*, 34:5443–5455, 2021.

[70] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306, 2024.

[71] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

[72] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023.

[73] Siddharth Karamcheti, Suraj Nair, Ashwin Balakrishna, Percy Liang, Thomas Kollar, and Dorsa Sadigh. Prismatic vlms: Investigating the design space of visually-conditioned language models. In *Forty-first International Conference on Machine Learning*, 2024.

[74] Shaolei Zhang, Qingkai Fang, Zhe Yang, and Yang Feng. LLaVA-mini: Efficient image and video large multimodal models with one vision token. In *The Thirteenth International Conference on Learning Representations*, 2025.

[75] Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad S. Khan. Llava++: Extending visual capabilities with llama-3 and phi-3, 2024.

[76] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623, 2023.

[77] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 292–305, 2023.

[78] Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521, 2022.

[79] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9556–9567, 2024.

[80] Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, and Feng Zhao. Are we on the right way for evaluating large vision-language models? In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[81] Pooyan Rahmanzadehgervi, Logan Bolton, Mohammad Reza Taesiri, and Anh Totti Nguyen. Vision language models are blind. In *Proceedings of the Asian Conference on Computer Vision*, pages 18–34, 2024.

[82] Lin Long, Changdae Oh, Seongheon Park, and Sharon Li. Understanding language prior of lvlms by contrasting chain-of-embedding. *arXiv preprint arXiv:2509.23050*, 2025.

[83] Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

[84] Haochen Wang, Anlin Zheng, Yucheng Zhao, Tiancai Wang, Zheng Ge, Xiangyu Zhang, and Zhaoxiang Zhang. Reconstructive visual instruction tuning. In *The Thirteenth International Conference on Learning Representations*, 2025.

[85] Zhihan Zhou, Feng Hong, Jiaan Luo, Jiangchao Yao, Dongsheng Li, Bo Han, Ya Zhang, and Yanfeng Wang. Learning to instruct for visual instruction tuning. *arXiv preprint arXiv:2503.22215*, 2025.

[86] Min-Hsuan Yeh, Jeffrey Wang, Xuefeng Du, Seongheon Park, Leitian Tao, Shawn Im, and Yixuan Li. Position: Challenges and future directions of data-centric AI alignment. In *Forty-second International Conference on Machine Learning Position Paper Track*, 2025.

[87] Kang-il Lee, Minbeom Kim, Seunghyun Yoon, Minsung Kim, Dongryeol Lee, Hyukhun Koh, and Kyomin Jung. Vlind-bench: Measuring language priors in large vision-language models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 4129–4144, 2025.

[88] Yingjun Du, Jun Xu, Huan Xiong, Qiang Qiu, Xiantong Zhen, Cees GM Snoek, and Ling Shao. Learning to learn with variational information bottleneck for domain generalization. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*, pages 200–216. Springer, 2020.

[89] Bo Li, Yifei Shen, Yezhen Wang, Wenzhen Zhu, Dongsheng Li, Kurt Keutzer, and Han Zhao. Invariant information bottleneck for domain generalization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 7399–7407, 2022.

[90] Jiao Zhang, Xu-Yao Zhang, Chuang Wang, and Cheng-Lin Liu. Deep representation learning for domain generalization with information bottleneck principle. *Pattern Recognition*, 143:109737, 2023.

[91] Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, et al. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*, 2023.

[92] Sheng Liu, Haotian Ye, and James Zou. Reducing hallucinations in large vision-language models via latent space steering. In *The Thirteenth International Conference on Learning Representations*, 2025.

[93] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021.

[94] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[95] Vicente Ordonez, Girish Kulkarni, and Tamara Berg. Im2text: Describing images using 1 million captioned photographs. *Advances in neural information processing systems*, 24, 2011.

[96] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR, 2022.

[97] Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. Shikra: Unleashing multimodal llm's referential dialogue magic. *arXiv preprint arXiv:2306.15195*, 2023.

[98] Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, et al. mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*, 2023.

[99] Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9568–9578, 2024.

[100] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11975–11986, 2023.

[101] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. Featured Certification.

[102] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021.

[103] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part v 13*, pages 740–755. Springer, 2014.

[104] Bo Li, Kaichen Zhang, Hao Zhang, Dong Guo, Renrui Zhang, Feng Li, Yuanhan Zhang, Ziwei Liu, and Chunyuan Li. Llava-next: Stronger llms supercharge multimodal capabilities in the wild, May 2024.

[105] Yujie Lu, Dongfu Jiang, Wenhu Chen, William Yang Wang, Yejin Choi, and Bill Yuchen Lin. Wildvision: Evaluating vision-language models in the wild with human preferences. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024.

[106] Pengyu Cheng, Weituo Hao, Shuyang Dai, Jiachang Liu, Zhe Gan, and Lawrence Carin. Club: A contrastive log-ratio upper bound of mutual information. In *International conference on machine learning*, pages 1779–1788. PMLR, 2020.

[107] A Conneau. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*, 2019.

[108] Jhoan K Hoyos-Osorio and Luis G Sanchez-Giraldo. The representation jensen-shannon divergence. *arXiv preprint arXiv:2305.16446*, 2023.

[109] Oscar Skean, Md Rifat Arefin, Dan Zhao, Niket Patel, Jalal Naghiyev, Yann LeCun, and Ravid Shwartz-Ziv. Layer by layer: Uncovering hidden representations in language models. *arXiv preprint arXiv:2502.02013*, 2025.

[110] Evan Hernandez and Jacob Andreas. The low-dimensional linear geometry of contextualized word representations. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 82–93, 2021.

[111] Zhuo-Yang Song, Zeyu Li, Qing-Hong Cao, Ming-xing Luo, and Hua Xing Zhu. Bridging the dimensional chasm: Uncover layer-wise dimensional reduction in transformers through token correlation. *arXiv preprint arXiv:2503.22547*, 2025.

[112] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. A diversity-promoting objective function for neural conversation models. *arXiv preprint arXiv:1510.03055*, 2015.

[113] Chantal Shaib, Joe Barrow, Jiuding Sun, Alexa F Siu, Byron C Wallace, and Ani Nenkova. Standardizing the measurement of text diversity: A tool and a comparative analysis of scores. *arXiv preprint arXiv:2403.00553*, 2024.

[114] Sanjeev Arora, Rong Ge, Behnam Neyshabur, and Yi Zhang. Stronger generalization bounds for deep nets via a compression approach. In *International conference on machine learning*, pages 254–263. PMLR, 2018.

[115] Nick Littlestone and Manfred Warmuth. Relating data compression and learnability. *Unpublished notes*, 1986.

[116] Anselm Blumer, Andrzej Ehrenfeucht, David Haussler, and Manfred K. Warmuth. Occam's razor. *Information Processing Letters*, 24(6):377–380, 1987.

[117] J. Rissanen. Modeling by shortest data description. *Automatica*, 14(5):465–471, 1978.

[118] Geoffrey E Hinton and Drew Van Camp. Keeping the neural networks simple by minimizing the description length of the weights. In *Proceedings of the sixth annual conference on Computational learning theory*, pages 5–13, 1993.

[119] Peter Grünwald. Minimum description length tutorial. *Advances in minimum description length: Theory and applications*, 5:1–80, 2005.

[120] Milad Sefidgaran, Abdellatif Zaidi, and Piotr Krasnowski. Minimum description length and generalization guarantees for representation learning. *Advances in Neural Information Processing Systems*, 36:1489–1525, 2023.

[121] Horace B Barlow et al. Possible principles underlying the transformation of sensory messages. *Sensory communication*, 1(01):217–233, 1961.

[122] Andrew Gordon Wilson. Deep learning is not so mysterious or different. *arXiv preprint arXiv:2503.02113*, 2025.

[123] Marc Finzi, Gregory Benton, and Andrew G Wilson. Residual pathway priors for soft equivariance constraints. *Advances in Neural Information Processing Systems*, 34:30037–30049, 2021.

[124] Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. In *International Conference on Learning Representations*, 2022.

[125] Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, et al. Robust fine-tuning of zero-shot models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7959–7971, 2022.

[126] Yoonho Lee, Annie S Chen, Fahim Tajwar, Ananya Kumar, Huaxiu Yao, Percy Liang, and Chelsea Finn. Surgical fine-tuning improves adaptation to distribution shifts. In *The Eleventh International Conference on Learning Representations*, 2023.

[127] Sachin Goyal, Ananya Kumar, Sankalp Garg, Zico Kolter, and Aditi Raghunathan. Finetune like you pretrain: Improved finetuning of zero-shot vision models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19338–19347, 2023.

[128] Junjiao Tian, Zecheng He, Xiaoliang Dai, Chih-Yao Ma, Yen-Cheng Liu, and Zsolt Kira. Trainable projected gradient method for robust fine-tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7836–7845, 2023.

[129] Changdae Oh, Mijoo Kim, Hyesu Lim, Junhyeok Park, Euiseog Jeong, Zhi-Qi Cheng, and Kyungwoo Song. Towards calibrated robust fine-tuning of vision-language models. *Advances in Neural Information Processing Systems*, 37, 2024.

[130] Jaedong Hwang, Brian Cheung, Zhang-Wei Hong, Akhilan Boopathy, Pulkit Agrawal, and Ila Fiete. Imagenet-rib benchmark: Large pre-training datasets don't guarantee robustness after fine-tuning. *arXiv preprint arXiv:2410.21582*, 2024.

[131] Changdae Oh, Yixuan Li, Kyungwoo Song, Sangdoo Yun, and Dongyoon Han. Dawin: Training-free dynamic weight interpolation for robust adaptation. In *The Thirteenth International Conference on Learning Representations*, 2025.

