UMM-JEPA: GENERATIVE PRETRAINING VIA MASKED NEXT-SCALE REPRESENTATION PREDICTION FOR UNIFIED MULTIMODAL MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Unified multimodal models can support both visual understanding and generation, yet generative training does not necessarily improve understanding. We argue that this limitation may arise from a mismatch between the representation targeted by generation and the perceptual representation used for understanding. Conventional generation objectives primarily optimize synthesis-oriented spaces such as discrete visual codes or generative latents. We therefore introduce UMM-JEPA, which reformulates generation as progressive prediction of continuous visual-understanding representations. The model constructs a $64 \to 128 \to 256 \to 512$ representation hierarchy from coarse to fine, while pixel rendering is decoupled from representation pretraining and attached only afterward. Controlled comparisons show that the representation-prediction objective accounts for most of the understanding improvement, while a data-matched control leaves the group means nearly unchanged. The final UMM-JEPA reaches a 16-benchmark visual mean of 51.71, against 46.36 for the understanding-only Qwen-VL baseline, and the text mean also rises. Compared with discrete-code and latent-flow generation objectives, perceptual representation prediction yields stronger and more consistent transfer to understanding. Linear probes and visual-evidence interventions further reveal richer object and spatial structure and stronger use of image-specific information. These results suggest that generation can benefit understanding when it directly supervises the perceptual representations that understanding consumes.

1 INTRODUCTION

Unified multimodal large language models (MLLMs) increasingly support both visual understanding and generation within a single system (Team, 2024; Wang et al., 2024; Deng et al., 2025; Li et al., 2025; Xie et al., 2025). Coexistence, however, does not imply positive knowledge transfer: a model may learn to understand and generate images without generative training actually improving the representations used for understanding. Recent controlled studies reveal a notable asymmetry in this knowledge flow: visual-understanding supervision can substantially benefit visual generation, whereas simply increasing visual-generation training does not consistently improve visual understanding (Han et al., 2026). This raises the central question of this paper: *how should generation be formulated if it is expected to improve understanding?*

We hypothesize that this asymmetry originates from a mismatch between the representation targeted by generation and the representation consumed by understanding. Conventional unified models typically formulate generation in synthesis-oriented spaces, such as discrete visual codes, generative VAE latents, or pixel-oriented objectives. Visual understanding, in contrast, relies on perceptual representations, semantically organized visual states produced by a vision encoder and consumed by the language model. Under conventional generation objectives, generative supervision may update shared model parameters without directly shaping the perceptual state that understanding later reads. The key issue may therefore be not whether generation is introduced, but *what representation the generative objective forces the model to construct*.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

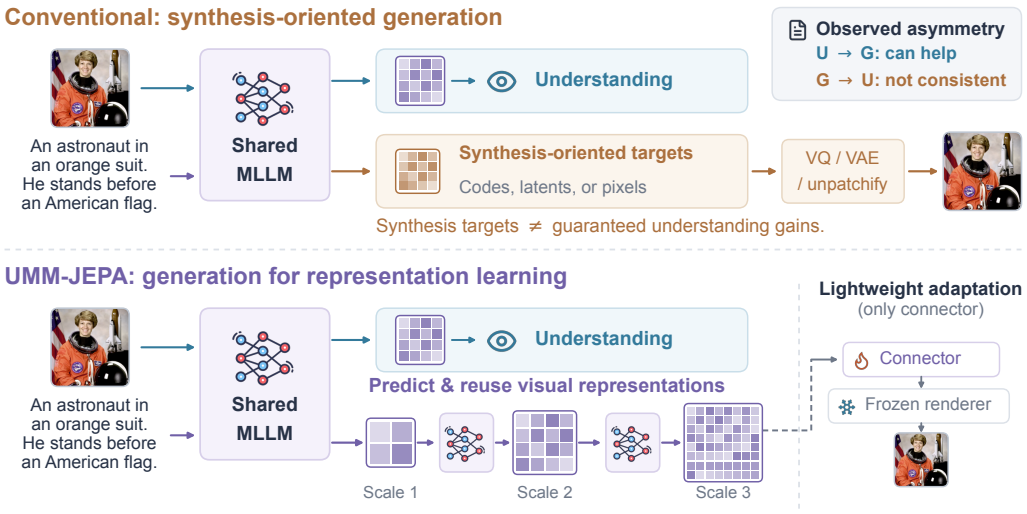


Figure 1: **Reformulating generation as representation learning.** Conventional unified models optimize synthesis-oriented targets, whereas UMM-JEPA directly predicts continuous visual-understanding representations through progressive coarse-to-fine construction. Pixel rendering is decoupled from core representation pretraining.

This perspective motivates us to reformulate generation as *progressive visual representation construction*. Instead of treating perceptual features as intermediate conditioning for image synthesis, we make prediction of those perceptual representations the generation objective itself.

We instantiate this formulation as UMM-JEPA, a unified multimodal model whose understanding and representation-generation paths share a Qwen3 LLM and a SigLIP2 visual front end (Team, 2025a; Tschannen et al., 2025). Instead of predicting pixels, discrete visual codes, or dedicated generative latents, UMM-JEPA predicts continuous SigLIP2 representations along a $64 \rightarrow 128 \rightarrow 256 \rightarrow 512$ hierarchy. During training, the coarse-to-fine conditional mappings are learned under teacher-forced visual context; during inference, predicted coarse states are recursively reused through the shared backbone to construct finer states. The formulation follows the predictive-representation principle of JEPA (Assran et al., 2023), but extends it to conditional multimodal generation. Pixel rendering is deliberately removed from core representation pretraining and attached only afterward through a lightweight connector and a pretrained diffusion renderer.

We evaluate this hypothesis through a controlled evidence chain. An understanding-only Qwen-VL baseline establishes whether generative representation learning provides any benefit, while a data-matched control separates the effect of the objective from that of additional image–text data. Generation-paradigm controls further compare continuous perceptual prediction against discrete visual-code prediction (UMM-VQ) and latent-flow generation (Show-o2), and component ablations examine bidirectional visual attention, masking, and auxiliary reconstruction. Finally, linear probes and visual-evidence interventions analyze how generative representation learning changes the learned visual states and their use during multimodal understanding. Our contributions are:

- We identify the representation targeted by generation as a key factor governing whether generative pretraining transfers to visual understanding, highlighting the importance of supervision that directly shapes the perceptual representation space used by understanding.
- We introduce UMM-JEPA, which reformulates visual generation as progressive next-scale prediction of continuous visual-understanding representations while decoupling pixel rendering from core representation pretraining.
- We provide controlled empirical evidence separating the effects of additional data, attention design, and generation paradigm, showing that perceptual representation prediction transfers more effectively to understanding than the compared synthesis-oriented objectives.
- We provide mechanistic evidence that generative representation learning changes both the structure of visual states and how visual information is used during reasoning, through linear probes and visual-evidence interventions.

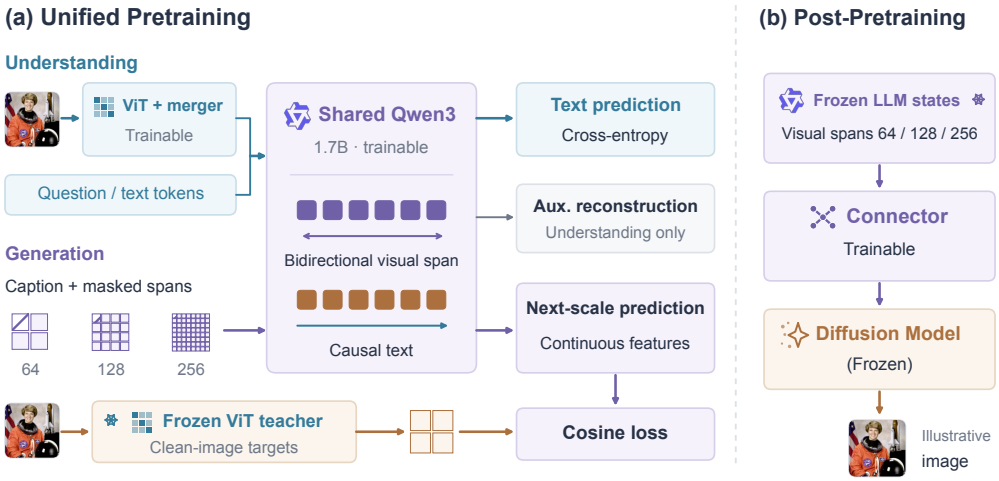


Figure 2: **Overview of UMM-JEPA.** Understanding and representation generation share a single multimodal backbone. Understanding uses language-modeling supervision, while generation progressively predicts frozen perceptual representations from coarse to fine. Pixel rendering is decoupled from representation pretraining and attached only afterward through a separately trained connector and pretrained diffusion renderer.

2 Related Work

Unified multimodal models adopt markedly different generative parameterizations, including autoregressive discrete visual tokens (Team, 2024; Wang et al., 2024; Wu et al., 2024), next-scale visual autoregression (Tian et al., 2024), latent flow matching (Xie et al., 2025; Chen et al., 2025b), and modality-specialized, mixture-of-transformer, or partially separated pathways (Liang et al., 2025; Li et al., 2025; Deng et al., 2025). Controlled studies report an asymmetric knowledge flow, understanding supervision benefits generation, whereas extra generation training does not consistently improve understanding and gains depend on the task and reasoning regime (Han et al., 2026; Wen et al., 2026; Wu et al., 2026), and auxiliary generative targets such as depth or segmentation can benefit perception (Su et al., 2026). Closest to us, semantically meaningful representations have been placed inside the generative pathway via continuous visual embeddings (Sun et al., 2023b;a), diffusion in a CLIP feature space (Chen et al., 2025a), or representation prediction as intermediate generative states (Wang et al., 2026), under the principle of predictive learning in representation space (Assran et al., 2023). We instead make prediction of the visual-understanding representation itself the pretraining objective and remove pixel rendering from core pretraining, isolating whether transfer strengthens when generative supervision shapes the representation that understanding consumes; Appendix A gives a detailed discussion.

3 Method

UMM-JEPA reformulates visual generation as progressive prediction of the continuous perceptual representations used by multimodal understanding. Understanding and representation generation share the same multimodal backbone, while a frozen visual encoder provides the prediction targets.

During pretraining, understanding samples use language-modeling supervision, whereas generation samples learn masked coarse-to-fine representation prediction. Pixel rendering is deliberately excluded from representation pretraining and attached only afterward through a lightweight connector and a pretrained diffusion renderer. Figure 2 summarizes the framework.

3.1 Design Motivation for Representation Prediction

Perceptual representation as the prediction target. Let $\mathbf{z} = E(\mathbf{x})$ denote the perceptual representation produced by the visual encoder and consumed by the multimodal understanding model h. Our central question is which generation target should be optimized if generative supervision is expected to improve the representation used by understanding.

Proposition 1 (Target-aligned representation prediction)**.** *Assume that the downstream understanding mapping h is locally L-Lipschitz in the perceptual representation space. Then*

$$\|h(\hat{\mathbf{z}}) - h(\mathbf{z})\| \leq L\|\hat{\mathbf{z}} - \mathbf{z}\|. \tag{1}$$

Thus, prediction error measured directly in the representation consumed by understanding provides direct control over the induced downstream deviation.

Design implication. We therefore use frozen perceptual representations from the visual encoder as the generation target, rather than supervising generation only through a separate synthesis-oriented space.

