Predictive Regularization Against Visual Representation Degradation in Multimodal Large Language Models

Supplementary Material

7. Implementation Details

Training and evaluation configuration. Our training configuration simply follows llava-1.5 [43] and llava-next [44]. Specificall, in Stage 1, we fine-tune the projector using a learning rate of 1e-3. The total batch size is 256, achieved by distributing a per-device batch size of 16 across 8 devices with a gradient accumulation step of 2. For Stage 2, two configurations are explored: if the vision encoder is frozen, the projector and LLM are fine-tuned with a learning rate of 2e-5; alternatively, if the entire model is fine-tuned, the vision encoder's learning rate is set to 2e-6, while the remaining model components (projector and LLM) use 1e-5. Stage 2 utilizes a total batch size of 128 (per-device batch size of 4 across 8 devices, with a gradient accumulation step of 4). Common to both stages are the use of DeepSpeed-ZeRO-3 for memory optimization, a cosine learning rate scheduler, AdamW optimizer, and weight decay and warmup ratio set to 0 and 0.03, respectively. Each stage is trained for a single epoch. For PRe parameters, we set $\lambda = 0.5$ as the default and select the middle layer of LLM as the target regularization layer, the ablation experiments are discussed in the Sec. 5 of the main paper. We evaluate most datasets using lmms-eval, except for evaluating MMVP using the standard scripts provided by Cambrian-1. A simple training algorithm of our PRe is shown in Algorithm 1

Implementation details of analysis experiments. In Sec. 4.1, we conduct extensive diagnostic experiments to point out the problem of visual representation degradation. For the *Global Functional Degradation* analysis (Figure 2), we use the publicly available, pre-trained weights of several popular MLLMs from huggingface: Qwen-2.5-VL-7B-Instruct, LLaVA-1.5-7B, LLaVA-1.6-7B, and InternVL-3.5-4B. This allowed us to demonstrate the generality of the degradation phenomenon across different models. For the diagnostic analyses in *Patch Structure Degradation* (Fig. 3 and Fig. 4) and *Degradation as a Trade-off for Language Capability* (Fig. 5 and Fig. 6), we trained a baseline model employs the Qwen-2.5-7B-Instruct as LLM paired with a CLIP-ViT-L/14@336 vision encoder. The vision encoder was kept frozen during the initial pre-training stage but was made trainable during the visual instruction tuning stage. All analyses in these paragraphs were conducted on this consistently trained baseline to ensure a controlled experimental environment.

Details of 'patch structure degradation' paragraph.

Algorithm 1 PRe Training Step

Input: Image batch $\mathbf{I}$, Text batch $\mathbf{T}$, MLLM M, PRe head f_{pred}.
Hyperparameters: Target layer l_{target}, weight λ.
1. Get initial & intermediate visual representations
$\mathbf{H}_v^0 \leftarrow M.\text{encode\_image}(\mathbf{I})$ ▷ Features after projection
$\mathbf{H}_t^0 \leftarrow M.\text{tokenizer}(\mathbf{T})$ ▷ Get text tokens
$\mathbf{H}^l \leftarrow M.\text{forward\_llm}(\mathbf{H}_v^0, \mathbf{H}_t^0)$ ▷ Get all layer hidden states
$\mathbf{H}_v^{l_{target}} \leftarrow \text{GetVisualTokens}(\mathbf{H}^{l_{target}})$ ▷ Visual tokens from the target layer
2. Define anchor (target) and online (prediction)
$\mathbf{z}_{\text{anchor}} \leftarrow \text{stop\_gradient}(\mathbf{H}_v^0)$
$\mathbf{p}_{\text{online}} \leftarrow f_{pred}(\mathbf{H}_v^{l_{target}})$
3. Compute losses
$\mathcal{L}_{\text{LM}} \leftarrow \text{LanguageModelLoss}$
$\mathcal{L}_{\text{PRe}} \leftarrow -\text{CosineSimilarity}(\mathbf{p}_{\text{online}}, \mathbf{z}_{\text{anchor}})$
$\mathcal{L}_{\text{total}} \leftarrow \mathcal{L}_{\text{LM}} + \lambda \mathcal{L}_{\text{PRe}}$
4. Update model parameters
$\mathcal{L}_{\text{total}}.\text{backward}()$
optimizer.step()

To understand the microscopic mechanism behind the global functional degradation, we analyze the semantic structure at the patch level. Our analysis hinges on quantifying the separability of patch representations by leveraging ground-truth segmentation masks from the COCO-Stuff dataset [8]. For each image and for the sequence of patch representations $\mathbf{H}_v^l$ at a given layer l, we compute the following metrics:

First, we measure the Intra-Object Cohesion, which quantifies how tightly clustered the representations of patches belonging to the same object instance are. It is defined as the average cosine similarity between all unique pairs of patches within the same object class c:

$$\text{Cohesion}(\mathbf{H}_v^l) = \mathbb{E}_{c \in \mathcal{C}} \left[\mathbb{E}_{i,j \in \mathcal{S}_c, i \neq j} [\cos(\mathbf{h}_{v,i}^l, \mathbf{h}_{v,j}^l)] \right] \quad (3)$$

where $\mathcal{C}$ is the set of all object classes in the image, cos is cosine similarity distance, and $\mathcal{S}_c$ is the set of patch indices belonging to class c. A higher cohesion value indicates that the model groups patches of the same object together more effectively.