[132] Fuxiao Liu, Kevin Lin, Linjie Li, Jianfeng Wang, Yaser Yacoob, and Lijuan Wang. Mitigating hallucination in large multi-modal models via robust instruction tuning. In *The Twelfth International Conference on Learning Representations*, 2024.

[133] Wei Han, Hui Chen, and Soujanya Poria. Towards robust instruction tuning on multimodal large language models. *arXiv preprint arXiv:2402.14492*, 2024.

[134] Alessandro Achille and Stefano Soatto. Information dropout: Learning optimal representations through noisy computation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2897–2905, 2018.

[135] Changjian Shui, Qi Chen, Jun Wen, Fan Zhou, Christian Gagné, and Boyu Wang. A novel domain adaptation theory with jensen–shannon divergence. *Knowledge-Based Systems*, 257:109808, 2022.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: We explicitly mentioned our research problem with scope and contribution in the abstract and introduction sections. These are consistently stated through the remaining sections (Sec 2 and 3), and supported by empirical evidence 4.2.

   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: We included the limitations of this work at Appendix **??** due to space constraint of the main paper.

   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: [Yes]

Justification: We clearly elaborated the full set of assumptions and proof in Appendix E.

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: In Section 4.1 and Section A, we provided the full experimental details for easy reproduction, and we also provide our code through the GitHub repository link in the abstract.

   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: [Yes]

Justification: We do provide the entire code for the full reproduction (from training to evaluation) with an appropriate level of instructions.

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: We provided all the training and test details, including data splits, hyperparameters, how they were chosen, type of optimizer in Appendix Section A.

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: [No]

Justification: Given the fact that this is a work on full fine-tuning of large-scale ($\geq$ 7B) models, learning multiple training runs to claim statistical significance was computationally intractable for us, but we conduct multiple runs for the GPT-4o-Judge-based evaluation to provide stable conclusion and mentioned it correctly.

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: We included the computation resource-related details in Appendix A

   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: The authors acknowledged the NeurIPS Code of Ethics, and this work conforms to 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: [Yes]

    Justification: We provide an impact statement section (Section F) to introduce potential societal impacts of this work.

    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: [Yes]

    Justification: As we propose a learning objective for (M)LLM, it inherently has the risk of misuse of LLM, and we discuss this in Section F.

    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: In the implementation details section (Section A) and the instruction file of our code, we clearly mentioned the source of all assets.

    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: [Yes]

    Justification: We provide our code through the anonymous repository link, and the code supports the generation of our new datasets.

    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: There is no crowdsourcing and research with human subjects.

    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: There is no crowdsourcing and 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: We only used LLM for the purpose of paper editing, logo generation, and model evaluation for the open-ended question-answer task following the academic benchmarking standard.

    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.

Appendix

Contents

A Extended Experiment Setup and Implementation Detail

A.1 Model and Training

In this work, we consider LLaVA-v1.5 [70] as our target multimodal large language model (MLLM) with CLIP ViT-L/14-336px [71] and Vicuna-v1.5-7B [72] as visual encoder and LLM backbone, respectively, and a two-layer MLP as projector (modality connector that maps features of the visual encoder into the text embedding space). Although all of the results presented in the main body of the paper were produced with LLaVA-v1.5-7B, we also experimented with LLaVA-v1.5-13B (with Vicua-v1.5-13B as the LLM backbone) to validate the scalability of our method, and consider Prism-7B [73] as an additional MLLM architecture to validate the broad applicability of `Vittle`. For fair comparison, all models are trained on the `LLaVA-pretrain-558k` and `LLaVA-mix-665k` datasets, consisting of a mixture of LAION [93], CC [94], SBU [95] datasets with BLIP captions [96] and a mixture of `LLaVA-instruct-158K` and academic-task-oriented (V)QA datasets, respectively. Training configurations such as optimizer, learning rate, and batch size are summarized in Table 7 and Table 8. All training runs are conducted with eight A100-80GB GPUs with DeepSpeed ZeRO library. The shortest run takes roughly 11 hours, whereas the longest run takes about 14 hours. Now, we elaborate on the overall workflow of LLaVA and Prism below.

Table 7: **Hyperparameter list of `Vittle` training.** We adopt exactly the same configurations with LLaVA-v1.5 [70] for Stage 1 and 2.

Config	Stage1	Stage2
Global batch size	256	128
Batch size per GPU	32	16
Learning rate	1e-3	2e-5
Learning rate schedule	Cosine decay w/ linear warmup	
Warmup ratio	0.03	
Weight decay	0.0	
Epoch	1	
Optimizer	AdamW	
Precision	bf16	

Table 8: **Hyperparameter list of `Prism-Vittle` training.** We adopt exactly the same configurations with Prism-DINOSigLIP-Controlled-7B [73] single stage training.

Config	Value
Global batch size	128
Batch size per GPU	16
Learning rate	2e-5
Learning rate schedule	cosine decay w/ linear-warmup
Warmup ratio	0.03
Weight decay	0.1
Epoch	1
Optimizer	AdamW
Precision	bf16

LLaVA is built with a pre-trained visual encoder that takes visual inputs, a pre-trained LLM backbone that takes text instructions, and a lightweight projector that maps features produced by the visual encoder into the text embedding space of LLM backbone so that the visually-grounded multimodal instruction input query can be processed by the LLM backbone. LLaVA undergoes a two-stage training: (1) The first stage takes into account modality alignment, where the projector is trained on image and corresponding instruction or caption with a conditional language modeling loss implemented by aggregating cross-entropy losses across response tokens, while the visual encoder and LLM backbone are frozen. (2) The second stage stands for the instruction tuning, where the projector and LLM backbone are jointly trained on multimodal instruction samples with the same conditional language modeling loss while the visual encoder is still frozen. This two-stage training has been considered a standard approach for developing MLLMs and is widely adopted [97, 98, 29].

Prism has a model architecture similar to LLaVA, but provides some valuable insight into the design of the MLLM training recipe, and we note two remarkable design choices of Prism that distinguish it from LLaVA: (1) incorporating multiple visual encoders rather than hosting a single visual encoder, and (2) reducing the two-stage alignment-then-instruction tuning into a single-stage instruction tuning. Note that different self-supervised visual representation learning induces features that have different strengths, and several works reveal the benefits of ensembling multiple different visual encoders to leverage complementary advantages [99, 73, 30]. Prism incorporates SigLIP [100] and DINOv2 [101] to enjoy both a robust global feature and a fine-grained local feature. Meanwhile, Karamcheti et al. [73] showed that the simplified single-stage training strategy can be a cost-effective alternative to the standard two-stage training.

To train these MLLMs, we consider five baseline approaches: (1) the standard full LLM fine-tuning with conditional language modeling loss, (2) parameter-efficient LoRA [83] fine-tuning with the conditional language modeling loss, (3) conditional language modeling loss with weight decay regularization, (4) reconstructive visual instruction tuning (ROSS) [84], and (5) learning to instruct (LIT) [85]. For the LoRA-based training configuration, we use the same one provided by the official LLaVA-v1.5 repository[5], and for the weight decay regularization, we select the regularization magnitude parameter among $\{0.1, 0.01, 0.001\}$ based on the POPE evaluation result. We now elucidate two competitive baseline methods, ROSS and LIT, in the following paragraphs. It is worth noting that these methods are designed to encode more (visual) information into the representation space, which is opposite to our `Vittle`'s design motivation that pursues a minimal sufficient representation for improving robustness to distribution shifts.

Reconstructive visual instruction tuning (ROSS) follows the two-stage training of LLaVA, but tries to reconstruct the visual inputs from the LLM backbone by adopting a regression or denoising learning objective in addition to language modeling loss during its second stage. By doing so, ROSS guides the MLLM to learn a much richer visual understanding, which is usually lacking in modern MLLMs [99, 81]. The reconstruction target can be a raw RGB pixel value or the latent representation from an external visual encoder such as VQGAN [102] or VAE [56], and ROSS requires an additional trainable module to reconstruct visual content, which is discarded during inference. We follow the training recipe from the official code repository[6] to replicate ROSS-D-7B with the same visual encoder and LLM backbone to LLaVA-v1.5. For a fair comparison with LLaVA and `Vittle`, we

[5] https://github.com/haotian-liu/LLaVA

[6] https://github.com/Haochen-Wang409/ross

train the ROSS with the same dataset (that of LLaVA-v1.5) for both training stages, while the original ROSS model was trained on a slightly larger dataset in the second stage.

Learning to Instruct (LIT) also focuses on the visual shortcomings of current MLLM and tries to improve the visual understanding capability of MLLM by incorporating an additional loss term that incentivizes the encoding of additional visual information. To be specific, while the cross-entropy loss in LLaVA's conditional language modeling objective is aggregated through the response tokens only, LIT introduces an extra cross-entropy loss term, which is aggregated over the instruction (question) tokens only, thereby enforcing MLLM to learn to predict a proper textual instruction given an image. As LIT uses the same visual and language backbone model and training dataset as LLaVA-v1.5, we use the pre-trained checkpoint of LIT from Hugging Face[7] for evaluation.

In Figure 6, we observe that while ROSS and LIT are somewhat effective in improving performance on object hallucination detection tasks with the aid of enhanced visual understanding capability, they significantly underperform `Vittle` and even the original LLaVA on the open-ended QA task under distribution shifts. This implies that pursuing more information encoding during visual instruction tuning may not result in better robustness to distribution shifts, but aiming to learn a minimal sufficient representation via `Vittle` can be a promising solution for this (See Table 12 for details).