Lemma 1 (Conditioning on coarse states cannot increase optimal prediction risk)**.** *Let $\mathbf{Z}^{k+1}$ denote a finer-scale representation and $\mathcal{C}_k = (C, \mathbf{Z}^{64}, \ldots, \mathbf{Z}^{k})$ the text condition together with available coarser states. Under squared prediction loss,*

$$\mathcal{R}^*(\mathbf{Z}^{k+1} \mid \mathcal{C}_k) \leq \mathcal{R}^*(\mathbf{Z}^{k+1} \mid C). \tag{2}$$

Design implication. We therefore factorize representation generation progressively, conditioning each finer scale on the previously available coarse states.

Proposition 2 (Masked prediction induces conditional recovery)**.** *Let M denote masked visual states and $\bar{M}$ the visible states. Under squared prediction loss, the Bayes-optimal predictor is*

$$f^*(C, \mathbf{Z}_{\bar{M}}) = \mathbb{E}[\mathbf{Z}_M \mid C, \mathbf{Z}_{\bar{M}}]. \tag{3}$$

Masking therefore removes direct access to the target and requires recovery from textual, spatial, and cross-scale context.

Design implication. We randomly mask visual states during training to prevent trivial direct observation of prediction targets. We additionally include fully masked samples so that the text-to-visual transition required at the start of inference is explicitly represented during training.

Relation to the implemented objective. The analysis above uses squared loss for clarity. Because the target representations are ℓ_2-normalized, squared Euclidean distance is equivalent to cosine distance up to a constant. We therefore optimize cosine representation loss in practice. Formal proofs and assumptions are provided in Appendix D.

3.2 Progressive Masked Representation Prediction

Representation targets. Let $\mathbf{z}^k$ denote the frozen SigLIP2 (layer 23) representation of the ground-truth image at scale k, where $k \in \{64, 128, 256, 512\}$, corresponding to $4/16/64/256$ visual tokens. The teacher encoder is frozen and applied independently to clean target images.

Progressive factorization. Generation samples follow the coarse-to-fine chain

$$p(\mathbf{z}^{64} \mid c), \qquad p(\mathbf{z}^{128} \mid c, \mathbf{z}^{64}), \qquad p(\mathbf{z}^{256} \mid c, \mathbf{z}^{64}, \mathbf{z}^{128}), \qquad p(\mathbf{z}^{512} \mid c, \mathbf{z}^{64}, \mathbf{z}^{128}, \mathbf{z}^{256}), \tag{4}$$

where c is the conditioning context (the language caption and any visible visual states under masking). Each parent predicts its four native children at the next scale, supervised by a cosine loss against the frozen teacher, with the four scales equally weighted and normalized.

Training: teacher-forced context. Training is teacher-forced. The 64/128/256 input spans are constructed from the ground-truth image through the student visual pathway (encoder, merger, and masking recipe); they are never obtained by feeding back the model's preceding predictions. The shared backbone contextualizes these inputs under cross-scale causal attention, with bidirectional attention within each visual span, and a prediction head maps each parent hidden state to the features of its four spatially corresponding children, supervised by clean-image features from the separate frozen teacher. The 4-to-16-token transition is therefore learned conditional prediction through the backbone and head, not interpolation or duplication of a coarse grid. The 512-scale enters the sequence only as a prediction target.

Inference: recursive construction. At inference no ground-truth visual context is available. Starting from the caption, predicted states $\hat{\mathbf{z}}^{64}$, $\hat{\mathbf{z}}^{128}$, and $\hat{\mathbf{z}}^{256}$ are recursively reused as context for subsequent prediction through the same shared backbone,

$$\hat{\mathbf{z}}^{64} \rightarrow \hat{\mathbf{z}}^{128} \rightarrow \hat{\mathbf{z}}^{256} \rightarrow \hat{\mathbf{z}}^{512}, \tag{5}$$

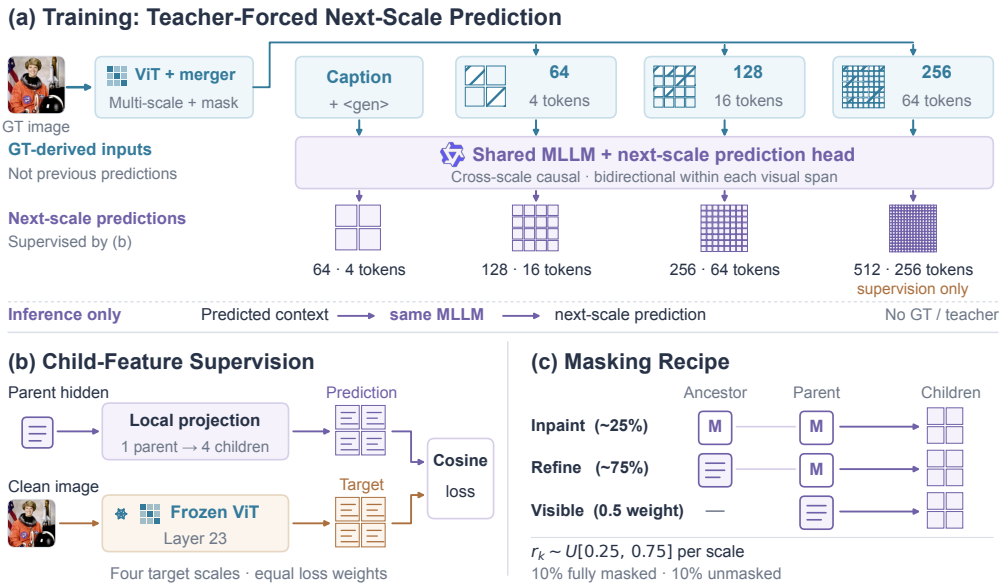


Figure 3: **Progressive masked representation prediction.** Each parent predicts four child representations at the next scale under frozen-teacher supervision, conditioned on ground-truth visual states during training and on recursively reused predictions at inference. Masking trains missing-context recovery and coarse-to-fine refinement.

so that a predicted visual state is both a generative output and a subsequent perceptual input. Conditioning each scale on its predecessors also distinguishes the objective from predicting all scales independently from the caption. The resulting train–test distinction is therefore explicit: training learns the conditional mappings using teacher-forced visual context, whereas inference composes the same mappings recursively using model-predicted states.

Masking. During training, visual states are randomly masked to expose the model to both missing-context recovery and coarse-to-fine refinement. In addition to partial masking, we include fully masked samples to train the text-only initialization required at the first generation scale, and unmasked samples to explicitly train refinement from available coarse states. Both masked and visible parents receive representation supervision. Detailed masking ratios and loss weights are provided in Appendix C. Figure 3 illustrates the resulting training and inference procedures.

3.3 Joint Multimodal Pretraining

Understanding and representation generation are optimized jointly within one backbone over a mixture of understanding and generation samples,

$$\mathcal{L} = \mathcal{L}_{\text{LM}} + \lambda_{\text{rep}}\,\mathcal{L}_{\text{rep}} + \lambda_{\text{rec}}\,\mathcal{L}_{\text{rec}}. \tag{6}$$

Understanding supervision. Understanding samples use standard autoregressive language-modeling supervision: the model predicts response tokens from the preceding text and visual tokens.

Representation generation. Generation samples are optimized using the masked progressive prediction objective of Section 3.2. All multimodal backbone parameters are shared with the understanding pathway.

Auxiliary reconstruction. For understanding samples, the final UMM-JEPA additionally appends a text-conditioned reconstruction tail that predicts the sample's multi-scale SigLIP2 representations without access to its original visual tokens; the reconstruction uses the same progressive representation objective and acts only as an auxiliary understanding-side signal. This component is not required for the central generation-to-understanding effect: we therefore use UMM-JEPA-core, which removes reconstruction, for causal attribution of the representation-prediction objective, and reserve the full UMM-JEPA for final performance reporting. Its contribution is separately quantified in Section 4.3, and implementation details are provided in Appendix C.

Shared attention structure. Visual tokens attend bidirectionally within each visual span, while text tokens remain causal (Guessous et al., 2024), allowing each visual span to form a complete internal representation before being read by text.

3.4 Decoupled Image Rendering

Rendering is post-pretraining. Pixel rendering does not participate in representation pretraining. After the multimodal representation backbone is learned, we freeze it and attach a lightweight connector together with a pretrained FLUX.2 renderer (Black Forest Labs, 2025); the connector maps the learned multimodal visual states to the conditioning space of the renderer. Representation pretraining and rendering are therefore two separate procedures with different training mechanisms: during representation pretraining (Sections 3.2 and 3.3) the visual context is teacher-forced and the supervision is applied in representation space, whereas during rendering the visual context is produced by the model itself and the supervision is applied in pixel space. This separation ensures that the pixel-space objective does not alter the representation-learning mechanism studied in this paper.

Connector alignment with a frozen renderer. In the first rendering stage, the representation backbone and the renderer are frozen and only the connector is optimized, on images sampled from the same image–text pool used for representation pretraining. The connector is fitted through the renderer's native rectified-flow objective. Training gradually transitions from teacher-forced visual conditioning toward prediction-feedback conditioning: the feedback probability is zero during early training and is then ramped up over a short interval, so that the connector is eventually fitted to the model-predicted visual states that are actually used at inference.

Prediction-feedback renderer adaptation. The representation backbone remains frozen. We jointly optimize the connector and lightweight LoRA adapters (Hu et al., 2022) inserted into the text-conditioning modules of the diffusion transformer, while the remaining renderer parameters stay frozen. To match recursive generation at inference, each training step uses a two-pass procedure. In the first pass, the frozen representation pathway predicts the multi-scale visual conditions without gradient updates, and the resulting one-step estimates are spliced back into the conditioning sequence as visual context. The second pass recomputes the forward pass from this model-predicted context and evaluates the diffusion velocity objective, and only this second-pass loss is back-propagated into the connector and the LoRA adapters. The rendering pathway is thus optimized under model-generated rather than purely teacher-forced visual conditions. A final generation instruction-tuning stage continues this training while additionally adapting low-rank adapters on the language backbone's attention projections. Text-only Qwen, Qwen-VL, and UMM-JEPA all use this recipe; their scores in Section 4.5 are taken after this stage. Numerical configurations of all rendering stages are given in Appendix C.

4 Experiments

4.1 Experimental setup

We train every model in Table 1 from the same pretrained Qwen3-1.7B checkpoint (Team, 2025a), on 10B text and 45B multimodal tokens. All of them also use SigLIP2 (Tschannen et al., 2025), except Qwen-EF, which keeps that LLM initialization but replaces the pretrained ViT with a patch embedding trained from scratch. Generation-bearing models add 10B image–text pairs, isolated in Section 4.3. Qwen-VL is understanding-only. Qwen-EF is an architecture reference, not a comparison of generation objectives. UMM-VQ and Show-o2 retrain discrete-code and latent-flow generation under the same extra budget. UMM-JEPA-core is masked next-scale representation prediction with bidirectional visual attention; UMM-JEPA adds auxiliary reconstruction. *Text*, *Visual Understanding*, and *Visual Specialty* are means of 4, 7, and 9 benchmarks. GenEval and DPG are kept separate. Definitions, citations, shots, and scoring are in Appendix B.

4.2 Does representation-based generative pretraining improve understanding?

UMM-JEPA reaches a 16-benchmark visual mean of 51.71 and a text mean of 61.29, compared with 46.36 and 59.16 for Qwen-VL (Table 1). Qwen-EF is substantially weaker on visual benchmarks, so the pretrained encoder is held fixed in the comparisons below.