Second, we measure the Inter-Object Coupling, which quantifies the degree of confusion or similarity between different objects. It is defined as the average cosine similarity between pairs of patches belonging to different object

Table 4. **The effectiveness of the predictive regularization on a 3B-LLM**. * indicates the visual encoder is frozen during training, without * indicates the visual encoder is trainable.

Visual Encoder	$\mathcal{L}_{\text{PRe}}$	General & Knowledge					OCR-related		Vision-centric	
		GQA	MMMU	AI2D	MMStar	SQA^{I}	VQA^{T}	OCRbench	RWQA	MMVP
LLM: Qwen2.5-3B-Instruct										
CLIP-ViT-L/14@336 *	–	61.3	40.7	62.2	41.3	71.9	43.3	318	52.5	31.3
	✓	61.3 +0.0	40.9 +0.2	62.5 +0.3	41.1 -0.2	72.5 +0.6	43.4 +0.1	320 +2	55.3 +2.8	30.7 -0.6
CLIP-ViT-L/14@336	–	61.0	39.6	62.2	42.6	72.7	45.5	359	55.6	38.0
	✓	60.9 -0.1	40.0 +0.4	62.4 +0.2	43.2 +0.6	72.9 +0.2	45.9 +0.4	362 +3	55.8 +0.2	38.7 +0.7

classes c and c':

$$\text{Coupling}(\mathbf{H}_v^l) = \mathbb{E}_{c,c' \in \mathcal{C}, c \neq c'} \left[\mathbb{E}_{i \in \mathcal{S}_c, k \in \mathcal{S}_{c'}} [\cos(\mathbf{h}_{v,i}^l, \mathbf{h}_{v,k}^l)] \right] \tag{4}$$

A lower coupling value is desirable, as it signifies better separation between different objects.

Finally, we define the Semantic Contrast Ratio as the ratio of these two metrics. It serves as a holistic measure of patch-level separability, where a higher value is better:

$$\text{Contrast}(\mathbf{H}_v^l) = \frac{\text{Cohesion}(\mathbf{H}_v^l)}{\text{Coupling}(\mathbf{H}_v^l)} \tag{5}$$

These metrics allow us to quantitatively track how the semantic boundaries between and within objects evolve across the layers of the MLLM.

Details of 'degradation as a visual sacrifice for language capability' paragraph. In Fig. 5 (left), we quantify the Geometric Complexity using the PCA effective dimension. We compute the covariance matrix of the centered features and perform eigenvalue decomposition. The effective dimension is the minimum number of principal components k required to explain 95% of the total variance:

$$\text{EffectiveDim}(\mathbf{X}) = \min \left\{ k \mid \frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{D} \lambda_i} \geq 0.95 \right\} \tag{6}$$

where λ_i are the eigenvalues sorted in descending order. A higher effective dimension indicates a more complex and higher-capacity representation.

In Fig. 5 (right), we measure the Linear Feature Redundancy by calculating the mean absolute off-diagonal correlation. This metric quantifies the degree of statistical dependence between feature dimensions. We compute the Pearson correlation matrix $\mathbf{C} \in \mathbb{R}^{D \times D}$ of the features, and the metric is then the average of the absolute values of its off-diagonal elements:

$$\text{Redundancy}(\mathbf{X}) = \frac{1}{D(D-1)} \sum_{i \neq j} |\mathbf{C}_{ij}| \tag{7}$$

where a lower value indicates better statistical independence (*i.e.*, less redundancy).

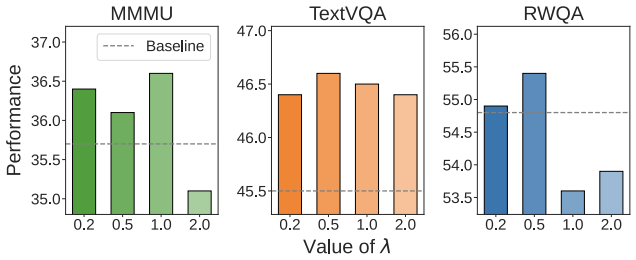

Figure 12. **The impact of λ.** The baseline architecture employed is CLIP-ViT-L/14@336 (frozen) and Vicuna-7B-v1.5.