A.2 `Vittle` Implementation Details

This section provides additional details on implementing `Vittle` through Python-style pseudo code in Figure 9 and text below. Following the standard two-stage LLaVA training recipe, we freeze the visual encoder and LLM backbone during the first stage and only train the projector module. In the second stage, `Vittle` inserts a bottleneck layer g_ϕ, consisting of two of the two-layer MLPs $\{g_{\phi_v}, g_{\phi_t}\}$ for visual and textual modalities, inside the LLM backbone to estimate the distributional parameters (mean and diagonal covariance) of the posterior Gaussian distributions for each visual and textual token. Each bottleneck module is constructed with {`nn.Linear(d,d)`, `nn.GELU()`, `nn.Linear(d,2*d)`} where `d` denotes the hidden dimension of the LLM backbone, and this results in a slightly increased number of model parameters (up to 1.5% from the baseline). We use these estimated distribution parameters to sample a representation from this posterior via $\tilde{z} = \mu + \sigma \odot \epsilon$ where $\epsilon \sim \mathcal{N}(\mathbf{0}, I)$. Then, for a given bottleneck layer index l and for the maximum length of visual M_v and textual input tokens M_t, the bottleneck layer g_ϕ takes a sequence of token representations $z = \{z_{v,1}, ..., z_{v,M_v}, z_{t,1}, ..., z_{t,M_t}\}$ produced from the layer l to build information-penalized representations $\hat{z} = \{\hat{z}_{v,1}, ..., \hat{z}_{v,M_v}, \tilde{z}_{t,1}, ..., \tilde{z}_{t,M_t}\}$, where $\hat{z} = (1-\alpha)z + \alpha g_\phi(z)$. Here, we use an interpolated representation between the original pre-bottleneck representation z and the post-bottleneck representation $g_\phi(z)$ with an interpolation coefficient α that progressively grows from 0 to 0.5 by a cosine schedule during training. We observe that solely using the post-bottleneck representation induces a diverging language modeling loss at the later steps of training, and speculate that it is hard to generate a valid response with the information-penalized representation only[8].

Then, we jointly train the LLM backbone, the projector, and this bottleneck layer together during the second stage of training with the objective function 5. As we assume a diagonal covariance Gaussian for the prior and posterior distributions, Kullback–Leibler divergence (KLD) between the prior p and posterior q can be easily expressed as below,

$$D_{\mathrm{KL}}(q, p) = \frac{1}{2} \sum_{j=1}^{d} \big(\log \frac{\sigma_p^2[j]}{\sigma_q^2[j]} - 1 + \frac{(\mu_p[j] - \mu_q[j])^2}{\sigma_p^2[j]} + \frac{\sigma_q^2[j]}{\sigma_p^2[j]} \big) \tag{8}$$

where $\mu_\cdot$ and $\sigma_\cdot$ denote d-dimensional distributional parameter vectors and $[j]$ indicates j-th element from the vectors. `Vittle` has two important hyperparameters: (1) target layer index l for bottleneck application, and (2) posterior KLD regularization strength parameter β. After tuning across $l \in \{24, 28, 31\}$ and $\beta \in \{\frac{0.01}{d}, \frac{0.05}{d}, \frac{0.1}{d}, \frac{0.2}{d}, \frac{1.0}{d}\}$ where d denotes the latent dimension of the LLM backbone, we set $l = 24$ and $\beta = 0.1/d$ based on the average performance of POPE and LB-COCO clean datasets. The interpolation coefficient α can also be tuned, but we found that increasing α beyond 0.5 hinders stable training and observing increased language modeling loss at the later parts of training progress. Figure 12 and Table 11 present the results of hyperparameter ablation study.

[7] https://huggingface.co/zhihanzhou/LIT-LLaVA-1.5-Vicuna-7B/tree/main

[8] We suspect that this is because the initial parameter of MLLM's instruction tuning are far from random standard Gaussian or zero therefore induces unstable learning signal from the KLD term.

```
def reparam(mu, logvar):
    std = (logvar / 2).exp()
    batch_size, seq_len, hidden_dim = mu.shape
    z = torch.randn(batch_size, seq_len, hidden_dim)
    return mu + std * z

def forward(self, input_embeds, img_seq_len, a, **kwargs):
    ...
    hidden_states = input_embeds
    for l_idx, llm_layer in enumerate(self.llm.layers):
        layer_outputs = llm_layer(hidden_states, **kwargs)
        hidden_states = layer_outputs[0]
        if l_idx == self.bottleneck_layer_idx:
            # posterior inference
            v_params = self.g_v(hidden_states[:,:i_seq_len,:])
            t_params = self.g_t(hidden_states[:,i_seq_len:,:])
            v_mean = v_params[:,:,:self.h_dim]
            v_logvar = v_params[:,:,self.h_dim:]
            t_mean = t_params[:,:,:self.h_dim]
            t_logvar = t_params[:,:,self.h_dim:]
            v_post = reparam(v_mean, v_logvar)
            t_post = reparam(t_mean, t_logvar)
            z_post = torch.cat((v_post, t_post))
            # interpolation between original and bottlenecked
            hidden_states = (1-a) * hidden_states + a * z_post
    ...
```

Listing 1: Forward pass of `Vittle`

```
def normalized_kld(mu, logvar, modality=None):
    if modality is None:
        # vittle (F) - fixed prior N(0,I)
        kl_loss = -0.5 * (1+logvar-mu**2-logvar.exp()).mean()
    else:
        # vittle (L) - learnable prior
        mu_pr, logvar_pr = self.l_prior[modality]
        logvar_d = logvar-logvar_pr
        scaled_mu_d = (mu-mu_pr).pow(2)/logvar_pr.exp()
        var_ratio = logvar.exp()/logvar_pr.exp()
        kl_loss = -0.5 * (1+logvar_d-scaled_mu_d-var_ratio).mean()
    return kl_loss

def loss(self, logits, labels, v_mean, v_logvar, t_mean, t_logvar):
    lm_loss = self.llm.loss_function(logits, labels)
    if self.learnable_prior:
        flag_v, flag_t = "v", "t"
    else:
        flag_v, flag_t = None, None
    kld_v = self.normalized_kld(v_mean, v_logvar, flag_v)
    kld_t = self.normalized_kld(t_mean, t_logvar, flag_t)
    return lm_loss + self.beta * (kld_v + kld_t)
```

Listing 2: Training objective of `Vittle`

Figure 9: PyTorch-style pseudo code for the forward pass and training objective of `Vittle`

A.3 Downstream Task Benchmark Construction

Open-ended QA task. One of the most representative applications of an MLLM is the generation of free-form responses given multimodal instruction queries. We consider LLaVA-Bench COCO (LB-COCO; [38]) as a typical in-distribution (ID) open-ended QA dataset, which is constructed from MS-COCO images [103] with GPT-generated text queries that have 90 pairs of image and text. We then generate 27 variants of this LB-COCO by applying nine types of image perturbations, nine types of text perturbations, and nine types of image-text joint perturbations, to benchmark MLLMs' robustness under various distribution shifts (which will be elaborated at the end of this section). Meanwhile, we also consider three datasets LLaVA-Bench in-the-wild (LB-Wild; [38]), LLaVA-Bench Wilder (LB-Wilder; [104]), and WildVision-Bench (WV-Bench; [105]) constructed by real-world web users' image-text paired queries of 60, 128, and 500 samples, respectively, to validate models' capability to address long-tailed queries in practice. This results in 31 different open-ended QA datasets in total: clean and 27 perturbed LB-COCO variants, and three long-tail datasets.

Object hallucination detection task. Meanwhile, one of the most crucial evaluation aspects of an MLLM is the degree of hallucination of its output. A representative benchmark for this is the POPE dataset [77], where the model is tasked to answer in binary {Yes, No} form given a question about the object's existence given an image. The POPE dataset was also created from the MS-COCO source images with 9,000 corresponding questions, and we consider this dataset as an ID dataset. As we did for the LB-COCO dataset, we generated 9 variants of POPE by applying nine types of visual perturbations to images. Textual perturbations were not considered here because the text query of this dataset is relatively short, so perturbing the core object word token can distort the desired semantics of the question. In summary, we conducted validation on 10 different POPE variants.

Closed-form QA task. There are numerous closed-form QA datasets that assess the internal knowledge of MLLMs from various perspectives. In this paper, we consider four representative datasets: ScienceQA [78], MMMU [79], MME [5], and MMStar [80], which are designed to validate multimodal knowledge and understanding capability across various domains.

Distribution shift simulation. The goal of this study is to improve the robustness of MLLM under distribution shifts. We mainly focus on subtle perturbations on image and text, which is worth-noting problem given the fact that current MLLMs undergo systematic performance degradation under perturbations. We consider nine visual perturbations listed in Table 9, nine textual perturbations listed in Table 10, and nine image-text joint perturbations: {zoom_blur, frost, gaussian_noise} $\times$ {arabic, greek, hindi}. The translations for Arabic, Greek, and Hindi languages from English are conducted by OpenAI GPT-4o with a prompt: "Please translate a {SOURCE} sentence provided by the user into {TARGET}.", and all the remaining perturbations are generated MMRobustness source code[9]. The actual examples of each visual textual perturbation are presented in Figure 11 and Figure 10.

Table 9: **List of visual perturbations.** We consider nine visual perturbations from four categories: (1) Blur, (2) Digital, (3) Weather, and (4) Noise, to validate the robustness of MLLMs under diverse types of visual perturbations.