Where the gains concentrate. The largest gains are on visually grounded specialty tasks: ChartQA ($+15.44$), RefCOCO ($+13.91$), DocVQA ($+11.73$), and VSR ($+6.54$), with OCRBench and

Table 1: Main pretraining comparison. Every system, including Qwen-EF, is trained by us from the same pretrained Qwen3-1.7B checkpoint. Qwen-EF keeps that initialization and replaces only the ViT with a patch embedding trained from scratch. The other systems also share SigLIP2; generation-bearing models add 10B image–text data (Table 2). Bold is best among the five systems. Blue parentheses are the gain of UMM-JEPA over Qwen-VL. HallusionBench uses aAcc.

	Benchmark	Qwen-EF	Qwen-VL	UMM-VQ	Show-o2	UMM-JEPA
Text understanding	MMLU	49.54	51.23	50.95	51.27	**51.57** (+0.34)
	GSM8K	76.12	76.42	74.60	74.60	**76.80** (+0.38)
	MGSM-zh	59.20	59.20	60.80	60.40	**61.60** (+2.40)
	MATH-500	53.20	49.80	52.20	49.60	**55.20** (+5.40)
Visual understanding	MMMU	43.72	41.61	44.67	44.11	**45.11** (+3.50)
	MMMU-Pro	20.17	20.64	20.95	21.24	**21.50** (+0.86)
	MMBench EN	14.86	**47.76**	47.14	47.69	**47.76** (+0.00)
	MMBench CN	12.00	33.98	37.27	37.30	**37.54** (+3.56)
	SEED-Bench	40.13	60.37	61.64	62.13	**62.42** (+2.05)
	SEED-Bench-2+	42.69	51.43	52.48	52.00	**52.88** (+1.45)
	ScienceQA	69.21	74.81	75.09	75.06	**75.61** (+0.80)
Visual specialty	ChartQA	9.32	41.20	56.36	56.40	**56.64** (+15.44)
	OCRBench	2.70	48.30	53.10	52.50	**55.90** (+7.60)
	TextVQA	5.78	48.83	57.70	57.89	**58.12** (+9.29)
	DocVQA	6.10	31.36	43.06	43.02	**43.09** (+11.73)
	HallusionBench	54.26	60.25	60.78	61.72	**62.04** (+1.79)
	VSR	51.88	49.02	52.78	52.70	**55.56** (+6.54)
	ViewSpatial	28.17	31.48	32.62	33.07	**34.21** (+2.73)
	RefCOCO	7.55	32.62	37.80	38.10	**46.53** (+13.91)
	CountBench	11.50	68.17	72.28	71.87	**72.48** (+4.31)

Table 2: Attribution and component analysis. "Extra 10B" is image–text data beyond the shared corpus. NSP arrives together with this data, so it is read against the data-matched row. The no-mask variant is scored on understanding only. Per-benchmark results are in Appendix F.

Variant	Extra 10B	Bi-Attn	NSP	Mask	Recon	Text ↑	VisU ↑	VisS ↑
Qwen-VL	×	×	×	–	×	59.16	47.23	45.69
Qwen-VL data-matched	✓	×	×	–	×	58.90	47.27	45.65
Qwen-VL + Bi-Attn	×	✓	×	–	×	59.65	47.79	45.81
w/o Mask	✓	✓	✓	×	×	59.90	48.42	51.43
UMM-JEPA-core (w/o Recon)	✓	✓	✓	✓	×	59.89	48.55	52.62
UMM-JEPA	✓	✓	✓	✓	✓	**61.29**	**48.97**	**53.84**

TextVQA following the same trend. The specialty mean rises from 45.69 to 53.84, while visual understanding rises more moderately, from 47.23 to 48.97.

No language cost. Text improves as well, from 59.16 to 61.29, led by MATH-500 ($+5.40$) and MGSM-zh ($+2.40$). Under this protocol, generative representation pretraining improves text and visual understanding together.

4.3 What contributes to the understanding improvement?

Table 2 reports the three group means. Adjacent rows are not all single-factor steps: next-scale prediction arrives together with the extra 10B, so that effect is read against the data-matched row.

Data-matched group means. Generation-bearing models consume an additional 10B image–text pairs. The data-matched control continues Qwen-VL on the same pairs, used only as image-to-caption supervision. The three means change by -0.26 / $+0.04$ / -0.04.

The generation objective matters. Relative to this control, UMM-JEPA-core improves Text / Visual Understanding / Visual Specialty by $+0.99$ / $+1.28$ / $+6.97$. The gain is concentrated on visual specialty. UMM-VQ and Show-o2, retrained under the same budget as paradigm controls, transfer less consistently, especially on text reasoning and the strongest visually grounded tasks.

The target space matters. UMM-VQ and UMM-JEPA-core share the extra data and the coarse-to-fine schedule. UMM-VQ predicts discrete DC-HT codes; UMM-JEPA-core predicts continuous SigLIP2 representations. The group means move from 59.64/48.46/51.83 to 59.89/48.55/52.62,

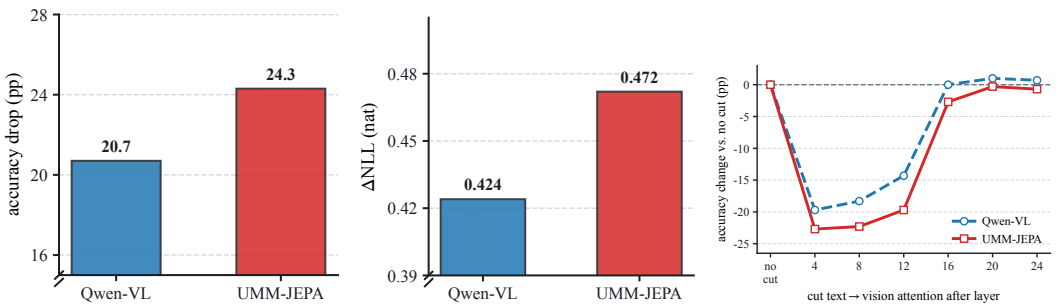


Figure 4: **Visual-evidence interventions** on 300 shared items (Appendix G). Left to right: (a) wrong-image accuracy drop and (b) gold-answer NLL increase, both larger for UMM-JEPA under a mismatched image; (c) blocking text-to-vision attention after later layers (uncut accuracy is 65.0 for Qwen-VL and 66.3 for UMM-JEPA). Value axes in (a–b) are truncated.

and full UMM-JEPA widens the gap. Perceptual representation targets transfer more directly to understanding. This comparison remains paradigm-level.

Which components carry the gain. Bidirectional attention alone changes the means only slightly. The jump in these means appears once representation prediction is introduced. Masking further raises visual specialty. Auxiliary reconstruction adds $+1.40$ / $+0.42$ / $+1.22$ and is not the main effect. The no-mask variant cannot initialize the empty 64-scale, so it is scored on understanding only. Per-benchmark numbers are in Appendix F.

4.4 How does generative representation learning change visual evidence use?

Linear probes are reported for Qwen-VL, the data-matched control, UMM-VQ, Show-o2, and UMM-JEPA (Table 3). Qwen-EF is left out because it does not use the SigLIP2 front end. Interventions compare Qwen-VL and UMM-JEPA, which share the Qwen3-1.7B LLM, the ViT, and DeepStack injection, but differ in attention, data, and training trajectory, so they compare two complete systems. Every intervention uses the same 300 items and teacher-forced gold-answer scoring. Protocol details are in Appendix G.

Table 3: Frozen final-layer probes, shared COCO split (Appendix H.1). mIoU: object-mask mean IoU at each patch center. Acc.: same-object grouping accuracy.

Model	mIoU ↑	Acc. ↑
Qwen-VL	22.97	61.28
Qwen-VL data-matched	23.21	61.89
UMM-VQ	24.61	63.58
Show-o2	23.90	63.37
UMM-JEPA	**27.58**	**68.06**

Linear probes. With the backbone frozen, linear heads on final-layer visual tokens test whether object and spatial structure can be read out without adapting the model (Appendix H.1). Object-mask mIoU measures category-level spatial structure; same-object accuracy measures instance grouping. UMM-JEPA reaches 27.58 mIoU and 68.06% (Table 3). The data-matched Qwen-VL control reaches only 23.21 and 61.89, so the extra image–text data does not explain the increase. UMM-VQ and Show-o2 fall in between: generative training enriches visual states, and predicting perceptual representations preserves more of that structure.

Wrong image. Replacing the image with a mismatched donor from the same pool, at unchanged token count, drops accuracy by 24.3 points for UMM-JEPA and 20.7 for Qwen-VL, and raises gold-answer NLL by 0.472 and 0.424, respectively (Figure 4a,b). The larger drop indicates a stronger dependence on image-specific evidence.

Layer-wise intervention. For every decoder layer with index $> k$ ($k \in \{4, 8, 12, 16, 20, 24\}$), text queries cannot attend to visual keys. Text→text, vision→vision, vision→text, and DeepStack injection into layers 0–2 stay intact, so the probe measures how long text keeps retrieving from visual tokens. Both models depend on that retrieval in early and middle layers. After layer 16, UMM-JEPA still loses 2.7 accuracy points, while Qwen-VL returns to about its uncut accuracy (Figure 4c; Appendix G.3). UMM-JEPA keeps retrieving from visual states deeper in the language backbone.

Together, the visual states of UMM-JEPA retain more linearly decodable object and spatial structure, its answers depend more on the specific image, and it retrieves visual information deeper

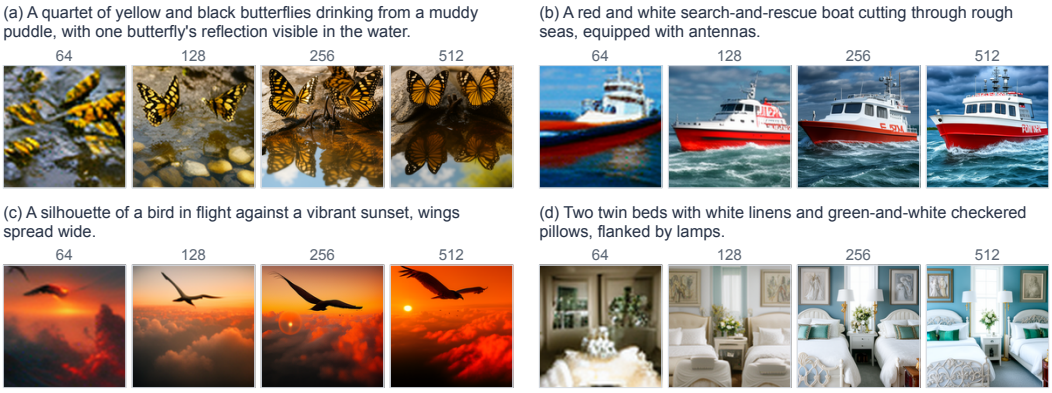


Figure 5: Progressive generation for four prompts. Each sequence is rendered at scales 64, 128, 256, and 512. Coarse scales set the layout; later scales add detail. Illustrative only.

into the LLM. Patch corruption, token diversity, scale-wise prediction error, and the prediction–understanding correlation are in Appendices G.2 and H.