8. More Experiments

Results on 3B LLMs. In Tab. 1 of the main paper, we demonstrated the effectiveness of our method across a wide range of 7B-LLM-based architectures. Here, in Tab. 4, we further supplement these findings with experimental results on 3B LLMs. Compared to the baseline, employing PRe yields performance improvements across multiple datasets. These results further underscore the importance of maintaining robust internal visual representations during MLLM's language-driven next-token prediction training, and prove the universally effectiveness of our PRe.

Impact of the Regularization Weight λ. In Fig. 11 of the main paper, we show the impact of hyperparameter λ set to 0.5, 1.0, and 2.0. Here, in Fig. 12, we provide more results. We can observe that when λ is too small (*e.g.*, 0.2), the regularization is too weak to effectively counteract the representation degradation, resulting in only marginal gains over the baseline. As we increase λ, the performance improves, however, as we further increase λ to larger values (*e.g.*, 2.0), the performance begins to decline. This suggests that a moderate regularization strength is optimal, providing a sufficient signal to preserve visual fidelity without unduly interfering with the language modeling task. Based on this analysis, we set $\lambda = 0.5$ as our default value for all main experiments, as it offers the best trade-off between preserving visual fidelity and maintaining language capability.

Results using different anchor features. In Tab. 3 of the main paper, we present results utilizing various sources

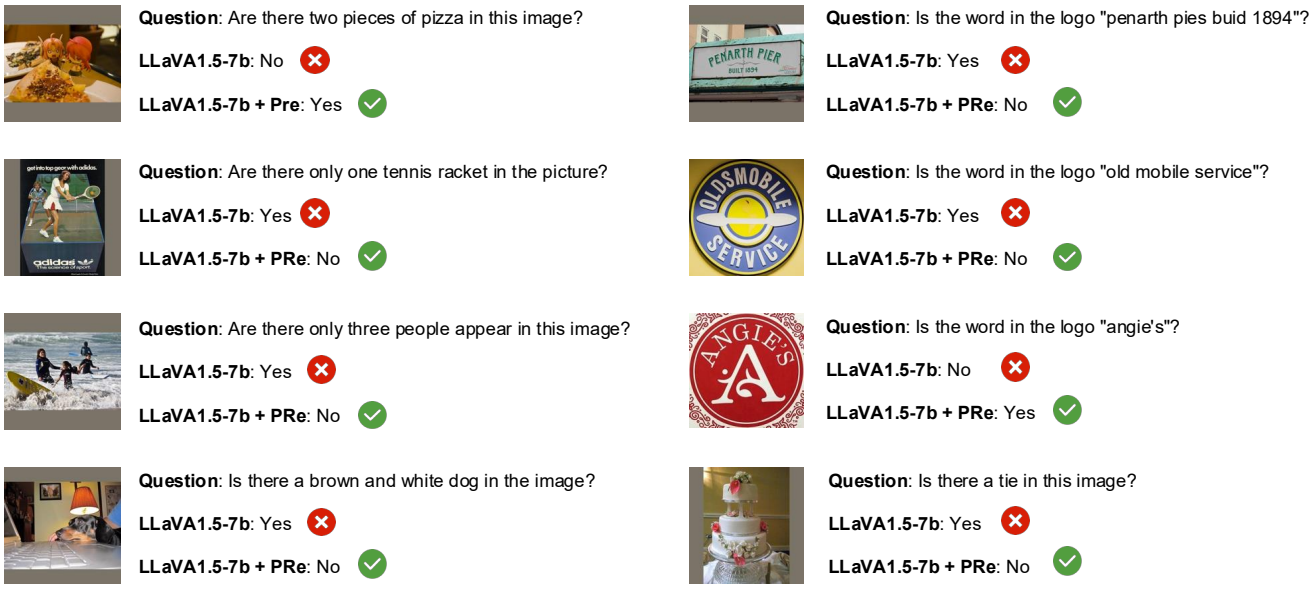

Figure 13. **Case studies comparing LLaVA-1.5-7B with and without our PRe method.**

Table 5. **Results using different anchor features.** When the CLIP is frozen, the 'Pre-Proj' setting corresponds to using the penultimate layer features from a pre-trained CLIP model, together with DINOv2, DINOv3, and SAM, which form an ablation using vision foundation models