Name	Category
Defocus Blur	Blur
Zoom Blur	Blur
Contrast	Digital
Brightness	Weather
Fog	Weather
Frost	Weather
Gaussian Noise	Noise
Shot Noise	Noise
Speckle Noise	Noise

Table 10: **List of textual perturbations.** We consider nine textual perturbations from three categories: (1) character-level, (2) word-level, and (3) sentence-level, to validate the robustness of MLLMs under diverse types of textual perturbations.

Name	Category
Char Typo	Character-level Perturbation
Char Delete	Character-level Perturbation
Char Insert	Character-level Perturbation
Word Swap	Word-level Perturbation
Word Delete	Word-level Perturbation
Word Insert	Word-level Perturbation
Arabic Translation	Sentence-level Perturbation
Greek Translation	Sentence-level Perturbation
Hindi Translation	Sentence-level Perturbation

[9] https://github.com/Jielin-Qiu/MM_Robustness

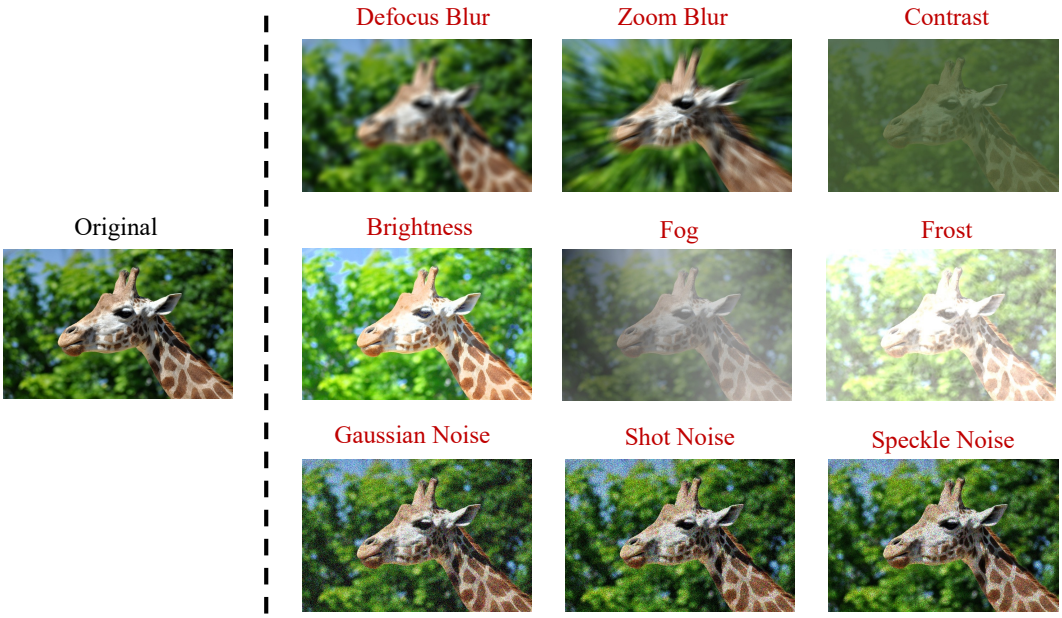


Figure 10: **Examples of visual perturbations.**

Figure 11: **Examples of textual perturbations.**

A.4 Evaluation Details

Open-ended QA task. Compared to multi-choice closed-form QA tasks that have a unique ground-truth answer per question, open-ended free-form generation-style QA tasks do not provide a single ground-truth answer. We follow the current standard evaluation paradigm, (M)LLM-as-a-Judge, that uses an external (usually more powerful) MLLM to gauge the quality of our target MLLM of interest via prompting. To be specific, for a given input query x, reference answer y, MLLM $f_\theta : \mathcal{X} \to \mathcal{Y}$, and the judge model $r : \mathcal{X} \times \mathcal{Y} \to \mathbb{Z}^+$, *relative preference* score is defined as, $\mathbb{E}_{x,y}[\frac{r(x,f_\theta(x))}{r(x,y)}]$.

For all of our open-ended QA evaluations, we used the same system prompt template provided by LLaVA authors[10], and we also adopted the MS-COCO annotation[11]-based GPT-4 response[12] and the `gpt_answer`[13] released by LLaVA-NeXT authors as reference answers for LB-COCO variants and LB-Wilder, respectively. For LB-Wild and WV-Bench, we generated reference answers with GPT-4o.

[10] https://github.com/haotian-liu/LLaVA/blob/main/llava/eval/table/rule.json

[11] https://github.com/haotian-liu/LLaVA/blob/main/llava/eval/table/caps_boxes_coco2014_val_80.jsonl

[12] https://github.com/haotian-liu/LLaVA/blob/main/playground/data/coco2014_val_qa_eval/qa90_gpt4_answer.jsonl

[13] https://huggingface.co/datasets/lmms-lab/LLaVA-Bench-Wilder

Object hallucination detection and closed-form QA task. In contrast to open-ended tasks, all object hallucination detection and closed-form QA tasks provide a single ground truth answer as a form of discrete labels such as {Yes, No} and {A, B, C, D, ...}. For the multi-choice QA datasets, MMMU, MMStar, and ScienceQA, we attached a subfix prompt: "Answer in a character from the given choices directly." at the end of each question for answer formatting, while using the original question text for YES-or-NO datasets, MME and POPE, without a formatting prompt. We measured the exact matching accuracy $\mathbb{E}_{x,y}[\mathbb{I}(_\theta(x) = y)]$ for these tasks.

Effective Mutual Information Difference (EMID) and Jensen-Shannon Divergence (JSD). In addition to the evaluation with traditional metrics, we also consider the EMID and JSD-based evaluation, which was recently proposed as an information-theoretic approach to measure the robustness of MLLMs [19]. To compute the empirical estimates of MI, which is required for EMID computation, we use the CLUB estimator [106] and reproduce the training and inference process of [19] by adopting image and text embeddings for the input image and text from CLIP-ViT-B/32 [71] and XLM-RoBERTa-Base [107] to replace X_v and X_t, and also the text embeddings of XLM-RoBERTa-Base for responses Y and Y_Θ. To compute empirical estimates of JSD, we adopted the representation JSD estimator [108] on top of the CLIP-ViT-B/32 and XLM-RoBERTa-Base, too.

Representation analysis. Inspired by a recent work that reveals the importance of intermediate layer representation of the LLM backbone [109], we use the last input token embedding of the 24th layer (out of 32 layers in a 7B LLM backbone) for all experiments carried out in the representation space (Figure 1 (b) right, Figure 8, Figure 13, and the JSD computation in Table 4).

B Additional Results

B.1 Ablation Study

We first investigate two important hyperparameters for `Vittle`: (1) bottleneck layer index l and (2) KLD regularization strength β, where we determined those parameter values based on the average performance on the clean POPE and LB-COCO, while not observing performance on perturbation datasets for fair model selection. We then further explore the impact of the interpolation coefficient α, which plays a role in controlling the balance between the original representation and the bottleneck representation. Note that we could not conduct such an extensive search due to the computational burden of training 7B 13B scale models, so the hyperparameter values found here may not be optimal, and `Vittle` can achieve better results with further hyperparameter tuning.

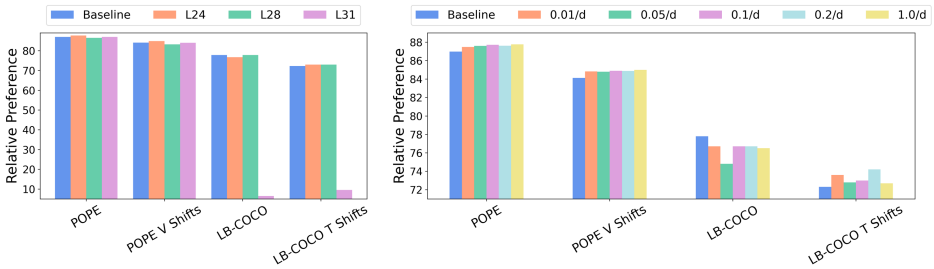


Figure 12: **Ablation study for the bottleneck layer index (left) and KLD regularization magnitude parameter β (right).**

For bottleneck target layer ablation (Figure 12 left), we swept across {8, 16, 20, 24, 28, 31} out of 32 layers of the 7B-size LLM backbone. However, applying the bottleneck on the early layer failed to make the language modeling loss converge, so we only provided results for 24, 28, and 31 layers. We observed that intermediate layers (L24 and L28) achieve better results than the penultimate layer (L31), and L24 shows better results on POPE while L28 outperforms L24 on LB-COCO. In conclusion, applying the bottleneck to too early parts hinders shaping some shallow syntactic features that will be actively used at later parts of the layers [110], whereas applying it to too late parts hurts output-specific alignment or formatting [111], which guide us to decide intermediate layer, i.e., 24th,

as a default choice. This is in line with a recent finding that the intermediate layer of LLM matters more than the early or later layers by showing that the quality measurements of the intermediate layer representations have a stronger correlation with performance in downstream tasks [109]. Although we can search different layer indices for visual and textual tokens, we leave this to future work.

For KLD regularization strength parameter ablation (Figure 12 right), we swept across $\{0.01, 0.05, 0.1, 0.2, 1.0\}$, and found that in the POPE dataset, strong regularization results in better performance, whereas it is not the case for LB-COCO. We choose $0.1/d$ as our default, which induces balanced clean-data performance on these two tasks.

Table 11: **Ablation study for the representation interpolation coefficient α of the bottleneck layer.** We observe that using the bottlenecked representation beyond the half portion of the total hinders the convergence of the language modeling loss.