4.5 Image generation results

Image generation is attached only after representation pretraining (Section 3.4; Appendix C) and is scored separately from understanding. UMM-JEPA reaches 81.2 on GenEval and 83.75 on DPG (Table 4). Figure 5 shows the same coarse-to-fine schedule at inference: scales 64 and 128 set the layout, and 256 and 512 add detail. Text-only Qwen, Qwen-VL, and UMM-JEPA share the same post-hoc pathway, a connector with DiT LoRA and generation SFT. Under that match, UMM-JEPA is above Text-only Qwen (69.2/71.62) and Qwen-VL (70.5/72.41). UMM-VQ and Show-o2 are likewise scored after generation SFT, on a VQ decoder and latent flow, at 57.3/62.52 and 67.3/67.36. UMM-JEPA stays above both and is competitive with the generation-oriented references.

Table 4: Separate from understanding (Appendix I). conn.+LoRA+SFT: connector, DiT LoRA, generation SFT. VQ dec.+SFT / latent flow+SFT: native decoder or latent flow, plus SFT. native FM: own renderer. FLUX.2: FLUX.2 [klein]. † Cited.

Model	Pathway	GenEval ↑	DPG ↑
Text-only Qwen	conn.+LoRA+SFT	69.2	71.62
Qwen-VL	conn.+LoRA+SFT	70.5	72.41
UMM-JEPA	conn.+LoRA+SFT	**81.2**	**83.75**
UMM-VQ	VQ dec.+SFT	57.3	62.52
Show-o2	latent flow+SFT	67.3	67.36
FLUX.2 4B†	native FM	78.6	83.30
BLIP3-o 4B†	native	81.0	79.36

4.6 Does the understanding advantage survive instruction tuning?

Starting from the Table 1 checkpoints, every model is fine-tuned on the same Honey-120K understanding data, a 120K subset of Honey-Data-1M (Zhang et al., 2025), for the same number of updates. UMM-JEPA remains strongest, at 62.76 / 54.15 / 58.39 on Text / Visual Understanding / Visual Specialty (Table 5). Qwen-EF stays far behind on both visual groups, consistent with the pretraining comparison. The understanding advantage therefore survives downstream adaptation. Per-benchmark results are in Appendix E.2.

Table 5: After the same understanding instruction tuning. Text, VisU, and VisS are means of the 4 text, 7 visual-understanding, and 9 visual-specialty benchmarks in Table 1. Per-benchmark results are in Appendix E.2.

Model	Text ↑	VisU ↑	VisS ↑
Qwen-EF	60.03	35.07	20.22
Qwen-VL	60.63	51.99	54.74
UMM-VQ	60.95	53.07	55.40
Show-o2	59.91	53.21	55.44
UMM-JEPA	**62.76**	**54.15**	**58.39**

5 Conclusion

UMM-JEPA learns hierarchical next-scale prediction directly over the continuous representations that understanding uses, with teacher-forced coarse-to-fine training, recursive inference, and pixel rendering kept out of core pretraining. Controlled comparisons show that this improves multimodal understanding beyond an understanding-only baseline, that a data-matched control leaves the three group means nearly unchanged (-0.26 / $+0.04$ / -0.04), and that perceptual representation targets transfer more consistently than discrete-code or latent-flow objectives; probes and interventions further indicate richer and deeper use of visual evidence. Generation is therefore most useful for understanding when it supervises that shared perceptual space.

AI USE STATEMENT

We used generative AI tools only to polish the language and wording of this manuscript. Their use was limited to improving clarity and expression, and was not extensive. The authors reviewed all suggested edits and take responsibility for the final content of this work.

REFERENCES

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael G. Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pp. 15619–15629. IEEE, 2023. doi: 10.1109/CVPR52729.2023.01499. URL https://doi.org/10.1109/CVPR52729.2023.01499.

Black Forest Labs. FLUX.2: Frontier visual intelligence. https://bfl.ai/blog/flux-2, November 2025.

Jiuhai Chen, Zhiyang Xu, Xichen Pan, Yushi Hu, Can Qin, Tom Goldstein, Lifu Huang, Tianyi Zhou, Saining Xie, Silvio Savarese, Le Xue, Caiming Xiong, and Ran Xu. Blip3-o: A family of fully open unified multimodal models-architecture, training and dataset. *CoRR*, abs/2505.09568, 2025a. doi: 10.48550/ARXIV.2505.09568. URL https://doi.org/10.48550/arXiv.2505.09568.

Junyu Chen, Han Cai, Junsong Chen, Enze Xie, Shang Yang, Haotian Tang, Muyang Li, and Song Han. Deep compression autoencoder for efficient high-resolution diffusion models. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025b. URL https://openreview.net/forum?id=wH8XXUOUZU.

Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pp. 1280–1289. IEEE, 2022. doi: 10.1109/CVPR52688.2022.00135. URL https://doi.org/10.1109/CVPR52688.2022.00135.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168, 2021. URL https://arxiv.org/abs/2110.14168.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, Shi Guang, and Haoqi Fan. Emerging properties in unified multimodal pretraining. *CoRR*, 2025. doi: 10.48550/arxiv.2505.14683. URL https://doi.org/10.48550/ARXIV.2505.14683.

Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/a3bf71c7c63f0c3bcb7ff67c67b1e7b1-Abstract-Datasets_and_Benchmarks.html.

Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, Dinesh Manocha, and Tianyi Zhou. Hallusionbench: An advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pp. 14375–14385. IEEE, 2024. doi: 10.1109/CVPR52733.2024.01363. URL https://doi.org/10.1109/CVPR52733.2024.01363.

Driss Guessous, Yanbo Liang, Joy Dong, and Horace He. Flexattention: The flexibility of pytorch with the performance of flashattention. https://pytorch.org/blog/flexattention/, August 2024. PyTorch Blog.

Junlin Han, Shengbang Tong, David Fan, Minghao Chen, Philip Torr, Filippos Kokkinos, and Mike Lewis. Towards physics of multimodal pretraining: Knowledge flow, modality synergy, early unification, and recipes. *CoRR*, abs/2608.05000, 2026. doi: 10.48550/ARXIV.2608.05000. URL https://doi.org/10.48550/arXiv.2608.05000.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021a. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In Joaquin Vanschoren and Sai-Kit Yeung (eds.), *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*, 2021b. URL https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/hash/be83ab3ecd0db773eb2dc1b0a17836a1-Abstract-round2.html.

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 *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Xiwei Hu, Rui Wang, Yixiao Fang, Bin Fu, Pei Cheng, and Gang Yu. ELLA: equip diffusion models with LLM for enhanced semantic alignment. *CoRR*, abs/2403.05135, 2024. doi: 10.48550/ARXIV.2403.05135. URL https://doi.org/10.48550/arXiv.2403.05135.

Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. Seed-bench: Benchmarking multimodal llms with generative comprehension. *CoRR*, abs/2307.16125, 2023. doi: 10.48550/ARXIV.2307.16125. URL https://doi.org/10.48550/arXiv.2307.16125.

Bohao Li, Yuying Ge, Yi Chen, Yixiao Ge, Ruimao Zhang, and Ying Shan. Seed-bench-2-plus: Benchmarking multimodal large language models with text-rich visual comprehension. *CoRR*, abs/2404.16790, 2024. doi: 10.48550/ARXIV.2404.16790. URL https://doi.org/10.48550/arXiv.2404.16790.

Chenliang Li, Haiyang Xu, Junfeng Tian, Wei Wang, Ming Yan, Bin Bi, Jiabo Ye, He Chen, Guohai Xu, Zheng Cao, Ji Zhang, Songfang Huang, Fei Huang, Jingren Zhou, and Luo Si. mplug: Effective and efficient vision-language learning by cross-modal skip-connections. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pp. 7241–7259. Association for Computational Linguistics, 2022. doi: 10.18653/V1/2022.EMNLP-MAIN.488. URL https://doi.org/10.18653/v1/2022.emnlp-main.488.

Dingming Li, Hongxing Li, Zixuan Wang, Yuchen Yan, Hang Zhang, Siqi Chen, Guiyang Hou, Shengpei Jiang, Wenqi Zhang, Yongliang Shen, Weiming Lu, and Yueting Zhuang. Viewspatial-bench: Evaluating multi-perspective spatial localization in vision-language models. In Paolo Favaro, Zuzana Kukelova, Atsuto Maki, Anna Rohrbach, Konrad Schindler, and Federico Tombari (eds.), *Computer Vision - ECCV 2026 - 19th European Conference, Malmö, Sweden, September 8-12, 2026, Proceedings, Part LXII*, volume 17062 of *Lecture Notes in Computer Science*, pp. 95–111. Springer, 2026. doi: 10.1007/978-3-032-37098-3_6. URL https://doi.org/10.1007/978-3-032-37098-3_6.

Han Li, Xinyu Peng, Yaoming Wang, Zelin Peng, Xin Chen, Rongxiang Weng, Jingang Wang, Xunliang Cai, Wenrui Dai, and Hongkai Xiong. Onecat: Decoder-only auto-regressive model for unified understanding and generation. *CoRR*, abs/2509.03498, 2025. doi: 10.48550/ARXIV.2509.03498. URL https://doi.org/10.48550/arXiv.2509.03498.

Weixin Liang, Lili Yu, Liang Luo, Srini Iyer, Ning Dong, Chunting Zhou, Gargi Ghosh, Mike Lewis, Wen-tau Yih, Luke Zettlemoyer, and Xi Victoria Lin. Mixture-of-transformers: A sparse and scalable architecture for multi-modal foundation models. *Trans. Mach. Learn. Res.*, 2025, 2025. URL https://openreview.net/forum?id=Nu6N69i8SB.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=v8L0pN6EOi.

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In David J. Fleet, Tomás Pajdla, Bernt Schiele, and Tinne Tuytelaars (eds.), *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*, volume 8693 of *Lecture Notes in Computer Science*, pp. 740–755. Springer, 2014. doi: 10.1007/978-3-319-10602-1_48. URL https://doi.org/10.1007/978-3-319-10602-1_48.

Fangyu Liu, Guy Emerson, and Nigel Collier. Visual spatial reasoning. *Trans. Assoc. Comput. Linguistics*, 11:635–651, 2023. doi: 10.1162/TACL_A_00566. URL https://doi.org/10.1162/tacl_a_00566.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. Mmbench: Is your multi-modal model an all-around player? In Ales Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part VI*, volume 15064 of *Lecture Notes in Computer Science*, pp. 216–233. Springer, 2024a. doi: 10.1007/978-3-031-72658-3_13. URL https://doi.org/10.1007/978-3-031-72658-3_13.

Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xu-Cheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. Ocrbench: on the hidden mystery of OCR in large multimodal models. *Sci. China Inf. Sci.*, 67(12), 2024b. doi: 10.1007/S11432-024-4235-6. URL https://doi.org/10.1007/s11432-024-4235-6.

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. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/11332b6b6cf4485b84afadb1352d3a9a-Abstract-Conference.html.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq R. Joty, and Enamul Hoque. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022*, volume ACL 2022 of *Findings of ACL*, pp. 2263–2279. Association for Computational Linguistics, 2022. doi: 10.18653/V1/2022.FINDINGS-ACL.177. URL https://doi.org/10.18653/v1/2022.findings-acl.177.

Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. Docvqa: A dataset for VQA on document images. In *IEEE Winter Conference on Applications of Computer Vision, WACV 2021, Waikoloa, HI, USA, January 3-8, 2021*, pp. 2199–2208. IEEE, 2021. doi: 10.1109/WACV48630.2021.00225. URL https://doi.org/10.1109/WACV48630.2021.00225.