Anchor Source	GQA	MMMU	VQA^T	RWQA	MMVP
CLIP (frozen) + Vicuna-7B-v1.5					
Baseline	62.0	35.7	45.5	54.8	20.0
Pre-LLM	62.7	**36.1**	**46.6**	**55.4**	22.0
Pre-Proj	62.7	35.1	46.4	54.4	**32.7**
DINOv2-vitb14-reg4	**62.8**	35.9	46.5	54.6	28.7
DINOv3-vitb16	62.6	35.3	46.1	54.9	28.7
SAM-vitb	62.6	34.2	**46.6**	54.4	26.0

Table 6. **Computational overhead.**

Metric	clip+qwen2.5-3b	+ PRe	Overhead (Δ)
Total Training PFLOPs	2.214	2.215	+0.045%
Throughput (samples/s)	18.74	18.58	−0.85%
Total Training Time (s)	35.5×10^3	35.8×10^3	+0.85%

Table 7. **Results on more datasets.**

	mme^p	mmb^{en}	mmb^{cn}	$seed^i$	pope(acc/f1)
clip+qwen2.5-3b	1408.3	70.1	66.1	69.9	87.5/86.3
+Pre	1434.8	68.3	68.1	70.8	88.7/87.8
clip*+vicuna-7b	1510.7	64.3	46.4	58.6	84.7/85.8
+Pre	1528.3	65.7	56.2	66.3	86.9/85.8

of anchor features. Here, in Tab. 5, we further supplement these findings with results on DINOv3 and SAM. Our observations indicate that employing stronger visual foundation models, such as DINOv3, does not necessarily lead to superior performance improvements. This is likely because the feature spaces of these pure visual foundation models are optimized solely for visual tasks. Forcing the MLLM's internal multimodal feature space to predict such a pure visual space might introduce conflicting optimization objectives. Consequently, while these approaches offer a slight improvement over the baseline, their performance does not surpass our default Pre-LLM strategy. This is because Pre-LLM's features, being projections from the projector, strike a better balance between alignment with the LLM's input space and retaining robust visual capabilities.

Case studies. In Fig. 13, we present several case studies demonstrating that the application of PRe leads to notable improvements on questions related to counting, OCR, color perception, and object existence. This underscores the importance of maintaining robust internal visual features within the MLLM and validates the effectiveness of our method.

Computational overhead. As shown in Tab. 6, the additional training cost is negligible, and there is zero overhead during inference since the PRe module is discarded after training.

More results. The main paper covers 9 diverse and general benchmarks spanning General Knowledge, OCR-related, and Vision-centric tasks. In Tab. 7, we provide additional results on 5 widely-used VQA datasets. These consistent gains across total 14 benchmarks confirm the effectiveness of PRe.

Generalizability to stronger encoders and higher resolutions. We have demonstrated the robustness of PRe across different **vision encoders** (CLIP, SigLIP2), **LLM backbones** (Vicuna, Qwen), **LLM scales** (7b, 3b), and

Table 8. **Results on stronger encoders and higher resolutions.** LLM: Qwen2.5-3b-Instruct.

	GQA	AI2D	MMStar	VQA^T	RWQA
CLIP (Res: 336^2)	**61.3**	62.2	**41.3**	43.3	52.5
+Pre	**61.3**	**62.5**	41.1	**43.4**	**55.3**
CLIP (Res: 672^2)	60.6	62.1	42.6	45.1	52.7
+Pre	**61.0**	**62.6**	**44.0**	**45.9**	**56.7**
Qwen2.5-VL(NaViT)	59.4	60.7	42.5	50.0	50.5
+Pre	**60.1**	**62.2**	**43.0**	**53.2**	**51.9**
SigLIP2-NaFlex (NaViT)	60.8	**63.1**	43.7	42.7	51.4
+Pre	**61.**4	62.9	**44.7**	**46.4**	**53.3**

training paradigms (Frozen, Unfrozen). In Tab. 8, we further validate PRe across **resolutions**, and **stronger encoders**. The consistent gains across all these settings prove the generalizability of PRe to stronger, modern baselines.

9. Limitations and Future Work.

In this work, we first identify the issue of visual degradation within MLLMs and subsequently propose Predictive Regularization (PRe) to maintain robust internal visual representations, thereby enhancing performance on vision-language tasks. Our approach is directly inspired by classic and influential works in visual representation learning, such as SimSiam and JEPA. However, a potential limitation of our current PRe strategy is that using internal representations to predict the initial visual representations might inherit biases inherent to the original visual encoder. Furthermore, the broader integration of various visual representation learning methods with MLLM pre-training remains largely unexplored. For instance, the potential of contrastive learning approaches, beyond our current predictive coding-inspired method, is yet to be fully investigated within this context. In future work, we aim to further strengthen the link between advanced visual representation learning techniques and MLLM pre-training. Our goal is to continuously optimize the visual representations within MLLMs and leverage a wider spectrum of self-supervised learning paradigms to foster even more robust and versatile multimodal models.