Alpha	POPE	POPE V Shifts Avg.	LB-COCO	LB-COCO T Shifts Avg.
Baseline	86.98	84.12	77.8	72.3
0.1	87.22	84.20	**77.9**	**73.1**
0.25	87.34	84.47	75.6	**73.1**
0.5	**87.71**	**84.90**	76.7	73.0
0.75	Failed to converge			
1	Failed to converge			

We also explore the effect of the representation interpolation parameter $\alpha \in [0, 1]$, which can be interpreted as a gating mechanism to control the information flow. As α approaches one, the later parts of the LLM backbone (LLM head in our notation) mainly use the information-penalized representation, while if α becomes smaller, the model strongly relies on the original representation. In Table 11, we observe that using too large values of α results in diverging language modeling loss, indicating that using a strongly penalized representation only cannot predict proper response tokens in sequence. Meanwhile, the larger value of α induces better POPE performance, whereas the trend is inconsistent in the LB-COCO data set, which is consistent with the observations from the previous ablation study in β.

B.2 Full Results of Pair-wise Cosine Distance Comparison

We speculate that the performance degradation of MLLMs under perturbations originates from the representation discrepancy between clean and perturbed samples. That is, in the ideal case, a clean sample and its semantically equivalent perturbed sample should be closely mapped in the representation space, but current MLLMs did not shape the representation space in that way (see Figure 1 and Figure 8). In Figure 13, we provide the histograms of representation space pair-wise cosine distance between clean and perturbed examples in 27 types of perturbations. As we can see, `Vittle` (F) consistently mitigates the representation gap by reducing the pair-wise distance over diverse types of perturbations.

B.3 Full Results with LLaVA-v1.5-7B and LLaVA-v1.5-13B

Table 12 summarizes the overall results of our perturbation benchmarks on object hallucination detection (POPE) and open-ended QA tasks (LB-COCO). We note two findings here: (1) weight-space regularization methods, such as LoRA and WD failed to achieve reasonable performance; (2) although information maximization-based instruction tuning methods, such as ROSS and LIT, somewhat improve performance on POPE and its perturbation datasets, they greatly underperform `Vittle`, indicating a non-trivial challenge to design a versatile instruction tuning objective that can improve MLLMs on broad tasks. Meanwhile, we explore whether `Vittle` can be effective for a much larger model, e.g., a 13B-scale model. Table 13 shows that `Vittle` achieves consistent performance gains in object hallucination detection and open-ended QA tasks under distribution shifts, implying the scalability of our method.

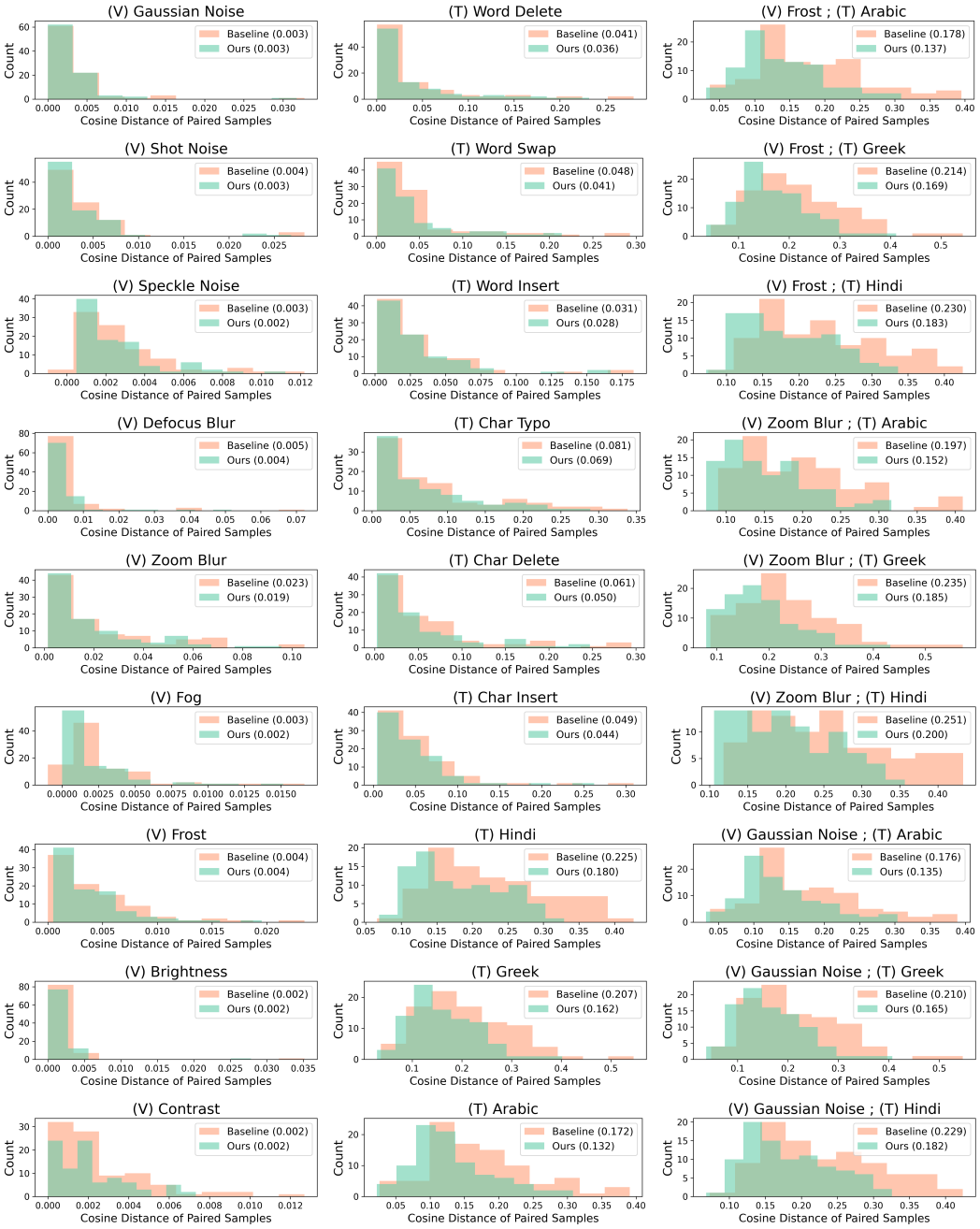


Figure 13: **Pair-wise cosine distance of intermediate representations between clean LB-COCO and 27 versions of perturbed LB-COCO datasets.** `Vittle` (F) consistently reduces the representation gap between the clean samples and their semantically equivalent perturbed ones.

Table 12: **Comparison with alternative training approaches.** We compare `Vittle` with weight-space regularization methods, LoRA [83] and weight decay (WD), and two recent visual instruction tuning learning objectives, ROSS [84] and LIT [85] on LLaVA-v1.5-7B model. Evaluations are conducted on POPE, its nine visually perturbed variants (POPE V Pert.), LB-COCO, and its nine {visually/textually/jointly} perturbed variants, where we mark the best one as bold and the second best one as underlined.

Method	POPE	POPE V Pert.	LB-COCO	LB-COCO V Pert.	LB-COCO T Pert.	LB-COCO J Pert.
Baseline	86.98	84.12	**77.8**	73.4	72.2	62.3
LoRA	83.33	80.23	73.4	70.4	62.7	39.7
WD	87.22	83.97	74.1	72.1	<u>73.0</u>	59.5
ROSS	<u>87.79</u>	84.67	74.4	72.0	71.3	60.0
LIT	87.38	84.21	<u>77.5</u>	72.1	72.9	58.9
`Vittle` (L)	87.71	<u>84.91</u>	76.7	<u>73.9</u>	<u>73.0</u>	<u>62.7</u>
`Vittle` (F)	**87.81**	**84.99**	76.1	**74.2**	**74.1**	**64.4**

Table 13: `Vittle` **on LLaVA-v1.5-13B model.** We compare `Vittle` with the standard learning objective on LLaVA-v1.5-13B model that uses Vicuna-v1.5-13B as an LLM backbone. We set the bottleneck layer index $l = 36$, interpolation coefficient $\alpha = 0.5$, and bottleneck KLD regularization strength $\beta = \frac{0.1}{d}$. `Vittle` outperforms baseline on perturbed datasets while showing rivaling performance on the clean dataset.

Method	POPE	POPE V Pert.	LB-COCO	LB-COCO V Pert.	LB-COCO T Pert.	LB-COCO J Pert.
Baseline	87.14	84.02	**76.9**	73.5	73.8	64.6
`Vittle` (L)	87.22	**84.85**	76.6	**74.5**	**74.0**	**65.4**
`Vittle` (F)	**87.32**	84.65	76.8	74.2	73.9	65.3

B.4 Further Investigation on `Vittle`

Peak memory allocation. `Vittle` hosts MLP blocks inside the LLM backbone to model the posterior distributions of the inner representations, and compute additional learning signals, e.g., KLD between priors and posteriors. In Table 5, we reported the wall-clock runtime, and here, we compare the maximum peak memory allocation across GPU chips (8 chips during the training and 1 chip during the inference) in Table 14. We see that the memory overhead induced by `Vittle` is marginal across both training and inference.

Table 14: **Memory comparison.** We compare baseline and `Vittle` (F) in terms of the maximum peak memory allocation (gigabytes; GB) across GPU chips during training and inference.

Method	Peak Mem GB (train)	Peak Mem GB (test)
Baseline	37.55	15.62
`Vittle`	38.98	15.84

Table 15: **Response diversity comparison.** We report scores of four representative measurements of text diversity on the models' responses to LB-COCO clean data queries.