Roni Paiss, Ariel Ephrat, Omer Tov, Shiran Zada, Inbar Mosseri, Michal Irani, and Tali Dekel. Teaching CLIP to count to ten. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pp. 3147–3157. IEEE, 2023. doi: 10.1109/ICCV51070.2023.00294. URL https://doi.org/10.1109/ICCV51070.2023.00294.

Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, Dipanjan Das, and Jason Wei. Language models are multilingual chain-of-thought reasoners. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=fR3wGCk-IXp.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards VQA models that can read. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pp. 8317–8326. Computer Vision Foundation / IEEE, 2019. doi: 10.1109/CVPR.2019.00851. URL http://openaccess.thecvf.com/content_CVPR_2019/html/Singh_Towards_VQA_Models_That_Can_Read_CVPR_2019_paper.html.

Zihan Su, Hongyang Wei, Kangrui Cen, Yong Wang, Guanhua Chen, Chun Yuan, and Xiangxiang Chu. Generation enhances understanding in unified multimodal models via multi-representation generation. *CoRR*, abs/2601.21406, 2026. doi: 10.48550/ARXIV.2601.21406. URL https://doi.org/10.48550/arXiv.2601.21406.

Quan Sun, Yufeng Cui, Xiaosong Zhang, Fan Zhang, Qiying Yu, Zhengxiong Luo, Yueze Wang, Yongming Rao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. Generative multimodal models are in-context learners. *CoRR*, 2023a. doi: 10.48550/arxiv.2312.13286. URL https://doi.org/10.48550/ARXIV.2312.13286.

Quan Sun, Qiying Yu, Yufeng Cui, Fan Zhang, Xiaosong Zhang, Yueze Wang, Hongcheng Gao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. Generative pretraining in multimodality. *CoRR*, abs/2307.05222, 2023b. doi: 10.48550/ARXIV.2307.05222. URL https://doi.org/10.48550/arXiv.2307.05222.

Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models. *CoRR*, abs/2405.09818, 2024. doi: 10.48550/ARXIV.2405.09818. URL https://doi.org/10.48550/arXiv.2405.09818.

Qwen Team. Qwen3 technical report. *CoRR*, abs/2505.09388, 2025a. doi: 10.48550/ARXIV.2505.09388. URL https://doi.org/10.48550/arXiv.2505.09388.

Qwen Team. Qwen3-vl technical report. *CoRR*, abs/2511.21631, 2025b. doi: 10.48550/ARXIV.2511.21631. URL https://doi.org/10.48550/arXiv.2511.21631.

Keyu Tian, Yi Jiang, Zehuan Yuan, Bingyue Peng, and Liwei Wang. Visual autoregressive modeling: Scalable image generation via next-scale prediction. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/9a24e284b187f662681440ba15c416fb-Abstract-Conference.html.

Michael Tschannen, Alexey A. Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier J. Hénaff, Jeremiah Harmsen, Andreas Steiner, and Xiaohua Zhai. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *CoRR*, abs/2502.14786, 2025. doi: 10.48550/ARXIV.2502.14786. URL https://doi.org/10.48550/arXiv.2502.14786.

Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, Yingli Zhao, Yulong Ao, Xuebin Min, Tao Li, Boya Wu, Bo Zhao, Bowen Zhang, Liangdong Wang, Guang Liu, Zheqi He, Xi Yang, Jingjing Liu, Yonghua Lin, Tiejun Huang, and Zhongyuan Wang. Emu3: Next-token prediction is all you need. *CoRR*, abs/2409.18869, 2024. doi: 10.48550/ARXIV.2409.18869. URL https://doi.org/10.48550/arXiv.2409.18869.

Yuqing Wang, Zhijie Lin, Ceyuan Yang, Yang Zhao, Fei Xiao, Hao He, Qi Zhao, Zihan Ding, Fuyun Wang, Shuai Wang, Youliang Zhang, Haoqi Fan, and Xihui Liu. Representation forcing

for bottleneck-free unified multimodal models. *CoRR*, abs/2605.31604, 2026. doi: 10.48550/ARXIV.2605.31604. URL https://doi.org/10.48550/arXiv.2605.31604.

Zimo Wen, Boxiu Li, Wanbo Zhang, Junxiang Lei, Xiaoyu Chen, Yijia Fan, Qi Zhang, Yujiang Wang, Lili Qiu, Bo Li, Ziwei Liu, Caihua Shan, Yifan Yang, and Yifei Shen. Unig2u-bench: Do unified models advance multimodal understanding? *CoRR*, abs/2603.03241, 2026. doi: 10.48550/ARXIV.2603.03241. URL https://doi.org/10.48550/arXiv.2603.03241.

Chengyue Wu, Xiaokang Chen, Zhiyu Wu, Yiyang Ma, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, Chong Ruan, and Ping Luo. Janus: Decoupling visual encoding for unified multimodal understanding and generation. *CoRR*, abs/2410.13848, 2024. doi: 10.48550/ARXIV.2410.13848. URL https://doi.org/10.48550/arXiv.2410.13848.

Penghao Wu, Haiwen Diao, Weichen Fan, Lewei Lu, Dahua Lin, and Ziwei Liu. Uncovering understanding-generation synergy in native unified multimodal models: From representation, task to system. *arXiv preprint arXiv:2609.01607*, 2026.

Jinheng Xie, Zhenheng Yang, and Mike Zheng Shou. Show-o2: Improved native unified multimodal models. *CoRR*, abs/2506.15564, 2025. doi: 10.48550/ARXIV.2506.15564. URL https://doi.org/10.48550/arXiv.2506.15564.

Licheng Yu, Patrick Poirson, Shan Yang, Alexander C. Berg, and Tamara L. Berg. Modeling context in referring expressions. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling (eds.), *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II*, volume 9906 of *Lecture Notes in Computer Science*, pp. 69–85. Springer, 2016. doi: 10.1007/978-3-319-46475-6_5. URL https://doi.org/10.1007/978-3-319-46475-6_5.

Xiang Yue, Yuansheng Ni, Tianyu Zheng, Kai Zhang, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pp. 9556–9567. IEEE, 2024a. doi: 10.1109/CVPR52733.2024.00913. URL https://doi.org/10.1109/CVPR52733.2024.00913.

Xiang Yue, Tianyu Zheng, Yuansheng Ni, Yubo Wang, Kai Zhang, Shengbang Tong, Yuxuan Sun, Botao Yu, Ge Zhang, Huan Sun, Yu Su, Wenhu Chen, and Graham Neubig. Mmmu-pro: A more robust multi-discipline multimodal understanding benchmark. *CoRR*, abs/2409.02813, 2024b. doi: 10.48550/ARXIV.2409.02813. URL https://doi.org/10.48550/arXiv.2409.02813.

Yi Zhang, Bolin Ni, Xin-Sheng Chen, Hengrui Zhang, Yongming Rao, Houwen Peng, Qinglin Lu, Han Hu, Meng-Hao Guo, and Shi-Min Hu. Bee: A high-quality corpus and full-stack suite to unlock advanced fully open mllms. *CoRR*, abs/2510.13795, 2025. doi: 10.48550/ARXIV.2510.13795. URL https://doi.org/10.48550/arXiv.2510.13795.

A Extended related work

This appendix expands the summary of Section 2 along the four axes that motivate UMM-JEPA.

A.1 Unified understanding and generation

Unified multimodal models adopt markedly different generative parameterizations, including autoregressive discrete visual tokens (Team, 2024; Wang et al., 2024; Wu et al., 2024), next-scale visual autoregression (Tian et al., 2024), and latent flow matching (Xie et al., 2025), together with modality-specialized or partially separated semantic and generative pathways (Liang et al., 2025; Li et al., 2025; Deng et al., 2025). Architectural unification, however, does not determine whether generative training improves the representations used for understanding; we study how the choice of generative target affects that transfer.

A.2 Generation-to-understanding transfer

Controlled studies reveal an asymmetric knowledge flow: understanding supervision benefits generation, whereas additional generation training does not consistently improve understanding, with gains depending on the task and reasoning regime (Han et al., 2026; Wen et al., 2026; Wu et al., 2026). Auxiliary generation targets such as pixel reconstruction, depth, or segmentation have also been shown to help perception (Su et al., 2026). We ask the complementary and more specific question of whether transfer depends on aligning the *target of generation* with the perceptual representation consumed by understanding; rather than adding several output modalities, we make prediction of that representation the pretraining objective and isolate the choice through data-matched and paradigm controls.

A.3 Representation-based visual generation

Prior work places semantically meaningful representations inside the generative pathway, through continuous visual embeddings (Sun et al., 2023b;a), diffusion in a CLIP feature space (Chen et al., 2025a), or representation prediction as native intermediate generative states that remain in context to guide pixel diffusion (Wang et al., 2026). We differ in using representation prediction as the object of generative pretraining and in removing image rendering from core pretraining; pixel synthesis is attached only afterward, through a connector and a pretrained diffusion renderer.

A.4 Predictive representation learning

Joint-Embedding Predictive Architectures predict target representations from available context instead of reconstructing the observation (Assran et al., 2023), showing that prediction in representation space encourages semantic visual representations. This principle motivates our learning target, but our setting is conditional multimodal generation with progressive prediction across a coarse-to-fine hierarchy, and our contribution is to use the principle to reformulate generation as shared representation learning and to study its transfer to understanding.

B Compared systems and evaluation

Models. Except for the published references marked † in Table 4, every model in our tables is trained by us from the same pretrained Qwen3-1.7B checkpoint. All except the encoder-free reference share the same SigLIP2 visual front end. Generation-bearing controls are trained under the same additional image–text data budget as UMM-JEPA, and the Qwen-VL baseline is complemented by a data-matched understanding-only control (Section 4.3).

- **Qwen-EF**: an encoder-free architecture reference, also initialized from the same pretrained Qwen3-1.7B checkpoint. Only the pretrained ViT is replaced by a patch embedding trained from scratch. It shows the effect of removing the pretrained visual encoder, and is not a controlled comparison of generation objectives.
- **Qwen-VL**: our understanding-only run from the same Qwen3-1.7B initialization (Team, 2025b), with causal attention and no additional image–text data.

- **UMM-VQ**: keeps the next-scale schedule but predicts discrete DC-HT tokens (4/16/64/256-way classification targets at scales 64/128/256/512) with cross-entropy.
- **Show-o2**: retrained with its native unified mechanism (Xie et al., 2025) (latent flow matching over a frozen DC-AE latent space (Chen et al., 2025b)) under the same data budget.
- **UMM-JEPA-core**: bidirectional visual attention and masked next-scale representation prediction, without reconstruction.
- **UMM-JEPA**: UMM-JEPA-core plus auxiliary reconstruction.

Data. All systems are trained on the same 10B text and 45B multimodal corpus. Generation-bearing models (UMM-VQ, Show-o2, UMM-JEPA-core, UMM-JEPA) additionally consume 10B image–text pairs. The data-matched control continues Qwen-VL on these pairs through image-to-caption understanding supervision only.