Method	Distinct-1 (↑)	Distinct-2 (↑)	CR (↓)	Hom RL (↓)
Baseline	0.2356	0.6117	3.234	0.155
`Vittle` (L)	**0.2413**	**0.6279**	**3.212**	0.147
`Vittle` (F)	0.2382	0.6260	3.238	**0.144**

Output diversity comparison. One may be concerned that learning a compressive representation with `Vittle` can hurt the diversity of the textual output, which is undesirable for a chat assistant. To investigate the output text diversity with and without bottleneck training, we evaluate with four common textual diversity metrics, Distinct n-gram [112], as well as the compression ratio (CR) and

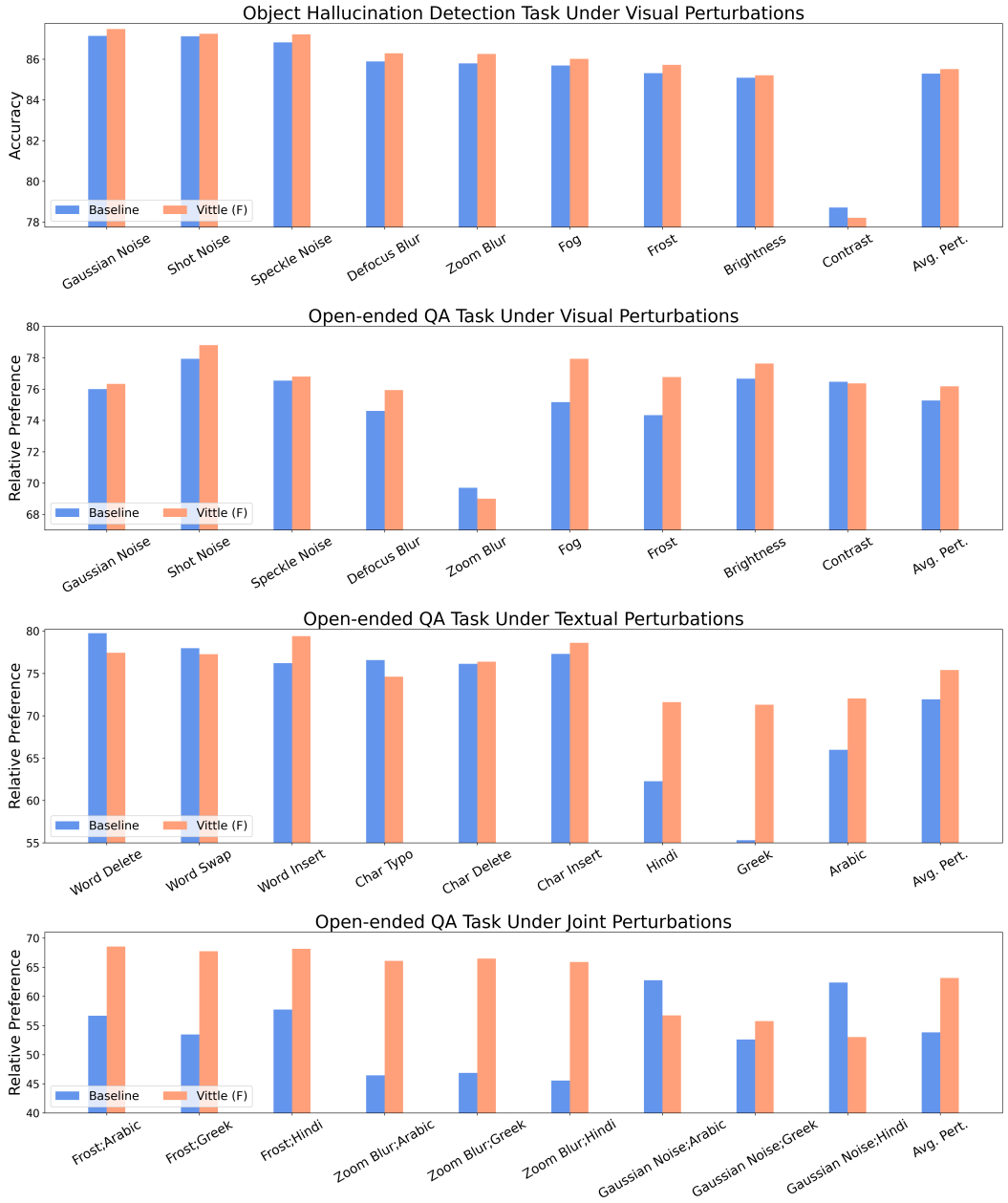


Figure 14: **Object hallucination detection performance on POPE perturbation datasets (top), and Open-ended QA performance on LB-COCO perturbation datasets (three below) of Prism-7B**. We enumerate the accuracy for the object hallucination detection task and relative preference score for the open-ended QA task of each method on perturbed datasets, where we observe consistent performance gains by `Vittle`.

the homogeneity score driven by Rouge-L (Hom RL) [113] over the outputs from each model trained by the baseline method and `Vittle` on the LB-COCO clean dataset.

B.5 Applicability to Other MLLMs

We now investigate `Vittle`'s effectiveness on another recent MLLM, Prism-7B, beyond LLaVA. As noted in Section A.1, Prism has quite a different design principle than LLaVA with respect to the visual encoder and the training strategy, so it is suitable for investigating the versatility of `Vittle`

between models. Table 16 shows summarized results on our perturbation benchmarks[14]. In object hallucination detection tasks, `Vittle` outperforms the standard cross-entropy only training baseline on clean and perturbed datasets. In open-ended QA tasks, `Vittle` consistently boosts performance in perturbation scenarios with large margins while maintaining performance on the clean dataset. The results of the perturbation-specific performance comparison are provided in Figure 14.

Table 16: `Vittle` **on Prism-7B model.** We compare `Vittle` (F) with the standard learning objective under the Prism-7B model training regime that adopts two visual encoders (DINOv2 and SigLIP) and the single-stage training rather than two-stage training with a single CLIP visual encoder. `Vittle` significantly improves perturbation-robustness compared with a naive learning objective.

Method	POPE	POPE V Pert.	LB-COCO	LB-COCO V Pert.	LB-COCO T Pert.	LB-COCO J Pert.
Baseline	87.54	85.29	79.4	75.3	71.9	53.8
`Vittle` (F)	88.11	85.52	79.0	76.2	75.4	63.2

C Extended Literature Review

Compression for generalization. There is a rich history in the machine learning field that connects compression of the model or its inner representation to generalization [114], from the classical learning theory with *Occam's razor* [115, 116] and *Minimal Description Length* [117, 118, 119] to *IB principle* [46, 47], by suggesting models that provide minimal and simplest representation of data generalize better [33, 118, 52, 120] analogy to human perception [13, 121, 15]. Recently, Wilson [122] proposed a new generalization bound for contemporary large-scale models where the ***compressibility*** of a learning algorithm plays a key role in better generalization. According to that discussion, even the maximally flexible billion-scale model can have a small effective dimensionality (indicating the higher compressibility) by embracing *soft inductive biases* [123], such as, a regularization term, to the learning problem. On top of these, IB-objective of `Vittle` can be understood as a soft inductive bias to seek a minimal sufficient representation that helps generalization for the challenging queries.

Robustness of fine-tuned foundation models. Although large-scale pre-trained models have appealing generalization capability across diverse data instances from different domains, their fine-tuned counterparts usually hurts that strong generalization capability while being adapting on task-specific in-distribution samples [124, 125]. This undesirable performance compromise between adaptation to in-distribution samples and generalization to samples from broad domains has spurred the community to work on *robust fine-tuning* of foundation models [124, 125, 126, 127, 128, 129, 130]. This line of work addresses the adaptation-robustness trade-off by (1) introducing a regularization term [128, 129], (2) tweaking the training procedure [124, 126], or (3) merging multiple models in the weight space [125, 131]. However, almost all of the existing robust fine-tuning literature has focused on a discriminative model, such as CLIP [71], under classification setups. Although there are a few works on robust instruction tuning of MLLMs [132, 133], they do not specifically focus on improving robustness under diverse types of distribution shifts and propose a *data-centric approach*, i.e., expanding instruction tuning datasets in terms of quantity or diversity, that requires external MLLM-based data generation process and/or careful post-processing from humans. In this work, we take a *representation-centric* approach that modifies the learning objective of visual instruction tuning to efficiently enhance the robustness of MLLM under diverse distribution shifts (27 types in total).

[14] Due to resource constraints, we only explore `Vittle` (F) one of our prior distribution instantiations.