Evaluation. Understanding covers 4 text benchmarks (MMLU (Hendrycks et al., 2021a), GSM8K (Cobbe et al., 2021), MGSM-zh (Shi et al., 2023), MATH-500 (Hendrycks et al., 2021b; Lightman et al., 2024)), 7 visual-understanding benchmarks (MMMU (Yue et al., 2024a), MMMU-Pro (Yue et al., 2024b), MMBench EN/CN (Liu et al., 2024a), SEED-Bench (Li et al., 2023), SEED-Bench-2+ (Li et al., 2024), ScienceQA (Lu et al., 2022)), and 9 visual-specialty benchmarks (ChartQA (Masry et al., 2022), OCRBench (Liu et al., 2024b), TextVQA (Singh et al., 2019), DocVQA (Mathew et al., 2021), HallusionBench (Guan et al., 2024), VSR (Liu et al., 2023), ViewSpatial (Li et al., 2026), RefCOCO (Yu et al., 2016), CountBench (Paiss et al., 2023)). *Text*, *Visual Understanding*, and *Visual Specialty* are the means of these 4, 7, and 9 benchmarks, and *Visual* is the mean of all 16. Table 1 reports every benchmark individually. Shot counts are in Appendix C. Image generation uses GenEval (Ghosh et al., 2023) and DPG (Hu et al., 2024) (Section 4.5): GenEval measures compositional fidelity on short prompts, and DPG evaluates entity and relation alignment on longer prompts. The two scores are reported separately and are never averaged with understanding scores. Both are scored locally (Appendix I).

C Implementation details

Architecture. Qwen3-1.7B LLM (Team, 2025a) (28 layers, hidden 2048, causal over text) with a SigLIP2 ViT (Tschannen et al., 2025) (24 layers, hidden 1024); DeepStack-style injection (Team, 2025b) injects ViT features into LLM layers 0/1/2. Every compared run initializes the LLM from the official pretrained Qwen3-1.7B weights. Understanding and generation share all LLM parameters.

Sequence and data. Packed sequence length 4096 (UMM-JEPA with the reconstruction tail: 4224). Data mixture 10B text + 45B multimodal; generation-bearing models add 10B image–text pairs (10:45:10). UMM-JEPA stage 2 uses batch size 2000; the Qwen-VL baseline uses a causal LLM with THD packing, batch size 2016, and no additional 10B.

Pretraining stages. Representation pretraining proceeds in two stages. *Alignment*: the ViT and the LLM are frozen and only the merger is trained on high-quality image–caption pairs for vision–language alignment. *Joint pretraining*: all parameters are unfrozen and trained with cross-entropy and the masked next-scale representation objective of Section 3.2 on 10B text + 45B multimodal tokens, plus 10B image–text tokens for the generation samples; visual patches attend bidirectionally while text remains causal (FlexAttention).

UMM-JEPA objective hyperparameters. NSP path caption $\to 64 \to 128 \to 256 \to 512$; frozen SigLIP2 layer-23 targets; cosine loss; four scales equally weighted; representation loss weight $\lambda_{\text{rep}} = 0.55$. For masking, each scale draws its ratio independently from $\text{Uniform}[0.25, 0.75]$ and clamps it to $[1, n-1]$ for its token count n, and the recipe additionally contains 10% fully masked samples (matching text-to-image inference, where the 64-scale is predicted from the caption alone) and 10% unmasked samples (matching the connector stage, which forbids next-scale masking). At the $128/256$ scales, masked tokens split into about 25% *inpaint* (ancestors also masked; children predicted purely from holes) and about 75% *refine* (ancestors visible; finer children predicted on top of coarse content), and masked and visible parents are jointly supervised as $(\mathcal{L}_{\text{masked}} + 0.5\,\mathcal{L}_{\text{visible}})/1.5$ so that the refinement mapping used at inference is trained rather than only the hole-filling mapping. Each scale owns a learnable mask token, while the local projection is shared across scales.

Connector and rendering stages. The connector is not part of UMM-JEPA pretraining; it is trained only after the representation has been learned. Text-only Qwen, Qwen-VL, and UMM-JEPA each receive this full recipe, including DiT LoRA and generation SFT, with weights fitted independently. UMM-VQ and Show-o2 keep their native decoders and add generation SFT. The connector architecture and the rendering data are shared across the connector-pathway sources. A ≈43.9M connector ($\mathrm{LN} \to \mathrm{Linear}(2048{\to}4096) \to \mathrm{SiLU} \to \mathrm{Linear}(4096{\to}7680) \to \mathrm{LN}$, plus a scale embedding marking which visual span each token comes from, timestep AdaLN, and 10% CFG dropout to the null condition) translates the 1.7B LLM's 2048-d final-layer hidden states into the 7680-d text-conditioning space of a frozen FLUX.2 [klein] base-4B diffusion transformer (Black Forest Labs, 2025). The conditioning sequence slices the visual spans by scale: the 64-scale contributes 4 tokens, the 128-scale $4{+}16$, and the 256-scale $4{+}16{+}64$ (the 512-scale is not encoded).

Stage 1: connector alignment. The LLM, SigLIP2, merger, VAE, and DiT are all frozen and only the connector is updated, using the DiT's native rectified-flow objective (v-prediction, $x_t = (1{-}t)z + t\epsilon$, logit-normal t, per-resolution weights 4:4:2:1) with gradients flowing through the frozen DiT's activations back to the connector. Training runs for 10,000 steps on 8 GPUs with batch size 256, on data randomly sampled from the same 10B image–text generation pool used for representation pretraining. We use Adam ($\beta_1 = 0.9$, $\beta_2 = 0.95$, $\epsilon = 10^{-8}$, weight decay 0.01, gradient clipping 1.0) with a cosine learning-rate schedule from 10^{-4} to 10^{-5} and 500 warmup steps. Prediction feedback is disabled during early training ($p_{\text{feedback}} = 0$) and is then ramped to $p_{\text{feedback}} = 0.8$, starting at step 6,000 over approximately 1,000 steps.

Stage 2: renderer adaptation with prediction feedback. The multimodal backbone, the VAE, the DiT main weights, the text-modulation modules, the image branch, and the single-stream blocks all remain frozen. We optimize only the connector together with lightweight LoRA adapters inserted into the text-conditioning modules of the diffusion transformer: rank $r = 16$ with $\alpha = 16$ (so the scaling factor is 1) on `txt_in` and on five double blocks, targeting `txt_attn.qkv`, `txt_attn.proj`, `txt_mlp.0`, and `txt_mlp.2`. The adapters contain 4,349,952 ≈ 4.3M parameters and, together with the ≈ 43.9M connector, give ≈ 48.2M trainable parameters. Every step is trained with prediction feedback ($p_{\text{feedback}} = 1.0$) using a two-pass procedure. In the first pass, the frozen representation pathway predicts the $64/128/256$ scales in a single denoising step with $t \sim \mathcal{U}(0.15, 0.45)$ and without gradient computation; the resulting one-step x_0 estimates are spliced back into the conditioning sequence as visual context. The second pass recomputes the forward pass from this predicted context and evaluates the diffusion velocity objective over the $64/128/256/512$ scales with per-resolution weights 4:4:2:1, and only this second-pass loss is back-propagated. Training runs for 10,000 steps on 8 GPUs with batch size 256 (no gradient accumulation) and CFG dropout 0.1, using the same Adam configuration, gradient clipping, and data source as above, with a cosine learning-rate schedule from 10^{-5} to 10^{-6} and 200 warmup steps.

Stage 3: generation instruction tuning. Starting from the adapted connector and diffusion adapters, we continue training the connector and the DiT LoRA adapters and additionally introduce LoRA adapters on the Qwen attention $q/k/v/o$ projections. The Text-only Qwen, Qwen-VL, and UMM-JEPA entries in Table 4 are taken from the checkpoint after this stage (6,000 steps). UMM-VQ and Show-o2 are scored from their own decoders after generation SFT. FLUX.2 and BLIP3-o are published references.

Evaluation protocol. Shots per benchmark: MMLU 0, GSM8K 0, MGSM-zh 8, MATH-500 0; MMMU 3, MMMU-Pro 3, MMBench EN 3, MMBench CN 0, SEED-Bench 0, SEED-Bench-2+ 3, ScienceQA 0; ChartQA 0, OCRBench 0, TextVQA 0, DocVQA 0, HallusionBench 0 (aAcc), VSR 0, ViewSpatial 3, RefCOCO 0, CountBench 0. All columns of Table 1 use the same protocol per benchmark.

Reconstruction tail of the final UMM-JEPA configuration. The tail is appended only to understanding samples; generation samples are unchanged, and the packed sequence length grows from 4096 to 4224 (+128 tokens). The original packed understanding row is kept intact and a reconstruction segment is appended to its end: a random instruction, a `<gen>` marker, the 64, 128, and 256 visual spans, and a closing `</gen>` marker. The target image is the last image of that packed row, and the entire reconstruction segment carries CE label $= -100$, so it does not contribute to the language-modeling loss.

Visual isolation. FlexAttention isolates the original image tokens preceding <gen> from the reconstruction segment, so the reconstruction cannot directly attend back to the target image and must rebuild that image's visual representations from the textual context of the understanding sample alone.

Targets, scales, and loss. The teacher is frozen SigLIP2 layer 23, i.e., the last layer of the 24-layer tower, and its features are LayerNorm-ed before serving as targets; the student prediction is read from the LLM decoder hidden states. The four scales $64/128/256/512$ are equally weighted (1:1:1:1) and normalized, and follow the same progressive relation as pretraining (<gen>$\rightarrow 64$, $64 \rightarrow 128$, $128 \rightarrow 256$, $256 \rightarrow 512$; as in pretraining, the 512 scale is supervised as a prediction target rather than encoded in the sequence). Visible parents receive weight 0.5 relative to masked parents. The loss is the cosine discrepancy $\mathcal{L}_{\text{rec}} = 1 - \cos(\hat{\mathbf{z}}, \mathbf{z})$, not MSE or L1, with weight $\lambda_{\text{rec}} = 0.55$.

Mask recipe. 10% of reconstruction samples are fully masked and 10% are unmasked; the remaining samples use connected-block masking with the ratio drawn from $[0.25, 0.75]$, which yields the approximately 25% inpaint setting of the main objective. No context noise, blur, or cutout is applied, and the understanding-side representation loss is disabled outside the reconstruction tail.

Training schedule: representation pretraining. The reported run is the stage-2 joint representation pretraining, initialized from the SigLIP2 v1 stage-1 alignment checkpoint; no separate alignment stage is run for this model. The final checkpoint is at iteration 7,694. The learning rate follows a cosine schedule with 125 warmup steps, decaying from 6×10^{-5} to 6×10^{-6} for the LLM and from 3×10^{-5} to 3×10^{-6} for the ViT and merger, using the Adam configuration given for the rendering stages. Batch size is fixed at 2000 on 8 GPUs with no batch-size ramp. Sequences are packed to $4224 = 4096 + 128$, where the 128 tokens are the reconstruction-tail headroom. The token budget is 45B multimodal, 10B text, and 10B generation image–text tokens.

Training schedule: rendering stages. The frozen-renderer connector stage and the renderer-adaptation stage use the configurations given above (10,000 steps each; 8 GPUs, batch size 256, $10^{-4} \rightarrow 10^{-5}$ cosine with 500 warmup and feedback $0 \rightarrow 0.8$ from step 6,000 for the connector; 8 GPUs, batch size 256, $10^{-5} \rightarrow 10^{-6}$ cosine with 200 warmup and $p_{\text{feedback}} = 1.0$ with a two-pass step for renderer adaptation). Both stages sample their images randomly from the same 10B image–text generation pool.

Training schedule: understanding instruction tuning. Understanding instruction tuning uses the Honey-120K subset of Honey-Data-1M (Zhang et al., 2025) with 45,765 packed rows for 2 epochs. The learning rate follows a cosine schedule with 35 warmup steps, from 10^{-5} to 10^{-6} for the LLM and from 5×10^{-6} to 5×10^{-7} for the ViT. Batch size is 256 on 8 GPUs with a packed sequence length of 4096.

D Theoretical Analysis

D.1 Assumptions and Notation

We consider an image $\mathbf{x} = \mathcal{R}(\mathbf{s}, \mathbf{n})$ generated from a task-relevant latent scene state $\mathbf{s}$ and rendering-specific nuisance variables $\mathbf{n}$, a perceptual encoder $\mathbf{z} = E(\mathbf{x})$, and an understanding mapping $\mathbf{y} = h(\mathbf{z})$. Vectors are compared with the Euclidean norm, $\sigma(\cdot)$ denotes the σ-algebra generated by a random variable, and risk is measured under squared loss. The analysis assumes:

- *Locally regular understanding mapping.* On the region of representation space visited during training, h is L-Lipschitz.
- *Squared prediction loss.* Prediction error is measured by squared Euclidean distance in representation space.
- *Nested conditioning.* For each scale, the coarse conditioning variable $\mathcal{C}_k = (C, \mathbf{Z}^{64}, \ldots, \mathbf{Z}^k)$ generates at least the information of the text condition C, i.e., $\sigma(C) \subseteq \sigma(\mathcal{C}_k)$.
- *Masked target exclusion.* When the states in M are masked, they are removed from the model's observation, and the observed variables are C and $\mathbf{Z}_{\bar{M}}$.

Nothing below relies on a parametric form for E, h, or the data distribution.

D.2 Proof of Proposition 1

Proof. By (A1), h is L-Lipschitz on the relevant region, so for any predicted state $\hat{\mathbf{z}}$ and target $\mathbf{z}$ we have $\|h(\hat{\mathbf{z}}) - h(\mathbf{z})\| \le L\|\hat{\mathbf{z}} - \mathbf{z}\|$. Squaring and taking expectations,

$$\mathbb{E}\|h(\hat{\mathbf{z}}) - h(\mathbf{z})\|^2 \le L^2\, \mathbb{E}\|\hat{\mathbf{z}} - \mathbf{z}\|^2, \tag{7}$$

so the deviation of the downstream prediction from its target is controlled by the representation-space prediction error itself. The constant is tight: for a linear mapping $h(\mathbf{z}) = W\mathbf{z}$ with $\|W\|_2 = L$, equality holds whenever $\hat{\mathbf{z}} - \mathbf{z}$ aligns with the leading singular direction of W. The statement bounds the sensitivity of the understanding output to errors in the state that understanding consumes; it does not compare this bound with the sensitivity of any other prediction target. □

D.3 Proof of Lemma 1

Proof. Under squared loss, the minimum achievable risk of predicting $\mathbf{Z}^{k+1}$ from a conditioning variable U is the expected conditional variance,

$$\mathcal{R}^*(\mathbf{Z}^{k+1} \mid U) = \mathbb{E}\big[\|\mathbf{Z}^{k+1} - \mathbb{E}[\mathbf{Z}^{k+1} \mid U]\|^2\big] = \mathbb{E}\big[\operatorname{Var}(\mathbf{Z}^{k+1} \mid U)\big], \tag{8}$$

since the conditional mean is the Bayes-optimal predictor. By (A3), $\sigma(C) \subseteq \sigma(\mathcal{C}_k)$, so $\mathcal{C}_k$ is an enrichment of C. The law of total variance gives, pointwise,

$$\operatorname{Var}(\mathbf{Z}^{k+1} \mid C) = \operatorname{Var}\big(\mathbb{E}[\mathbf{Z}^{k+1} \mid \mathcal{C}_k] \mid C\big) + \mathbb{E}\big[\operatorname{Var}(\mathbf{Z}^{k+1} \mid \mathcal{C}_k) \mid C\big] \;\ge\; \mathbb{E}\big[\operatorname{Var}(\mathbf{Z}^{k+1} \mid \mathcal{C}_k) \mid C\big]. \tag{9}$$

Taking expectations and substituting the risk identity yields

$$\mathcal{R}^*(\mathbf{Z}^{k+1} \mid C) \;\ge\; \mathbb{E}\big[\operatorname{Var}(\mathbf{Z}^{k+1} \mid \mathcal{C}_k)\big] = \mathcal{R}^*(\mathbf{Z}^{k+1} \mid \mathcal{C}_k). \tag{10}$$

Conditioning on additional coarse visual states can therefore only reduce the Bayes-optimal fine-scale risk, and it does so strictly whenever those states carry information about the fine scale beyond the text condition. The lemma constrains the value of the conditioning variables, not which scales or how many of them are used; the scale schedule is an implementation choice. □

D.4 Proof of Proposition 2

Proof. Under squared loss and (A4), the observed variables for a masked prediction are C and $\mathbf{Z}_{\bar{M}}$, and the risk of a predictor f is

$$\mathbb{E}\big[\|f(C, \mathbf{Z}_{\bar{M}}) - \mathbf{Z}_M\|^2\big], \tag{11}$$

which is finite and convex in f. The minimizer is therefore obtained pointwise, since the choice of f at one value of $(C, \mathbf{Z}_{\bar{M}})$ does not affect the integrand elsewhere,

$$f^*(C, \mathbf{Z}_{\bar{M}}) = \arg\min_v \; \mathbb{E}\big[\|\mathbf{Z}_M - v\|^2 \mid C, \mathbf{Z}_{\bar{M}}\big] = \mathbb{E}[\mathbf{Z}_M \mid C, \mathbf{Z}_{\bar{M}}]. \tag{12}$$

Because $\mathbf{Z}_M$ is excluded from the observation, the optimum must be formed from the caption and the visible states alone, i.e., from textual, spatial, and cross-scale context. If M is the full state set, (A4) leaves C as the only observation and the predictor degenerates to $f^*(C) = \mathbb{E}[\mathbf{Z} \mid C]$, the text-only case discussed after Proposition 2 in the main text. □

D.5 Scope of the Analysis

The analysis motivates the choice of prediction target, progressive conditioning, and masking under simplified prediction assumptions. It does not claim that the particular scale schedule or masking ratio is theoretically optimal; these choices are evaluated empirically.

Table 6: Encoder-free architecture reference (Qwen-EF), full per-benchmark results. Qwen-EF is included as an architecture reference in Table 1; the table here reports its complete before/after instruction-tuning breakdown.

	Benchmark	Qwen-EF (pretraining)	Qwen-EF (instr. tuned)
Text	MMLU	49.54	50.18
	GSM8K	76.12	77.33
	MGSM-zh	59.20	62.00
	MATH-500	53.20	50.60
Visual understanding	MMMU	43.72	39.67
	MMMU-Pro	20.17	20.29
	MMBench EN	14.86	16.41
	MMBench CN	12.00	15.25
	SEED-Bench	40.13	40.61
	SEED-Bench-2+	42.69	42.51
	ScienceQA	69.21	70.75
Visual specialty	ChartQA	9.32	12.44
	OCRBench	2.70	2.60
	TextVQA	5.78	7.29
	DocVQA	6.10	7.08
	HallusionBench	54.26	57.83
	VSR	51.88	50.08
	ViewSpatial	28.17	29.24
	RefCOCO	7.55	6.97
	CountBench	11.50	8.42

E Full per-benchmark results

E.1 Encoder-free reference (architecture sanity check)

Table 6 provides the full pretraining and instruction-tuning results of the encoder-free architecture reference. Qwen-EF is initialized from the same pretrained Qwen3-1.7B checkpoint as the other systems and replaces only the SigLIP2 front end with a patch embedding trained from scratch. Table 6 reports its per-benchmark results before and after understanding instruction tuning. It collapses on the visually grounded specialty benchmarks (for example 2.70 vs. 48.30 on OCRBench at pretraining), which is why a pretrained visual front end is treated as a prerequisite at this scale rather than as a variable under study.

E.2 Understanding instruction tuning per benchmark

Table 7 reports the per-benchmark results behind the group means of Table 5. Several of the largest pretraining gaps on visual specialty narrow after this tuning: ChartQA moves from $+15.44$ to $+0.04$, and RefCOCO from $+13.91$ to $+2.41$. OCRBench, TextVQA, DocVQA, and CountBench keep a clearer margin, and the group means in Table 5 remain the highest.

F Full component ablations

Table 8 reports the per-benchmark results behind the group means of the component rows of Table 2. Bold marks the best among the five columns. Auxiliary reconstruction is uneven across benchmarks: relative to the w/o Recon column, full UMM-JEPA is lower on MMBench EN, ScienceQA, ChartQA, DocVQA, and VSR. The group-mean increase comes mainly from text and from RefCOCO ($39.80 \rightarrow 46.53$).

Table 7: Per-benchmark results after understanding instruction tuning.

	Benchmark	Qwen-VL	UMM-VQ	Show-o2	UMM-JEPA
Text	MMLU	51.91	49.59	50.47	**52.76**
	GSM8K	76.42	76.19	76.57	**77.48**
	MGSM-zh	63.60	62.40	62.40	**64.00**
	MATH-500	50.60	55.60	50.20	**56.80**
Visual understanding	MMMU	40.50	43.17	43.00	**44.94**
	MMMU-Pro	22.43	22.77	23.06	**23.53**
	MMBench EN	53.56	54.02	55.65	**56.58**
	MMBench CN	53.02	53.41	53.95	**54.41**
	SEED-Bench	65.77	66.07	65.73	**66.29**
	SEED-Bench-2+	54.72	54.28	53.97	**54.90**
	ScienceQA	73.92	77.74	77.14	**78.38**
Visual specialty	ChartQA	62.76	61.60	62.76	**62.80**
	OCRBench	57.80	58.30	58.00	**62.40**
	TextVQA	61.16	61.89	61.66	**65.41**
	DocVQA	53.48	52.39	52.39	**60.40**
	HallusionBench	61.09	61.30	61.72	**62.25**
	VSR	48.53	48.53	48.53	**51.39**
	ViewSpatial	33.05	33.14	34.40	**35.08**
	RefCOCO	46.17	46.33	43.12	**48.58**
	CountBench	68.58	75.15	76.39	**77.21**

Table 8: Per-benchmark component-ablation results (pretraining stage).

	Benchmark	Qwen-VL	+Bi-Attn	w/o Mask	w/o Recon	UMM-JEPA
Text	MMLU	51.23	51.35	51.36	51.40	**51.57**
	GSM8K	76.42	75.66	75.82	75.97	**76.80**
	MGSM-zh	59.20	60.00	60.40	60.00	**61.60**
	MATH-500	49.80	51.60	52.00	52.20	**55.20**
Visual understanding	MMMU	41.61	43.00	41.89	42.11	**45.11**
	MMMU-Pro	20.64	21.18	21.16	21.39	**21.50**
	MMBench EN	47.76	45.67	48.03	**48.07**	47.76
	MMBench CN	33.98	37.31	37.28	37.31	**37.54**
	SEED-Bench	60.37	61.64	62.25	62.26	**62.42**
	SEED-Bench-2+	51.43	51.60	52.61	52.17	**52.88**
	ScienceQA	74.81	74.17	75.71	**76.55**	75.61
Visual specialty	ChartQA	41.20	41.60	56.12	**58.00**	56.64
	OCRBench	48.30	48.50	55.00	53.30	**55.90**
	TextVQA	48.83	48.72	56.92	57.23	**58.12**
	DocVQA	31.36	31.67	41.67	**43.71**	43.09
	HallusionBench	60.25	58.99	58.99	61.41	**62.04**
	VSR	49.02	46.97	52.54	**56.06**	55.56
	ViewSpatial	31.48	32.49	33.30	32.04	**34.21**
	RefCOCO	32.62	32.50	36.90	39.80	**46.53**
	CountBench	68.17	70.84	71.46	72.07	**72.48**

G VISUAL-EVIDENCE INTERVENTION PROTOCOL

Wrong-image replacement, patch corruption, and the layer-wise attention cut in this section share 300 items with the token-diversity, scale-error, and prediction–understanding analyses in Appendix H. The 300 items are drawn once and fixed across probes and arms. Scoring is teacher-forced: each item is scored twice per condition, and the metric is the full-vocabulary probability of the gold answer letter at the answer position (accuracy by argmax; NLL as the mean token-level negative log-likelihood of the gold letter).