D Derivation of Variational Bound for IB in MLLM

Here we provide a full derivation for the variational lower bound for IB. The derivation skeleton was mainly inspired by existing works [134, 32]. We begin with the mutual information term $I(Z, X)$. Given the sequential nature of MLLM, we decompose both the input $X = (X_v, X_t)$ and the latent representation $Z = (Z_v, Z_t)$ into **v**isual and **t**extual components. We can then derive the following upper bound for $I(Z, X)$:

$$
\begin{aligned}
I(Z,X) &= \int p(x,z)\log\frac{p(x,z)}{p(x)p(z)}dxdz = \int p(x,z)\log\frac{p(z|x)}{p(z)}dxdz \\
&= \int p(x,z)\log\frac{p(z|x)}{r(z)}dxdz - D_{\mathrm{KL}}(p(z)||r(z)) \\
&\leq \int p(x,z)\log\frac{p(z|x)}{r(z)}dxdz \\
&= \int p(x_v,x_t,z_v,z_t)\log\frac{p(z_t|x_v,x_t)p(z_v|x_v)}{r(z_v)r(z_t)}dx_v dx_t dz_v dz_t \\
&= \int p(x_v,x_t)\int p(z_t|x_v,x_t)\int p(z_v|x_v)\log\frac{p(z_v|x_v)}{r(z_v)}dx_v dx_t dz_v dz_t \\
&+ \int p(x_v,x_t)\int p(z_v|x_v)\int p(z_t|x_v,x_t)\log\frac{p(z_t|x_v,x_t)}{r(z_t)}dx_v dx_t dz_v dz_t \\
&= \mathbb{E}_{x_v}[D_{\mathrm{KL}}(p(z_v|x_v)||r(z_v))] + \mathbb{E}_{x_v,x_t}[D_{\mathrm{KL}}(p(z_t|x_v,x_t)||r(z_t))],
\end{aligned} \tag{9}
$$

where the first inequality holds given the non-negativity of $D_{\mathrm{KL}}[r(z), p(z)]$ and $p(z_v|x_v, x_t) = p(z_v|x_v)$ due to causal attention in MLLM. Here, we introduce $r(z) = r(z_v, z_t) = r(z_v)r(z_t)$ as a factorizable variational approximation of the true prior for the latent representation $p(z)$. Next, for the output-relevant term $I(Z, Y)$, we have the lower bound:

$$
\begin{aligned}
I(Z,Y) &= \int p(y,z)\log\frac{p(y,z)}{p(y)p(z)}dydz = \int p(y,z)\log\frac{p(y|z)}{p(y)}dydz \\
&= \int p(y,z)\log q(y|z)dydz + D_{\mathrm{KL}}(p(y|z)||q(y|z)) - \int p(y)\log p(y)dy \\
&\geq \int p(y,z)\log q(y|z)dydz \\
&= \int p(x,y,z)\log q(y|z)dxdydz = \int p(x)p(y|x)p(z|x)\log q(y|z)dxdydz, \\
&= \mathbb{E}_{x,y}\mathbb{E}_{z|x}\left[\log q(y|z)\right].
\end{aligned} \tag{10}
$$

where $p(x, y, z) = p(x)p(z|x)p(y|x)$ given the Markov assumption $Y \leftrightarrow X \leftrightarrow Z$, and $p(z|x, y) = p(z|x)$ holds given that the representation Z can not directly depend on Y, and the entropy term of y, i.e., $\int -p(y)\log p(y)dy = H(Y)$, is ruled out due to its independence for optimization problem. Here, we replace the intractable $p(y|z)$ with a variational approximation $q(y|z)$ that will be parameterized by a model. Finally, combining the lower bound of $I(Z, Y)$ and the upper bound of $I(Z, X)$ yields a variational lower bound for the IB objective as follows,

$$
\begin{aligned}
\mathrm{IB}(X,Y) \geq\ & \mathbb{E}_{x,y}\left[\mathbb{E}_{z|x}[\log q(y|z)]\right] \\
& - \beta\left(\mathbb{E}_{x_v}\left[D_{\mathrm{KL}}(p(z_v|x_v)||r(z_v))\right] + \mathbb{E}_{x_v,x_t}\left[D_{\mathrm{KL}}(p(z_t|x_v,x_t)||r(z_t))\right]\right).
\end{aligned} \tag{11}
$$

Asymmetric posterior $p(z|x)$ modeling in practice. As we fed the (pre/post-bottleneck) representation interpolation $\hat{z} = (1-\alpha)z + \alpha\tilde{z}$ into the LLM-head $f_{\theta^{l+}}$ for output decoding $q(y|\hat{z})$, this yields an asymmetric specification of $p(z|x)$ being modeled in $I(Z, X)$ and $I(Z, Y)$. To be precise, given our posterior instantiation $p(z|x) := \mathcal{N}(z; \mu(x), \sigma^2(x) \cdot I)$ in the upper bound of $I(Z, X)$, which is modeled by the LLM-stem f_{θ^l} plus the bottleneck layer g_ϕ, we use $\mathcal{N}(z; \alpha\mu(x) + (1-\alpha)c, \alpha^2\sigma^2(x) \cdot I)$, where $c = f_{\theta^l}(x)$ denotes a constant vector, to model the $p(z|x)$ in the lower bound of $I(Z, Y)$.

E Missing Proof

E.1 Preliminary

We start by providing a definition of Mutual Information (MI) below.

Definition E.1 (**Mutual Information (MI)**). *For a joint distribution P_{XY} over $\mathcal{X} \times \mathcal{Y}$, the mutual information with respect to P_{XY} is defined as,*

$$I(P_{XY}) := \mathbb{E}_{x,y\sim P_{XY}}[\log \frac{P_{XY}(x,y)}{P_X(x)P_Y(y)}]. \tag{12}$$

If X is an instruction and Y is a corresponding response, we regard $I(P_{XY})$ as a relevance between the instruction and the response that can be seen as a possible quantification of *instruction following* capability of MLLMs. Effective MI is defined based on the MI as follows:

Definition E.2 (**Effective Mutual Information (EMI) [19]**). *Given the joint distribution P_{XY} and MLLM P_Θ parameterized with Θ, the effective mutual information between the input and model response is defined as,*

$$\textit{EMI}(P_{XY}; P_\Theta) := I(P_{XY_\Theta}) - I(P_{XY}), \tag{13}$$

where P_{XY_Θ} denotes the joint distribution between the input X and the output of the model Y_Θ. Although the vanilla MI can also be used as a metric to evaluate models' output response by $I(P_{XY_\Theta})$, the scale of it varies depending on the target data distribution which is undesired when our interest is to compare performance of model across multiple domains which can be addressed by EMI. Recall that we are ultimately interested in the performance difference of MLLMs across two different datasets, and this can be captured by the EMI difference (EMID) as follows:

Definition E.3 (**EMID**). *Let $P_\Theta : \mathcal{X} \to \mathcal{Y}$ be an MLLM with parameters Θ that produces an output response Y_Θ given an input instruction X. For joint distributions P_{XY} and Q_{XY}, effective mutual information difference of P_Θ over P and Q is defined as below,*

$$\textit{EMID}(P_{XY}, Q_{XY}; P_\Theta) := [I(P_{XY_\Theta}) - I(P_{XY})] - [I(Q_{XY_\Theta}) - I(Q_{XY})]. \tag{14}$$

By setting P as an instruction tuning distribution (training data) and Q as an arbitrary test time distribution (evaluation data), we prefer a model that has a smaller EMID value, which indicates better robustness under distribution shifts between P and Q. Now, based on the original theorem provided by Oh et al. [19], we are ready to derive a new upper bound for EMID tailored to our representation-centric visual instruction tuning setup.

E.2 A New Upper Bound for Effective Mutual Information Difference

We first review Lemma 1 of Shui et al. [135] and its adapted version, a conditional entropy bound [19] as follows,

Lemma E.4 (Lemma 1 from Shui et al. [135]). *Let $Z \in \mathcal{Z}$ be the real-valued integrable random variable, and denoting two distributions on a common space $\mathcal{Z}$ by P and Q such that Q is absolutely continuous w.r.t. P. If for any function f and $\lambda \in \mathbb{R}$ such that $\mathbb{E}_P[\exp(\lambda(f(z) - \mathbb{E}_P(f(z))))] < \infty$, then we have:*

$$\begin{aligned}\lambda(\mathbb{E}_{z\sim Q}[f(z)] - \mathbb{E}_{z\sim P}[f(z)]) \leq\ & D_{\mathrm{KL}}(Q||P) \\ & + \log \mathbb{E}_{z\sim P}[\exp(\lambda(f(z) - \mathbb{E}_{z\sim P}[f(z)]))]\end{aligned}$$

Lemma E.5 (Conditional entropy bound [19]). *Let $f(x) := H(Q_{Y|x})$ and $\hat{H}(Q_{Y|x}) := \max_{x\in\mathcal{X}} H(Q_{Y|x})$, given the marginal distributions P_X and Q_X, and conditional distributions $P_{Y|X}$ and $Q_{Y|X}$, according to Lemma E.4, we have a conditional upper bound:*

$$i)\ \ \mathbb{E}_{x\sim P}[H(Q_{Y|x})] - \mathbb{E}_{x\sim Q}[H(Q_{Y|x})] \leq \hat{H}(Q_{Y|\mathbf{x}})\sqrt{2D_{\mathrm{JS}}(P_X||Q_X)}.$$

Similarly, given the marginal distribution P_X and Q_X, and an MLLM P_Θ, let $f(x) := H(P_\Theta(\cdot|x))$ and $\hat{H}(P_\Theta) := \max_{x\in\mathcal{X}} H(P_\Theta(\cdot|x))$, then, according to Lemma E.4, we have another conditional upper bound:

$$ii)\ \ \mathbb{E}_{x\sim Q}[H(P_\Theta(\cdot|x)] - \mathbb{E}_{x\sim P}[H(P_\Theta(\cdot|x))] \leq \hat{H}(P_\Theta)\sqrt{2D_{\mathrm{JS}}(P_X||Q_X)}.$$

Next, we should also need to formulate the relationship between JSD in the input space and JSD in the representation space, which is done through Lemma E.6.

Lemma E.6. *Let $f : \mathcal{X} \to \mathcal{Z}$ be an encoder that maps an input X to a representation Z, for the input distributions P_X and Q_X and f-induced representation distribution P_Z and Q_Z, we have an inequality below,*

$$\sqrt{2D_{\mathrm{JS}}(P_X||Q_X)} \leq \sqrt{2D_{\mathrm{JS}}(P_Z||Q_Z)} + \sqrt{\mathbb{E}_{z\sim P}[D_{\mathrm{KL}}(P_{X|z}||M_{X|z})] + \mathbb{E}_{z\sim Q}[D_{\mathrm{KL}}(Q_{X|z}||M_{X|z})]} \tag{15}$$

where $M_{X|z} := \frac{P_{X|z}+Q_{X|z}}{2}$.

Proof. We start from the definition of JSD,

$$D_{\mathrm{JS}}(P_X||Q_X) = \frac{1}{2}D_{\mathrm{KL}}(P_X||M_X) + \frac{1}{2}D_{\mathrm{KL}}(Q_X||M_X), \quad M_X = \frac{P_X+Q_X}{2}.$$

By applying the chain rule of KLD under a deterministic map[15] $X \to Z$, we know that,

$$\begin{aligned} D_{\mathrm{KL}}(P_{XZ}||M_{XZ}) &= D_{\mathrm{KL}}(P_Z||M_Z) + \int P_Z(z) D_{\mathrm{KL}}(P_{X|z}||M_{X|z})dz \\ &= D_{\mathrm{KL}}(P_X||M_X) + \cancel{\int P_X(x) D_{\mathrm{KL}}(P_{Z|x}||M_{Z|x})dx} \\ &\Leftrightarrow D_{\mathrm{KL}}(P_X||M_X) \end{aligned}$$

Then, we have,

$$D_{\mathrm{JS}}(P_X||Q_X) = D_{\mathrm{JS}}(P_Z||Q_Z) + \frac{1}{2}(\mathbb{E}_{z\sim P}[D_{\mathrm{KL}}(P_{X|z}||M_{X|z})] + \mathbb{E}_{z\sim Q}[D_{\mathrm{KL}}(Q_{X|z}||M_{X|z})]),$$

which results in ineq. (15) by applying the triangular inequality after multiplying 2 on both sides. □

Now we derive a new upper bound for EMID, which is defined over the representation space rather than the previous one defined over the input space [19] in Proposition E.7.

Proposition E.7 (EMID upper bound). *Let P_Θ be an MLLM that maps $X = \{X_v, X_t\}$ to $Z = \{Z_v, Z_t\}$, and then subsequently maps Z to Y_Θ. Given joint distributions $P_{XY} = P_X \times P_{Y|X}$ and $Q_{XY} = Q_X \times Q_{Y|X}$, by assuming consistent conditional distributions over $Z_v|Z_t$, $Z_t|Z_v$, and $Y|X$ between P and Q, we have an upper bound for EMID$(P_{XY}, Q_{XY}; P_\Theta)$ as follow,*

$$\hat{H}\big(D^{\frac{1}{2}}_{\mathrm{JS}}(P_{Z_v}||Q_{Z_v}) + D^{\frac{1}{2}}_{\mathrm{JS}}(P_{Z_t}||Q_{Z_t}) + \sqrt{\Delta_{X|Z}}\big) + |H(P_{Y_\Theta}) - H(P_Y)| + |H(Q_{Y_\Theta}) - H(Q_Y)|, \tag{16}$$

where H and $D^{\frac{1}{2}}_{\mathrm{JS}}$ indicate the entropy and square root of Jensen-Shannon divergence (JSD), respectively, $\Delta_{X|Z} := \mathbb{E}_{z\sim P}[D_{\mathrm{KL}}(P_{X|z}||M_{X|z})] + \mathbb{E}_{z\sim Q}[D_{\mathrm{KL}}(Q_{X|z}||M_{X|z})]$ with a mixture distribution $M = \frac{P+Q}{2}$, and $\hat{H} := \max_{x\in\mathcal{X}}[H(Q_{Y|x}) + H(P_{Y_\Theta})]$.

Proof. Given the entropy-based definition of the mutual information, $I(P_{XY}) := H(P_Y) - \mathbb{E}_{x\sim P}[H(P_{Y|x})]$, let $P_{Y_\Theta} = \mathbb{E}_{x\sim P}[P_\Theta(\cdot|x)]$ and $Q_{Y_\Theta} = \mathbb{E}_{x\sim Q}[P_\Theta(\cdot|x)]$, then, EMID can be expressed as follows,

$$\begin{aligned} &\mathrm{EMID}(P_{XY}, Q_{XY}; P_\Theta) \\ &= \mathrm{EMI}(P_{XY}; P_\Theta) - \mathrm{EMI}(Q_{XY}; P_\Theta) \\ &= (H(P_{Y_\Theta}) - \mathbb{E}_{x\sim P}[H(P_\Theta(\cdot|x))] - H(P_Y) + H(P_{Y|X})) \\ &- (H(Q_{Y_\Theta}) - \mathbb{E}_{x\sim Q}[H(P_\Theta(\cdot|x))] - H(Q_Y) + H(Q_{Y|X})) \\ &\leq (H(P_{Y|X}) - H(Q_{Y|X})) + \big(\mathbb{E}_{x\sim Q}[H(P_\Theta(\cdot|x))] - \mathbb{E}_{x\sim P}[H(P_\Theta(\cdot|x))]\big) \\ &+ |H(P_{Y_\Theta}) - H(P_Y) + H(Q_Y) - H(Q_{Y_\Theta})| \\ &\leq \underbrace{(H(P_{Y|X}) - H(Q_{Y|X}))}_{\text{(A)}} + \underbrace{\big(\mathbb{E}_{x\sim Q}[H(P_\Theta(\cdot|x))] - \mathbb{E}_{x\sim P}[H(P_\Theta(\cdot|x))]\big)}_{\text{(B)}} \\ &+ |H(P_{Y_\Theta}) - H(P_Y)| + |H(Q_Y) - H(Q_{Y_\Theta})|. \end{aligned} \tag{17}$$

[15] Note that although `Vittle` governs a probabilistic encoder during training, it produces deterministic output during inference by using the learned posterior mean rather than a random sample (Section 3.3).

Moreover, we have the following inequality for $H(P_{Y|X})$ proposed by [19],

$$H(P_{Y|X}) - H(Q_{Y|X}) \leq 4\mathbb{E}_{x\sim P}[D_{\mathrm{JS}}^{\frac{1}{4}}(P_{Y|x}||Q_{Y|x})] + \mathbb{E}_{x\sim P}[H(Q_{Y|x})] + \mathbb{E}_{x\sim Q}[H(Q_{Y|x})] \tag{18}$$

By plugging inequalities in Lemma E.5 and ineq. (18) into the ineq. (17) to replace the terms (A) and (B), and given the consistent conditional distribution assumption for $Y|X$, i.e., $P_{Y|X} = Q_{Y|X}$, we have a much simpler upper bound as follows,

$$\text{EMID}(P_{XY}, Q_{XY}; P_\Theta) \leq \hat{H}\sqrt{2D_{\mathrm{JS}}(P_X||Q_X)} + |H(P_{Y_\Theta}) - H(P_Y)| + |H(Q_Y) - H(Q_{Y_\Theta})|,$$

where $\hat{H} := \max_{x\in\mathcal{X}}[H(Q_{Y|x}) + H(P_{Y_\Theta})]$. Then, we can further replace the term $D_{\mathrm{JS}}(P_X||Q_X)$ by using Lemma E.6 to get a bound defined by representation divergence as below,

$$\begin{aligned}
&\text{EMID}(P_{XY}, Q_{XY}; P_\Theta) \\
&\leq \hat{H}(\sqrt{2D_{\mathrm{JS}}(P_Z||Q_Z)} + \sqrt{\mathbb{E}_{z\sim P}[D_{\mathrm{KL}}(P_{X|z}||M_{X|z})] + \mathbb{E}_{z\sim Q}[D_{\mathrm{KL}}(Q_{X|z}||M_{X|z})]}) \\
&+ |H(P_{Y_\Theta}) - H(P_Y)| + |H(Q_Y) - H(Q_{Y_\Theta})|.
\end{aligned} \tag{19}$$

Meanwhile, the chain rule of KLD and the definition of JSD with our consistency assumption for conditional distributions for $Z_v|Z_t$ and $Z_t|Z_v$, one can easily show below.

$$\begin{aligned}
2D_{\mathrm{JS}}(P_{Z_v Z_t}||Q_{Z_v Z_t}) &= D_{\mathrm{KL}}(P_{Z_v Z_t}||M_{Z_v Z_t}) + D_{\mathrm{KL}}(Q_{Z_v Z_t}||M_{Z_v Z_t}) \\
&= D_{\mathrm{JS}}(P_{Z_v}||Q_{Z_v}) + D_{\mathrm{JS}}(P_{Z_t}||Q_{Z_t})
\end{aligned} \tag{20}$$

Plugging Eq. 20 into ineq. (19) and applying the triangular inequality complete the proof. □

F Impact Statement

Multimodal large language models (MLLMs) today have many societal applications. This work tackles the robustness of MLLMs to distribution shifts between training and test time data. We observed a consistent improvement of our proposal `Vittle` in various types of visual and textual shifts, allowing users to trust the model more than before to safely use AI in a variety of environments. Moreover, although we focused on the robustness perspective in this work, improved invariance-sensitivity trade-off also benefits the fairness-discriminativeness trade-off, which is another crucial desideratum towards reliable AI. Meanwhile, even though its robustness to distribution shifts was improved, there are still potential misuse cases with MLLMs that can affect humanity by producing systematically biased outputs, given the existence of some adversarial data providers or attackers.