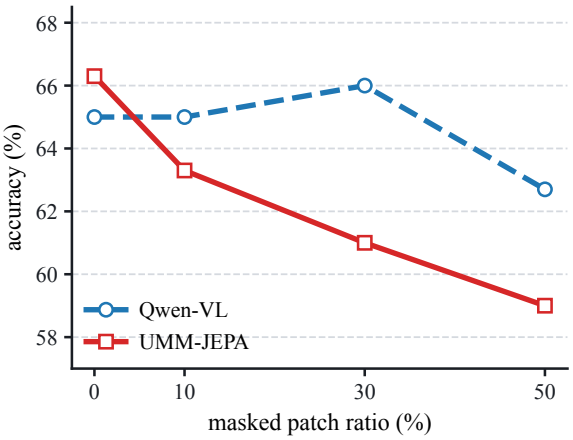


Figure 6: Corruption probe on the same 300 items: accuracy after zeroing r patch positions before the SigLIP2 front end ($r \in \{0, 10, 30, 50\}\%$), with grid size, token count, and the random mask held identical across the two arms. Qwen-VL is essentially flat (-2.3 points at 50%), whereas UMM-JEPA falls by 7.3 points.

Table 9: Layer-wise intervention: accuracy drop (percentage points) after blocking text→vision attention at layers $> k$, on the same 300 items (uncut accuracy 65.0 for Qwen-VL, 66.3 for UMM-JEPA).

Model	k=4	k=8	k=12	k=16	k=20	k=24
Qwen-VL	19.7	18.3	14.3	0.0	−1.0	−0.7
UMM-JEPA	22.7	22.3	19.7	2.7	0.3	0.7

G.1 Wrong image

The input image is replaced by a mismatched donor drawn from the same item pool; the donor is resized (bicubic) to the target's pixel size so the visual-token count is unchanged. Prompt, few-shot prefix, and gold answer are held fixed. Acc drop = Acc(correct) − Acc(wrong); NLL increase = NLL(wrong) − NLL(correct). Figure 4(a,b) reports the results.

G.2 Corruption

Before the SigLIP2 front-end, $k = \mathrm{round}(N_{\text{patch}} \cdot r)$ patch positions are zeroed with $r \in \{10\%, 30\%, 50\%\}$, plus a clean baseline; grid size and token count are unchanged. The random mask is identical between the two arms. This is random patch masking, not the training-time NSP cascade mask.

Figure 6 reports the accuracy under patch corruption. UMM-JEPA is *more sensitive* to the removal of visual evidence: at 50% corruption its accuracy falls by 7.3 points, whereas Qwen-VL falls by 2.3. This is not a robustness result and we do not present it as one. It is consistent with the wrong-image probe in indicating stronger reliance on the actual visual content, but it is equally compatible with lower robustness to missing pixels, so it is reported here rather than in the main text.

G.3 Layer-wise intervention

At every LLM decoder layer with 1-based index $> k$ ($k \in \{4, 8, 12, 16, 20, 24\}$; 28 layers total), attention scores from text queries to visual keys are set to $-\infty$; text→text, vision→vision (including UMM-JEPA's within-span bidirectional attention in uncut layers), vision→text, causality, and padding are preserved. Visual keys are image-token positions; vision start/end markers are treated as text. DeepStack residuals injecting ViT features into LLM layers 0–2 are unaffected at every k, so the probe measures late direct retrieval only. Drop = uncut accuracy − cut accuracy. Table 9 reports the per-k drop, also shown as Figure 4(c).

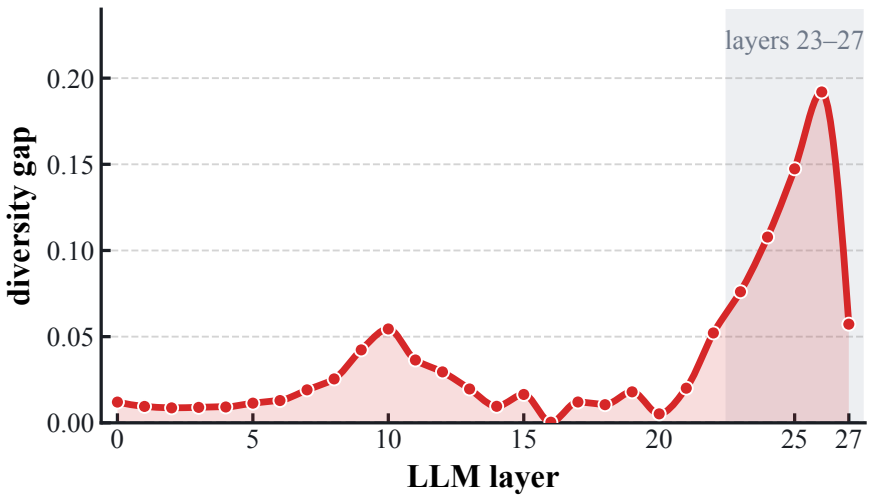


Figure 7: Per-layer visual-token diversity gap (UMM-JEPA minus Qwen-VL). Diversity at each layer is 1− the mean pairwise cosine among that layer's visual tokens, and each point is the mean gap over the 300 shared items. The shaded band marks layers 23–27, whose means are 0.486 for UMM-JEPA and 0.370 for Qwen-VL.

H Additional representation analyses

H.1 Linear probe protocol

The LLM and the visual front end are frozen and only the linear heads are trained. We use the final-layer visual-token hidden states (not the DeepStack-injected early layers). Training images are 500 images stratified over the 80 COCO categories (Lin et al., 2014) from `train2017` and validation images are 1000 images stratified the same way from `val2017` (seed 42), shared by all models. The object-mask probe assigns each visual patch the COCO semantic category of the instance whose mask covers the patch center, and background (class 0) when the center falls outside every instance, giving $80+1=81$ classes; when instances overlap, the smaller-area instance takes precedence. It is scored by mIoU. The same-object probe instead samples visual-token pairs within an image (up to 16 positive and 16 negative pairs per image) and predicts whether both patches belong to the same object, using the per-image instance identity as the label; instance identity does not enter the mIoU target. It is scored by accuracy. The object-mask probe therefore evaluates category-level spatial structure, whereas the same-object probe evaluates instance-level grouping. The head is trained with AdamW at learning rate 1×10^{-3} for 20 epochs for every model. Table 3 reports the results.

H.2 Late-layer token diversity

For each layer, token diversity $=1-$ mean pairwise cosine among visual tokens, computed in the same forward pass for both arms. UMM-JEPA reaches 0.486 versus 0.370 for Qwen-VL (Figure 7) when averaging layers 23–27, i.e., late-layer visual states remain more differentiated. This is a descriptive pairwise statistic and does not by itself demonstrate richer semantic content.

H.3 Scale-wise prediction error

For UMM-JEPA only, each item additionally runs an all-mask NSP forward in which the item's own image is the generation target and the conditioning text is the item's question. Per-scale error is 1−cosine: predictions are projected to the teacher dimension, both prediction and frozen SigLIP2 target are length-normalized, and the inner product is averaged over tokens (training weight 0.55 not applied). Figure 8 reports the per-scale errors. The error drops from 0.435 at scale 64 to 0.394 at scale 128 and then plateaus near 0.40, and correct and incorrect items separate mainly at the coarsest scale (0.419 vs. 0.465 at 64, and about 0.01 thereafter). In other words, the coarse scale already carries much of the understanding-relevant signal.

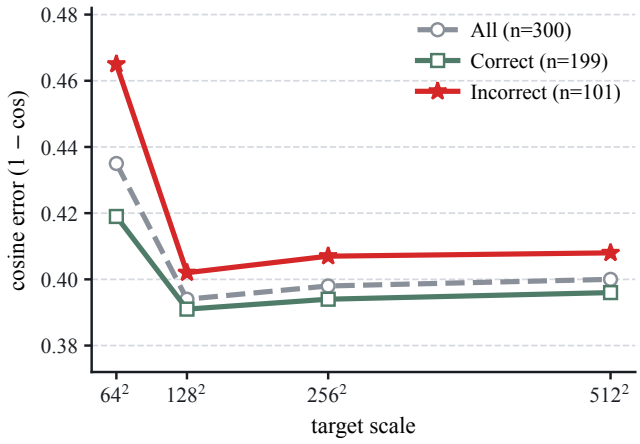


Figure 8: Scale-wise next-scale prediction error of UMM-JEPA (1−cosine) on the 300 shared items, split by whether the item was answered correctly: mean 0.435 / 0.394 / 0.398 / 0.400, correct (n=199) 0.419 / 0.391 / 0.394 / 0.396, incorrect (n=101) 0.465 / 0.402 / 0.407 / 0.408 at scales 64 / 128 / 256 / 512.

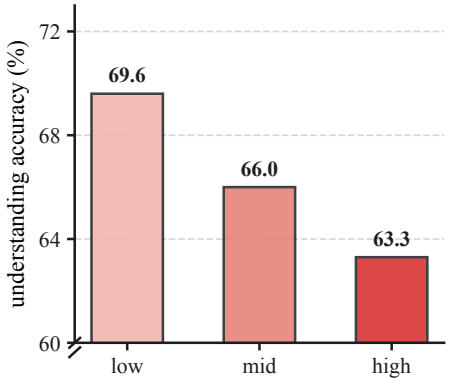


Figure 9: Understanding accuracy (%) by tercile of the 512-scale prediction error (300 items), split at the $1/3$ and $2/3$ quantiles. The value axis is truncated and the break is marked.

H.4 Prediction–understanding correlation

Using the 512-scale errors of the same 300 items, we split the items into low/mid/high terciles by the $1/3$ and $2/3$ quantiles and report the understanding accuracy of each tercile (Figure 9). Accuracy decreases from 69.6 to 66.0 to 63.3 with increasing prediction error, i.e., better next-scale prediction is associated with higher understanding accuracy, but the effect is modest and correlational: these analyses do not establish that better prediction causes better understanding.

I Generation evaluation protocol

Generation is evaluated with two metrics, both scored locally with no paid APIs. **GenEval** (Ghosh et al., 2023): the official 553 prompts covering objects, counts, colors, and positions, scored by a local Mask2Former (Cheng et al., 2022) (raw 0–1 score scaled by 100). **DPG** (Hu et al., 2024): 1065 long prompts probing entities and relations, scored by a local mPLUG-large (Li et al., 2022) (natively 0–100). The two metrics are reported separately and are never averaged with understanding scores.